Lesson 2 Demo 6: Multiple Scheduler

This section will guide you to:

* Explain the use of multiple scheduler

This lab has one sub-section, namely:

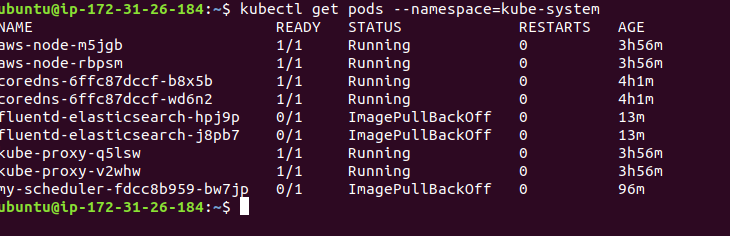
1. Use of multiple scheduler

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

**Step 1:** Using multiple scheduler

* Verify that the scheduler pod is running as shown below:

*kubectl get pods --namespace=kube-system*



* Create a pod config file to schedule a given pod using a specific scheduler

*vi pod1.yaml*

* Add the following code into the *pod1.yaml* file:

*apiVersion: v1*

*kind: Pod*

*metadata:*

*name: no-annotation*

*labels:*

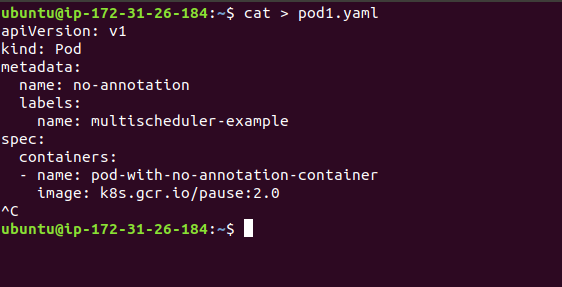
*name: multischeduler-example*

*spec:*

*containers:*

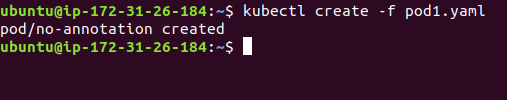
*- name: pod-with-no-annotation-container*

*image: k8s.gcr.io/pause:2.0*



* When no scheduler name is supplied, the pod is automatically scheduled using the default-scheduler.
* Save this file as **pod1.yaml** and submit it to the Kubernetes cluster using kubectl command as shown below:

*kubectl create -f pod1.yaml*



* We will now specify that pod should be scheduled using the scheduler that we deployed (custom scheduler)- my-scheduler.   
  **Note**: The value of **spec.schedulerName** should match the name supplied to the scheduler command as an argument in the deployment config for the scheduler, as shown below:

*vi pod2.yaml*

* Enter the following content in pod2.yaml file.

*apiVersion: v1*

*kind: Pod*

*metadata:*

*name: annotation-default-scheduler*

*labels:*

*name: multischeduler-example*

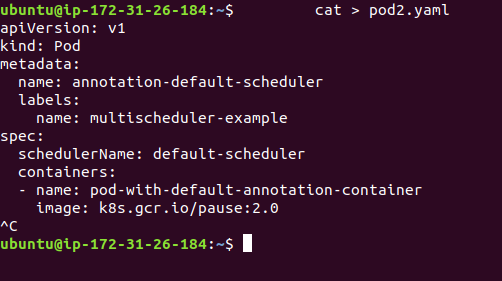
*spec:*

*schedulerName: default-scheduler*

*containers:*

*- name: pod-with-default-annotation-container*

*image: k8s.gcr.io/pause:2.0*



* Save this file as **pod2.yaml** and submit it to the Kubernetes cluster using kubectl commands as shown below:

*kubectl create -f pod2.yaml*

