Lesson 04 Demo 12

Creating Jobs

Objective: To demonstrate the creation of jobs in Kubernetes, allowing for efficient task management within the cluster

Tools required: kubeadm, kubectl, kubelet, and containerd

Prerequisites: A Kubernetes cluster should already be set up (refer to the steps provided in Lesson 02, Demo 01 for guidance).

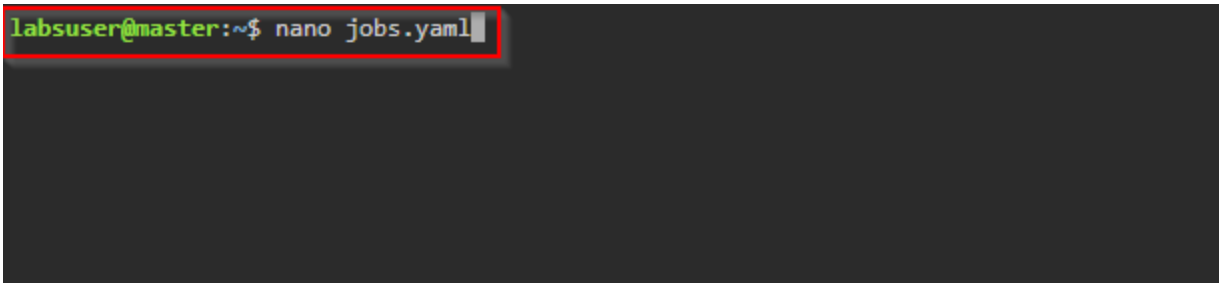
Steps to be followed:

1. Configure and set up the pod files

Step 1: Configure and set up the pod files

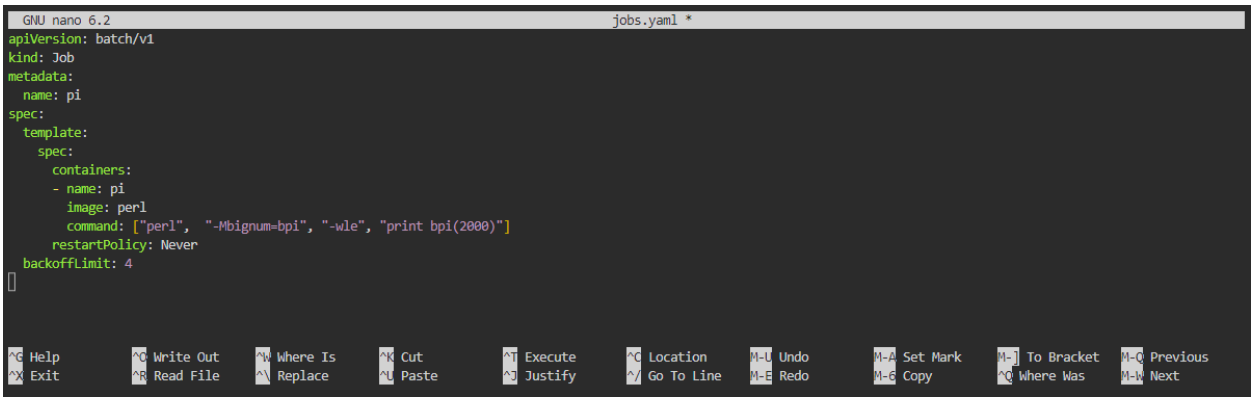
1.1 Create a YAML file by using the following command:

nano jobs.yaml



1.2 Add the following code in **jobs.yaml** to create the pod:

```
apiVersion: batch/v1
kind: Job
metadata:
  name: pi
spec:
  template:
    spec:
      containers:
      - name: pi
        image: perl
        command: ["perl",  "-Mbignum=bpi", "-wle", "print bpi(2000)"]
      restartPolicy: Never
  backoffLimit: 4
```



1.3 Press the **ctrl + o** keys to write, and then press the **enter** key; press the **ctrl + x** keys to exit the editor.

1.4 Use the **cat** command to validate the content of the **jobs.yaml** file

```
labsuser@master:~$ nano jobs.yaml
labsuser@master:~$ cat jobs.yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: pi
spec:
  template:
    spec:
      containers:
      - name: pi
        image: perl
        command: ["perl",  "-Mbignum=bpi", "-wle", "print bpi(2000)"]
      restartPolicy: Never
  backoffLimit: 4
labsuser@master:~$
```

1.5 Create the job resource by using the following command:

kubectl create -f jobs.yaml

```
labsuser@master:~$ nano jobs.yaml
labsuser@master:~$ cat jobs.yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: pi
spec:
  template:
    spec:
      containers:
      - name: pi
        image: perl
        command: ["perl",  "-Mbignum=bpi", "-wle", "print bpi(2000)"]
      restartPolicy: Never
  backoffLimit: 4
labsuser@master:~$ kubectl create -f jobs.yaml
job.batch/pi created
labsuser@master:~$
```

1.6 Verify the pod you created by using the following command:

kubectl get pods

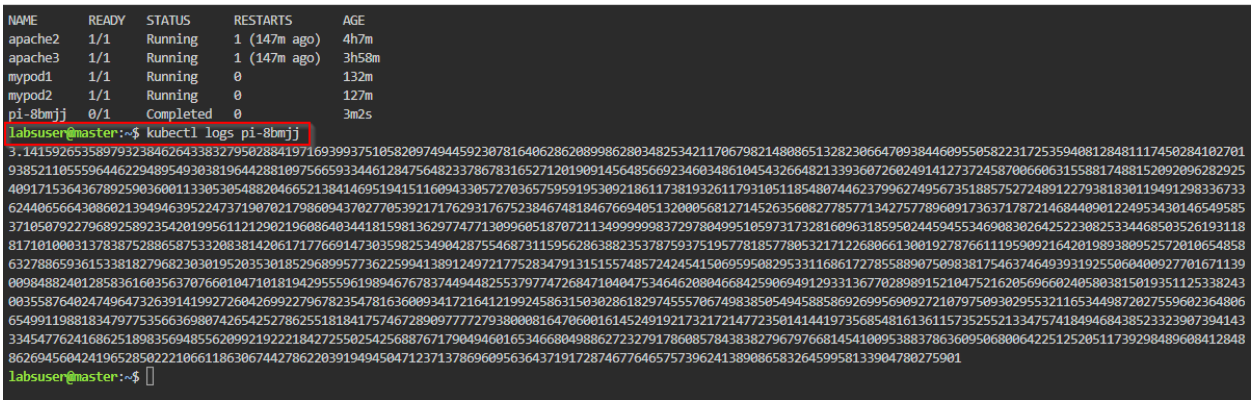
```
  name: pi
spec:
  template:
    spec:
      containers:
      - name: pi
        image: perl
        command: ["perl",  "-Mbignum=bpi", "-wle", "print bpi(2000)"]
      restartPolicy: Never
  backoffLimit: 4
labsuser@master:~$ kubectl create -f jobs.yaml
job.batch/pi created
labsuser@master:~$ kubectl get pods
NAME       READY   STATUS      RESTARTS       AGE
apache2    1/1     Running     1 (147m ago)   4h7m
apache3    1/1     Running     1 (147m ago)   3h58m
mypod1     1/1     Running     0              132m
mypod2     1/1     Running     0              127m
pi-8bmjj   0/1     Completed   0              3m2s
labsuser@master:~$
```

1.7 Copy the **name** of the pod

```
  name: pi
spec:
  template:
    spec:
      containers:
      - name: pi
        image: perl
        command: ["perl",  "-Mbignum=bpi", "-wle", "print bpi(2000)"]
      restartPolicy: Never
  backoffLimit: 4
labsuser@master:~$ kubectl create -f jobs.yaml
job.batch/pi created
labsuser@master:~$ kubectl get pods
NAME       READY   STATUS      RESTARTS       AGE
apache2    1/1     Running     1 (147m ago)   4h7m
apache3    1/1     Running     1 (147m ago)   3h58m
mypod1     1/1     Running     0              132m
mypod2     1/1     Running     0              127m
pi-8bmjj   0/1     Completed   0              3m2s
labsuser@master:~$
```

1.8 Replace <**Filename**> with the pod's name and verify the logs by using the following command, as shown in the screenshot below:

kubectl logs <Filename>



By following these steps, you have successfully configured and created jobs, enhancing your ability to automate and manage tasks effectively in a Kubernetes environment.