Configure liveness and readiness probes for Containers

SSH to your AWS Workstation
ssh devops@<public-ip-addr> of your Workstation
Password is : **Dev0p$!!/**
Replace <your-name> with your name throughout the lab.

1. Run the below commands to create a pod object as <your-name>probe.yaml.

```
$ sudo su
# cd application/
# curl -f https://pastebin.com/raw/wsFvgeSZ > <your-name>-probe.yaml
# vim <your-name>probe.yaml
```

Update <your-name> with your Name

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:

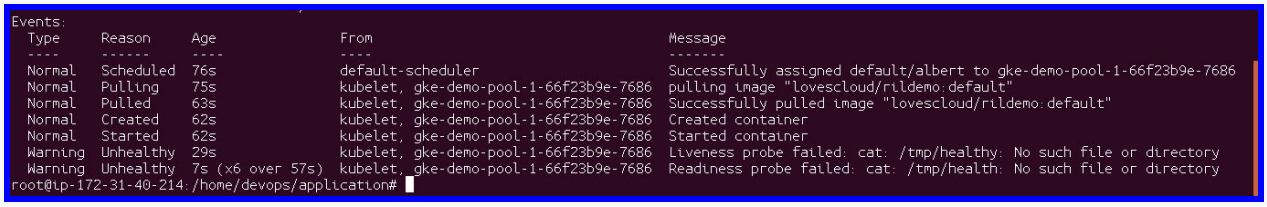
```
# kubectl get po <your-name>
```

```
root@ip-172-31-40-214:/home/devops/application# kubectl get po albert
NAME     READY   STATUS    RESTARTS   AGE
albert   0/1     Running   0          47s
root@ip-172-31-40-214:/home/devops/application#
```

4. Run the below command to describe the pod and check pod events.

```
# kubectl describe pod <your-name>
```

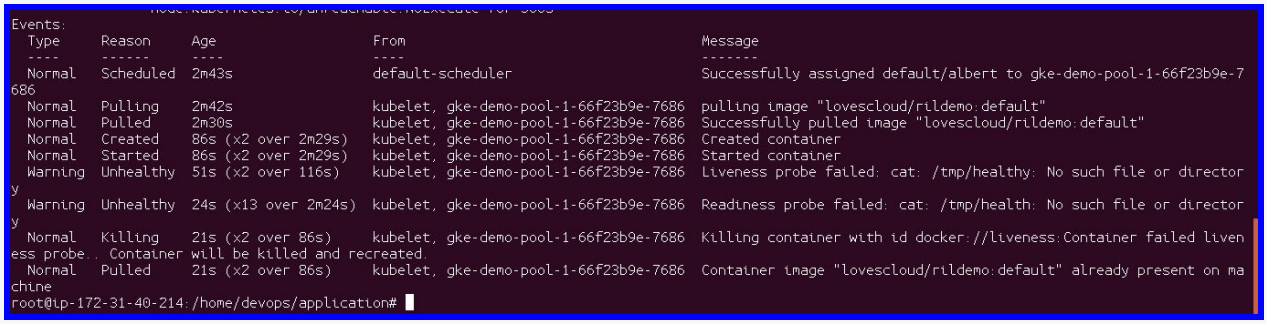
The output indicates that Liveness probe failed and Readiness probe failed.



5. After 30 seconds, view the Pod events again:

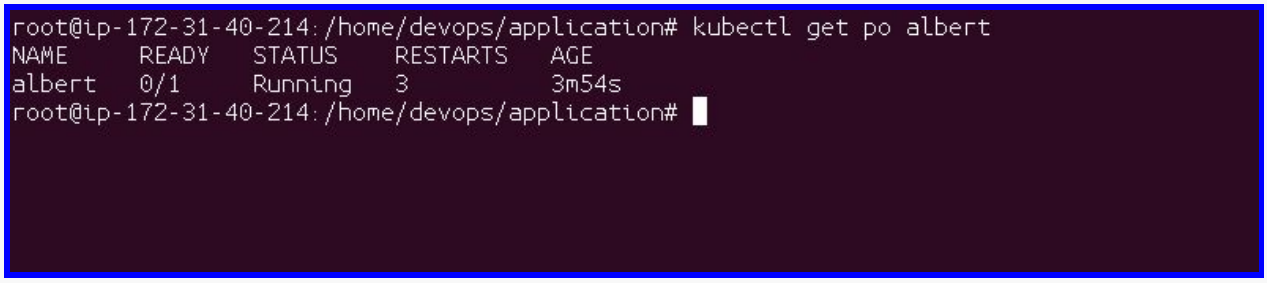
```
# kubectl describe pod <pod-name>
```

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



6. 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:

7. Login to your **Kubernetes Dashboard** and check the pod events. You can see that the pod Is being killed and recreated once the probe check fails.