Lesson 4 Demo 2: Deployment Rolling Update using the Deployment File

This section will guide you to:

* Update deployment rolling by editing the deployment file

This lab has one sub-section, namely:

1. Update deployment rolling by editing the deployment file

**Note:** If you don’t have an existing Kubernetes cluster, refer to the Demo 1.1 of Lesson 1.

**Step 1:** Update deployment rolling by editing the deployment file

* Start the kubernetes cluster in your lab

**Note:** A rolling update applies changes to the configuration of the pods being managed by a replication controller. The changes can be passed as a new replication controller configuration file, or, if just an image has to be updated, a new container image can be specified directly.

* Applying update to deployment rolling by editing configuration file

Let’s say you are running version 1.7.9 of nginx:

*cat > replication-nginx-1.7.9.yaml*

apiVersion: v1

kind: ReplicationController

metadata:

name: my-nginx

spec:

replicas: 5

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: nginx:1.7.9

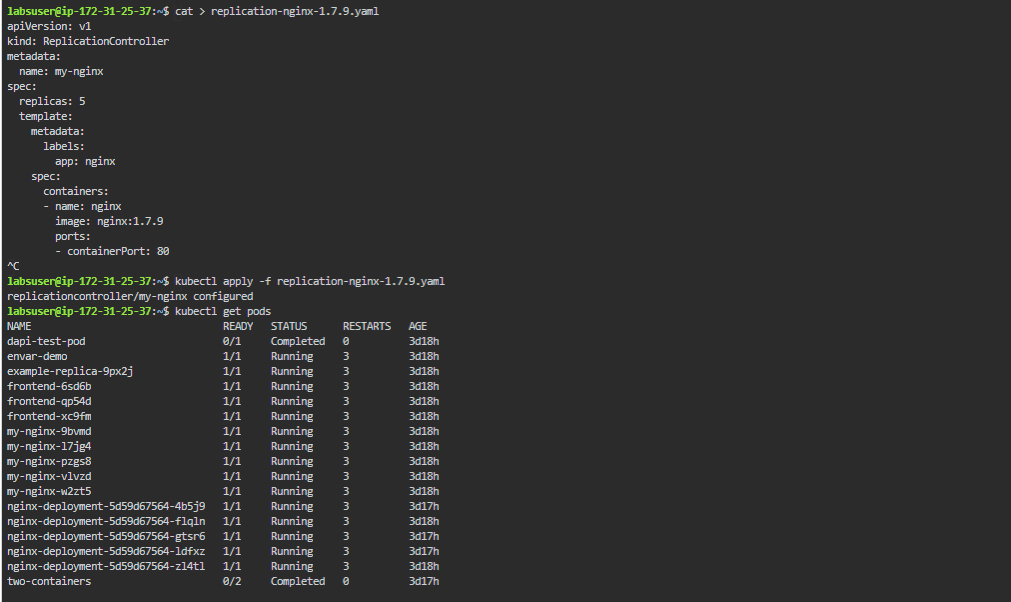
ports:

- containerPort: 80

* Next, submit the ReplicationController to the Kubernetes cluster using kubectl command as shown below:

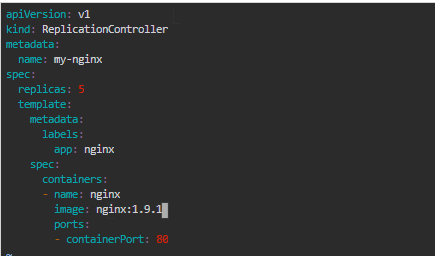
*kubectl apply -f replication-nginx-1.7.9.yaml*

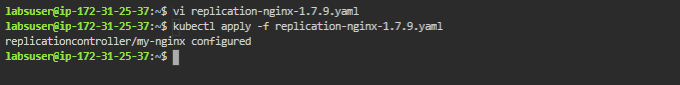
* Run **kubectl get pods** to verify nginx is running



* To update to version 1.9.1, you can edit the yaml file as shown below and update the version to 1.9.1. Next, submit the file in the cluster

*vi replication-nginx-1.7.9.yaml*



* Once edited, you can submit the updated yaml to the Kubernetes cluster using **kubectl apply -f replication-nginx-1.7.9.yaml** command as shown below:  
  
* You can see the rollout status as shown below after applying below changes:  
    
  *kubectl describe replicationcontrollers/my-nginx*