**Semester: V Name of Student:**

**Academic Year: 2022-23 Student ID:**

**Class / Branch: TE IT**

**Subject: Advanced Devops Lab (ADL)**

**Name of Instructor: Prof. Manasi Choche**

**EXPERIMENT NO. 04**

**Aim: To install Kubectl and execute Kubectl commands to manage the Kubernetes cluster and deploy Your First Kubernetes Application.**

### Kubernetes Deployments

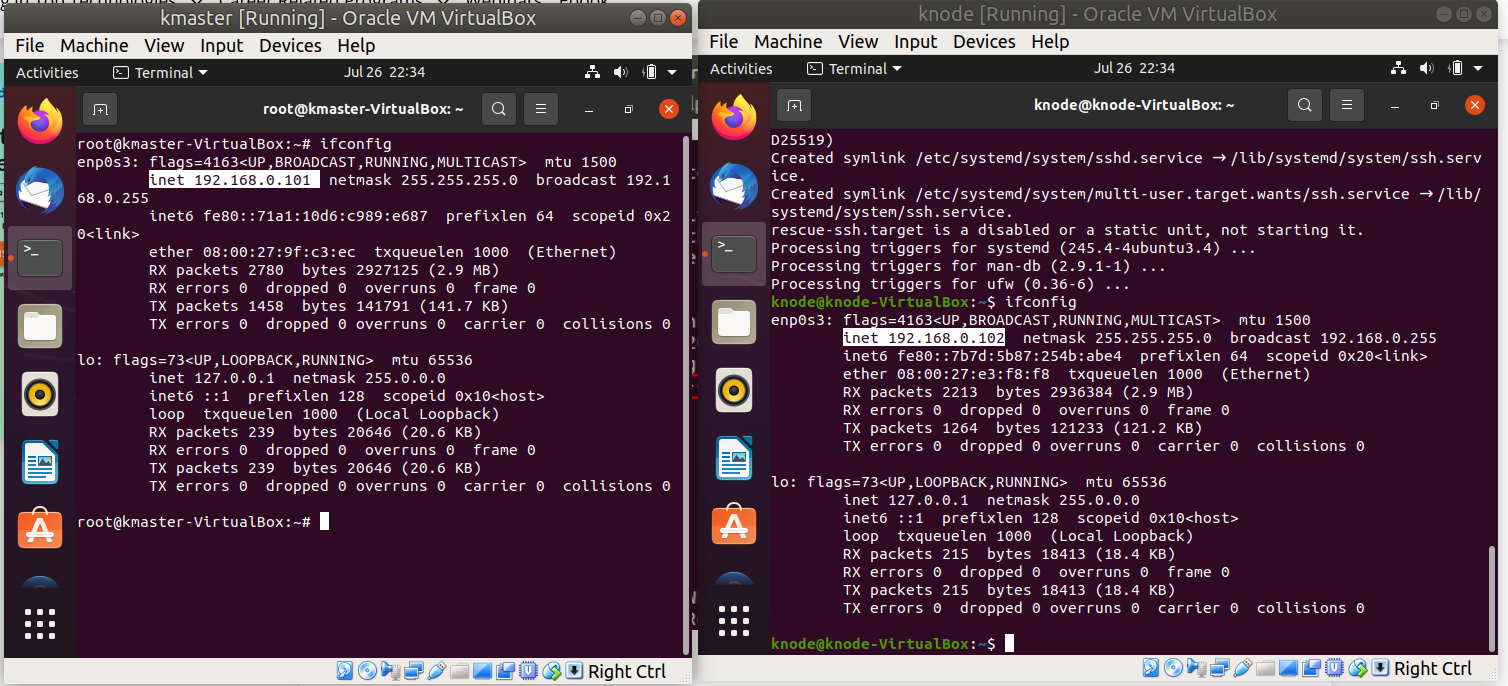
In Order to configure Kubernetes Cluster we require 2 systems we can consider it as one master and one slave.

I have launch two Virtual machines master as kmaster and slave as knode as shown in Fig.

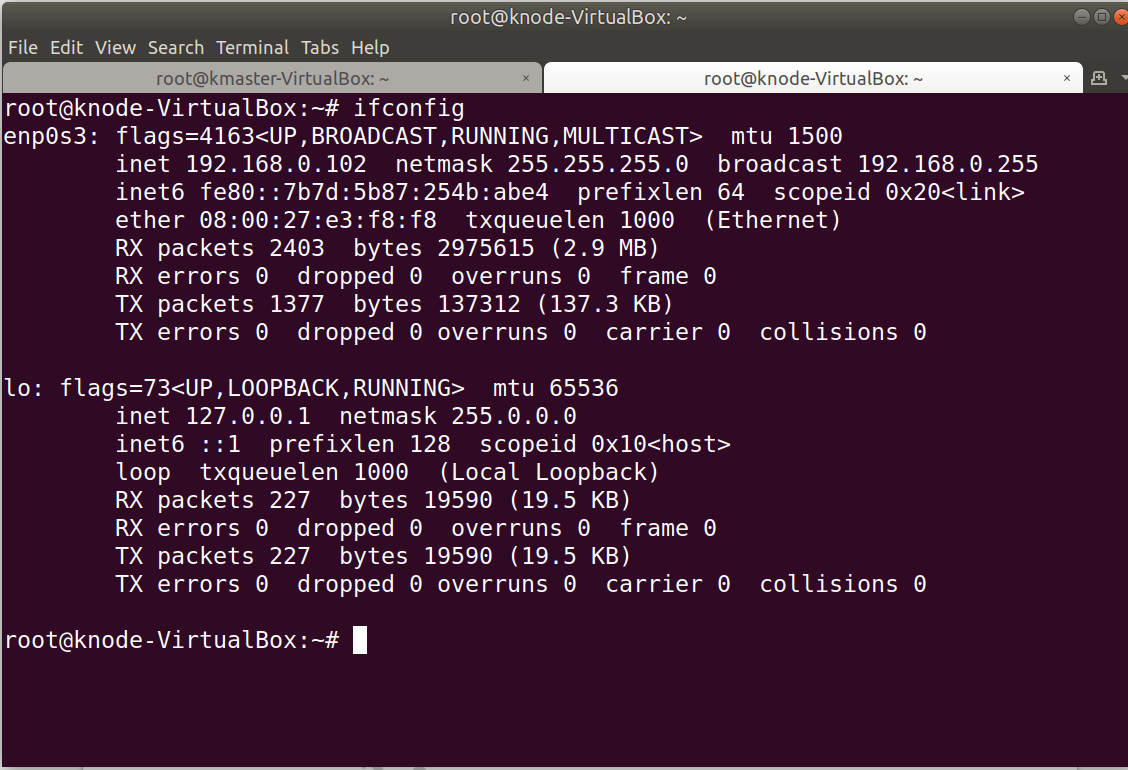
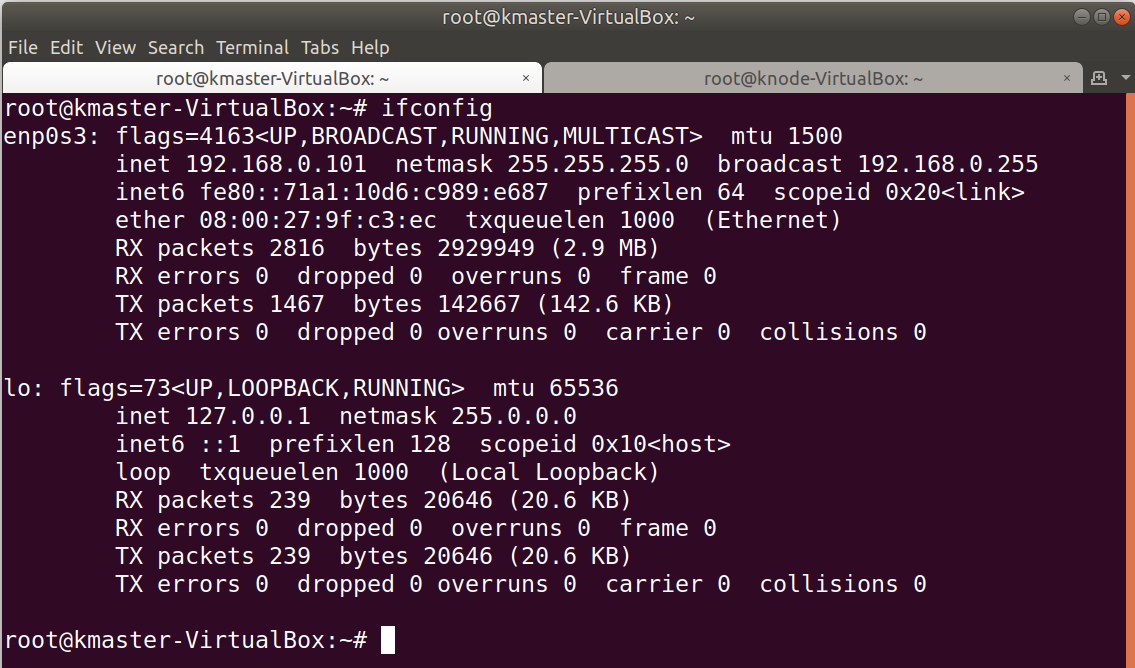
Configure static IP as:

**Master: 192.168.0.101 (kmaster)**

**Slave: 192.168.0.102 (knode)**



**Master: 192.168.0.101 (kmaster) Slave: 192.168.0.102 (knode)**

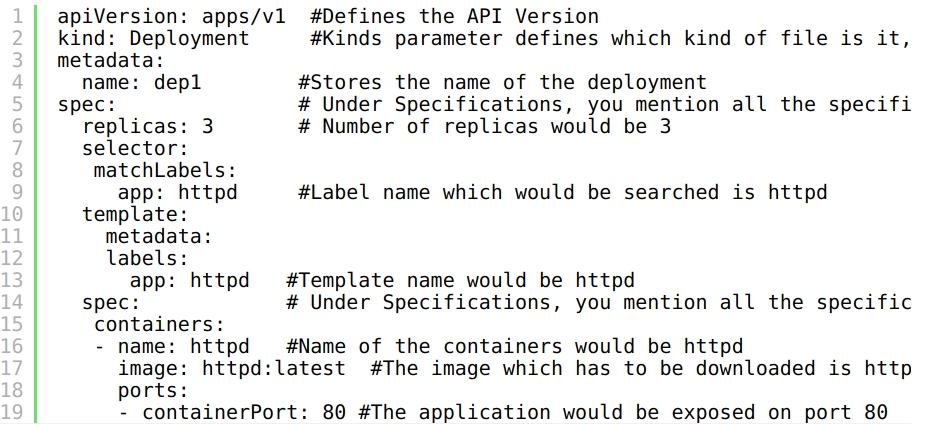


**For easy to connect and perform i have taken access of VMs on my machine for further steps.**

**Step 1:** First create a folder inside which you will create your deployment and service. After that, use an editor and open a Deployment file.

**Step 2:** Once you open the deployment file, mention all the specifications for the application you want to deploy. Here I am trying to deploy an httpd application.

**Relace httpd to nginx**



**Step 3:** After you write your deployment file, apply the deployment using the following command.

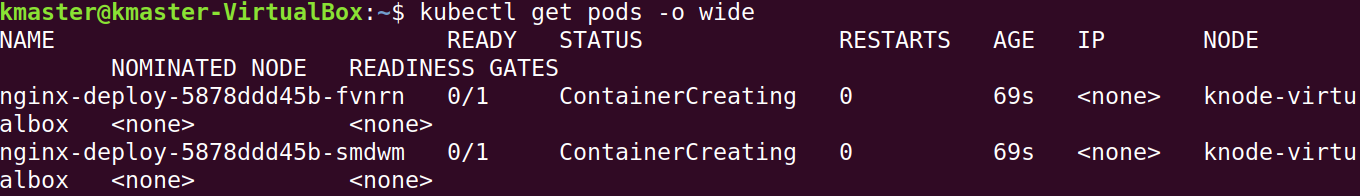
Here -f is a flag name used for the file name.





**Step 4:** Now, once the deployment is applied, get the list of pods running.

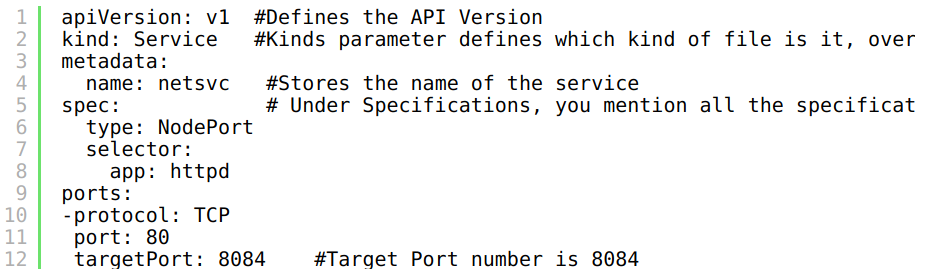
Here, -o wide are used to know on which node is the deployment running.

**Step 5:** After you have created a deployment, now you have to create a service. For that again use an editor and open a blank service.yaml file.

nano service.yaml

**Step 6:** Once you open a service file, mention all the specifications for the service.

**Relace httpd to nginx**



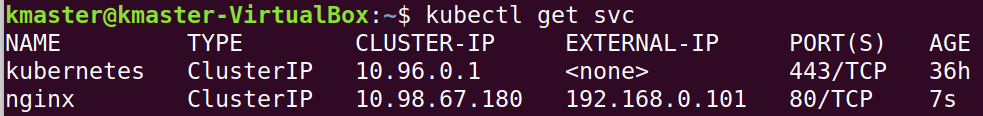
**Step 7:** After you write your service file, apply the service file using the following command.

kubectl apply -f service.yaml



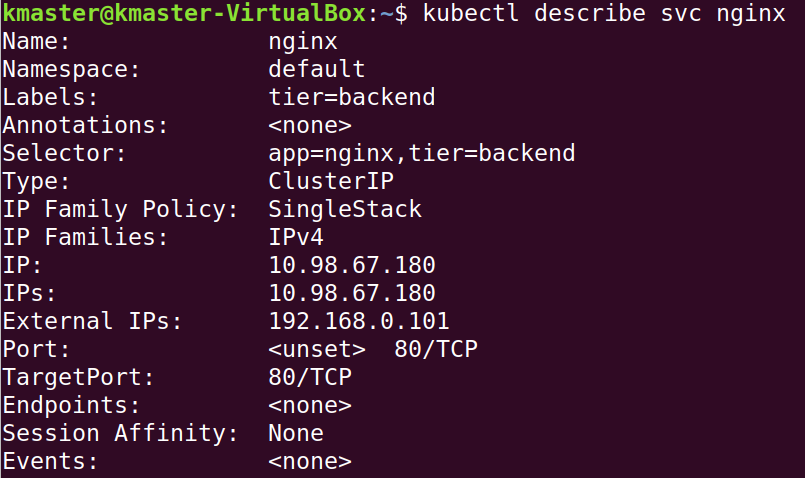
**Step 8:** Now, once your service is applied to check whether the service is running or not use the following command.

kubectl get svc

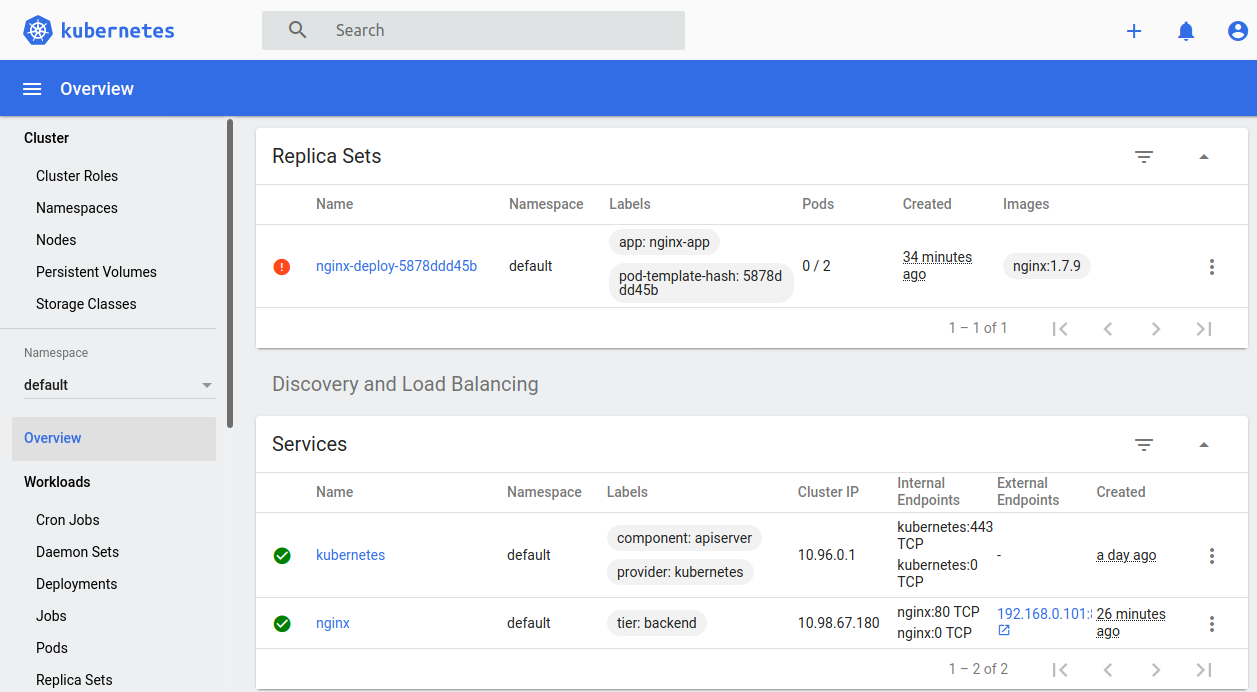
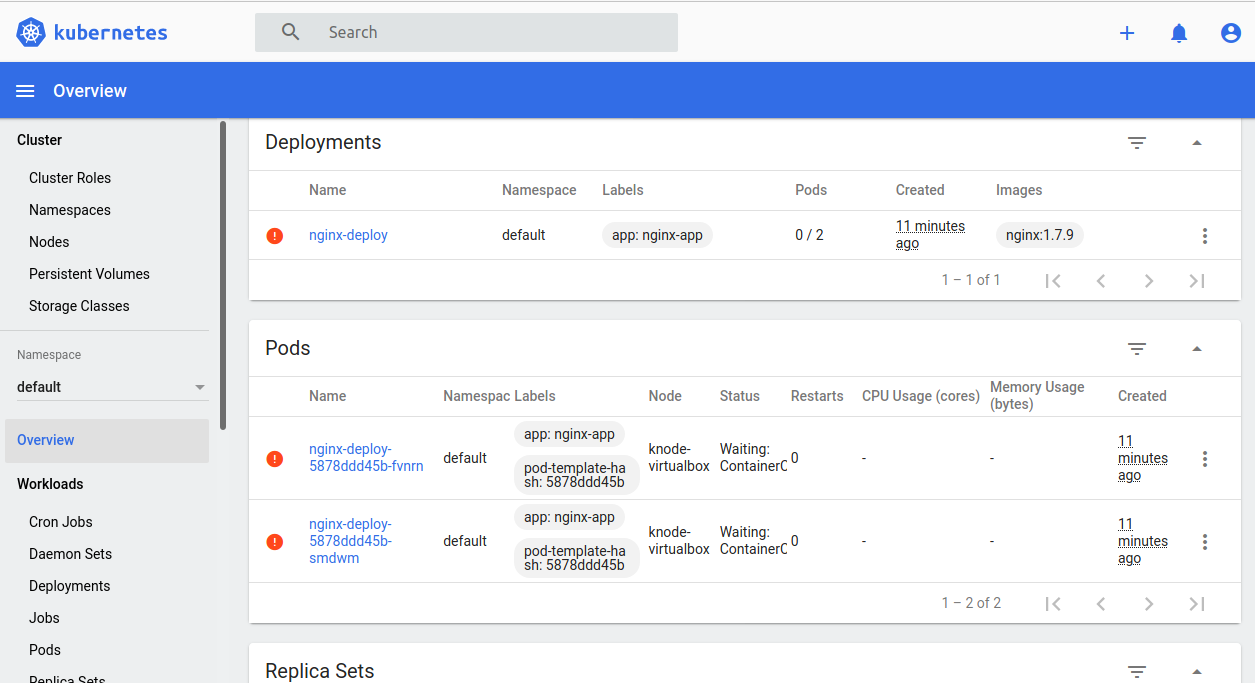


**Step 9:** Now, to see the specifications of service, and check which Endpoint it is binded to, use the following command.

kubectl describe svc <name of the service>



**Step10:** Check status in dashboard for Pods, Services, Replicas, etc.



**Conclusion: Write your own findings.**