Primitives and Self-Healing

What is a Primitive?

Kubernetes API primitives are data objects that represent the state of the cluster. They're often referred to as Kubernetes Objects.

Examples are

- Node
- Pod
- Replica Set
- Service

To see a full list of Kubernetes Objects available execute: *kubectl api-resources*

The Three Container States

Waiting – created but not running. A container which is in waiting stage, will still run operations like pulling images or applying secrets.

Running Pods – containers that are running without issues.

Terminated Pods – containers, which fail or complete their execution; stand terminated.

Pod Phase

Pending Pods – created but not running.

Running Pods – runs all the containers.

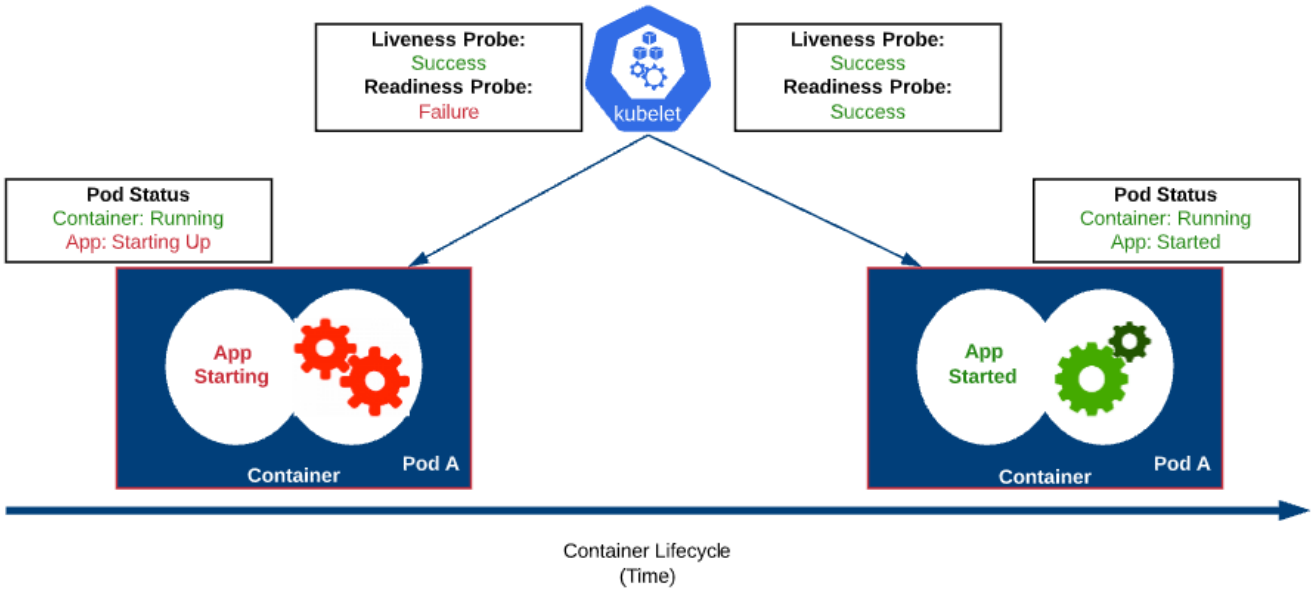
Succeeded Pods – successfully completed container life-cycle.

Failed Pods – minimum one container failed and all containers terminated.

Liveness and Readiness Probes

Kubernetes execute liveliness and readiness probes for the Pods to check if they function as per the desired state. The liveliness probe will check a container for its running status. If a container fails the probe, Kubernetes will terminate it and create a new container in accordance with the restart policy. The readiness probe will check a container for its service request serving capabilities. If a container fails the probe, then Kubernetes will remove the IP address of the related pod.

Probes include:
-ExecAction – to execute commands in containers.
-TCPSocketAction – to implement a TCP check w.r.t to the IP address of a container.
-HTTPGetAction – to implement a HTTP Get check w.r.t to the IP address of a container.



Hands On – Demonstrate Self-healing

I. Demonstrate Self-healing

Start by deploying 4 nginx pods, apply the deployment, and then verify.

```
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ cat doms-selfhealing.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment-sample
spec:
  selector:
    matchLabels:
      app: nginx
  replicas: 4
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.14.2
        ports:
        - containerPort: 80
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl apply -f doms-selfhealing.yaml
deployment.apps/nginx-deployment-sample created
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl get pods
NAME                                     READY   STATUS              RESTARTS   AGE
nginx-deployment-sample-66b6c48dd5-48f7s   1/1     Running             0          7s
nginx-deployment-sample-66b6c48dd5-7vqlz   0/1     ContainerCreating   0          7s
nginx-deployment-sample-66b6c48dd5-f5g64   1/1     Running             0          7s
nginx-deployment-sample-66b6c48dd5-n9mkx   1/1     Running             0          7s
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$
```

Now, delete a pod from the list and verify the new pod has a different name than the one you just deleted.

```
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl get pods
NAME                                     READY   STATUS    RESTARTS   AGE
nginx-deployment-sample-66b6c48dd5-48f7s   1/1     Running   0          2m4s
nginx-deployment-sample-66b6c48dd5-7vqlz   1/1     Running   0          2m4s
nginx-deployment-sample-66b6c48dd5-f5g64   1/1     Running   0          2m4s
nginx-deployment-sample-66b6c48dd5-n9mkx   1/1     Running   0          2m4s
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl delete pod nginx-deployment-sample-66
b6c48dd5-n9mkx
pod "nginx-deployment-sample-66b6c48dd5-n9mkx" deleted
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl get pods
NAME                                     READY   STATUS    RESTARTS   AGE
nginx-deployment-sample-66b6c48dd5-48f7s   1/1     Running   0          2m13s
nginx-deployment-sample-66b6c48dd5-7vqlz   1/1     Running   0          2m13s
nginx-deployment-sample-66b6c48dd5-dr6qr   1/1     Running   0          2s
nginx-deployment-sample-66b6c48dd5-f5g64   1/1     Running   0          2m13s
```

The change will happen quickly so as long as you can see a new nginx-deployment name, you have completed this training.