**How to configure liveness and readiness probes for Containers.**

## **Define a liveness and readiness command**

1. SSH to the AWS Instance and create a pod object as <your-name>probe.yaml.

|  |
| --- |
| $ vim <your-name>-liveness-probe.yaml |

Paste the below content to the yaml file.

|  |
| --- |
| *apiVersion: v1 kind: Pod metadata:  labels:  test: <your-name>  name: <your-name> spec:  containers:  - name: liveness-<your-name>  image:* lovescloud/java-docker:latest *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  failureThreshold: 1  readinessProbe:  exec:  command:  - cat  - /tmp/health* |

Note :

**- touch /tmp/healthy; sleep 30; rm -rf /tmp/healthy; sleep 600**

This command will create a healthy file in the /tmp dir for the first 30 secs and then deletes this file. When the pod is being scheduled the liveness probe will be able to find the file for the first 30 secs and when this file gets deleted the probe will fail and hence will kill the container and recreates it.

**initialDelaySeconds: 5** Number of seconds after the container has started before liveness or readiness probes are initiated**.**

**periodSeconds: 5** How often (in seconds) to perform the probe. Default to 10 seconds.

**timeoutSeconds:** Number of seconds after which the probe times out. Defaults to 1 second.

Edit the fields **test name image (your hub image)**

2. Create a Pod

|  |
| --- |
| $ kubectl create -f <your-name>probe.yaml |

3. Within 30 seconds, view the Pod events:

## copy the pod name from the output of this command

|  |
| --- |
| **$ kubectl get po** |

|  |
| --- |
| $ kubectl describe pod <pod-name> |

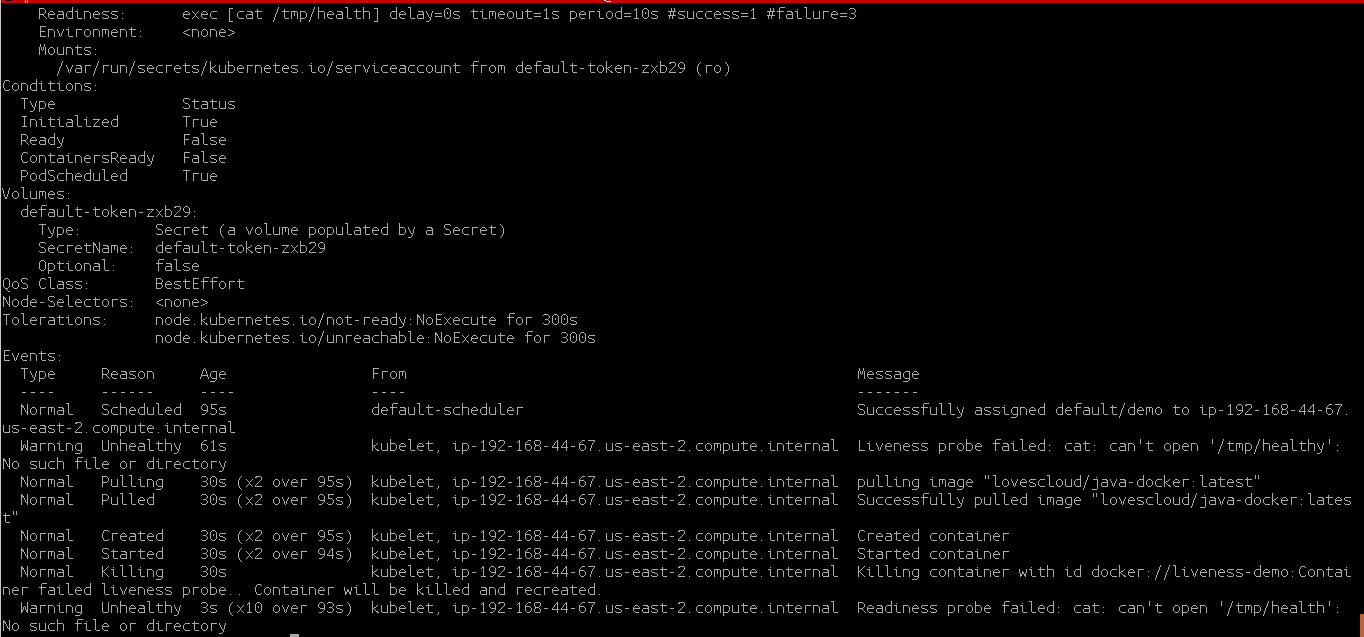
The output indicates that no liveness probes have failed yet:

|  |
| --- |
| FirstSeen LastSeen Count From SubobjectPath Type Reason Message --------- -------- ----- ---- ------------- -------- ------ ------- 24s 24s 1 {default-scheduler } Normal Scheduled Successfully assigned liveness-exec to worker0 23s 23s 1 {kubelet worker0} spec.containers{liveness} Normal Pulling pulling image "lovescloud/java-docker:latest" 23s 23s 1 {kubelet worker0} spec.containers{liveness} Normal Pulled Successfully pulled image "lovescloud/java-docker:latest" 23s 23s 1 {kubelet worker0} spec.containers{liveness} Normal Created Created container with docker id 86849c15382e; Security:[seccomp=unconfined] 23s 23s 1 {kubelet worker0} spec.containers{liveness} Normal Started Started container with docker id 86849c15382e |

4. After 35 seconds, view the Pod events again:

|  |
| --- |
| $ kubectl describe pod <pod-name> |

At the bottom of the output, there are messages indicating that the liveness probes have failed, and the containers have been killed and recreated.



5. Wait another 30 seconds, and verify that the Container has been restarted:

|  |
| --- |
| $ kubectl get pod <pod-name> |

The output shows that RESTARTS has been incremented:

|  |
| --- |
| NAME READY STATUS RESTARTS AGE demo 0/1 Running 3 4m |

If you login to your dashboard and check the pod events. You can see that the pod

Is being killed and recreated once the probe check fails.

