Troubleshooting Application Failures

Hands On – Troubleshoot a failing Application

In this lab we will deploy a pod that wont run. For the purposes of this training I will point out the issue and then we will resolve the root cause of failure from cloud shell.

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-pod
spec:
  containers:
    - name: nginx-configmap
      image: nginx:invalidversion
```

I. Deploy the faulty manifest:

Create the manifest above, apply your deployment to the cluster, and verify.

```
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ cat doms-faultypod.yaml
apiVersion: v1
kind: Pod
metadata:
  name: doms-pod
spec:
  containers:
    - name: nginx-configmap
      image: nginx:invalidversion
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl apply -f doms-faultypod.yaml
pod/doms-pod created
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl get pods
NAME       READY   STATUS             RESTARTS   AGE
doms-pod   0/1     ImagePullBackOff   0          3s
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$
```

We see under *Status* an message saying *ImagePullBackoff.*

II. Explore the reason for failure:

First let's pull the logs for the pod.

```
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl logs doms-pod
Error from server (BadRequest): container "nginx-configmap" in pod "doms-pod" is waiting to start: image can't be pulled
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$
```

We see that the container in my pod is waiting to apply an image but can't do so. Now let's look at the description of the pod. We see that the image is invalid.

```
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl describe pod doms-pod
Name:         doms-pod
Namespace:    default
Priority:     0
Node:         minikube/192.168.49.2
Start Time:   Thu, 24 Mar 2022 22:11:27 +0000
Labels:       <none>
Annotations:  <none>
Status:       Pending
IP:           172.17.0.3
IPs:
  IP:  172.17.0.3
Containers:
  nginx-configmap:
    Container ID:
    Image:          nginx:invalidversion
    Image ID:
    Port:           <none>
    Host Port:      <none>
    State:          Waiting
      Reason:       ErrImagePull
    Ready:          False
    Restart Count:  0
    Environment:    <none>
```

If look further down we also see in the events that a warning was triggered 20 seconds ago from the kublet. We have failed to pill an image because the description is unknown as well as the manifest.

```
                    node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason     Age                From               Message
  ----     ------     ----               ----               -------
  Normal   Scheduled  109s               default-scheduler  Successfully assigned default/doms-pod to minikube
  Warning  Failed     33s (x6 over 108s)  kubelet            Error: ImagePullBackOff
  Normal   Pulling    21s (x4 over 109s)  kubelet            Pulling image "nginx:invalidversion"
  Warning  Failed     20s (x4 over 108s)  kubelet            Failed to pull image "nginx:invalidversion": rpc error: code = Unknown desc = Error response
from daemon: manifest for nginx:invalidversion not found: manifest unknown: manifest unknown
  Warning  Failed     20s (x4 over 108s)  kubelet            Error: ErrImagePull
  Normal   BackOff    5s (x7 over 108s)   kubelet            Back-off pulling image "nginx:invalidversion"
```

III. Resolve the issue:

To resolve the issue we need to provide the correct version information for our pod. Let's edit the manifest to reflect version 1.14.2 as listed here:

https://kubernetes.io/docs/concepts/workloads/controllers/deployment/

Since we've changed the version and applied the new manifest we can see that the pod is back in working order.

```
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ nano doms-faultypod.yaml
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ cat doms-faultypod.yaml
apiVersion: v1
kind: Pod
metadata:
  name: doms-pod
spec:
  containers:
    - name: nginx-configmap
      image: nginx:1.14.2
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl apply -f doms-faultypod.yaml
pod/doms-pod configured
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$ kubectl get pods
NAME       READY   STATUS    RESTARTS   AGE
doms-pod   1/1     Running   0          5m46s
dominickhrndz314@cloudshell:~ (sandbox-io-289003)$
```

If your pod is now running and ready you have completed this training.