

Step 1 - Creating a Pod with Label NGINX

Create a YAML File vi pod.yaml

```
apiVersion: v1
kind: Pod
metadata:
   name: label-demo
   labels:
       environment: production
       app: nginx
spec:
   containers:
   - name: nginx
   image: nginx
   ports:
   - containerPort: 80
```

kubectl create -f pod.yaml

root@kubernetes=master:/home/ubuntu# kubectl create =f pod.yaml pod/label=demo created

Valide if Pod is running correctly

Kubectl get pods

```
root@kubernetes-master:/home/ubuntu# kubectl<sup>l</sup>get<sup>E</sup>pods
NAME READY STATUSt<sup>@</sup>W:RESTARTShomAGEbuntu# sudo swapoff
label-demo 1/1 Running<sup>@w:@ker</sup>-1:/hom8subuntu#
```



Step 2 - Creating a Node Port Service

vi pod-node-selector.yaml

apiVersion: v1
kind: Service

metadata:

name: nginx-nodeport

spec:

type: NodePort

ports:

- port: 8080
 targetPort: 80
 nodePort: 30111

selector:
 app: nginx

kubectl create -f pod-node-selector.yaml

[root@kubernetes_master:/home/ubuntu#|kubectl|creates_f=nodeport_service.yaml=service.yaml=service.yaml=service.yaml=service.yaml=service.https://kubernetes.io/docs/setup

Validate Service is running NodePort

kubectl get svc

root@kubernetes-	-master:/home	e/ubuntu# kubect	egëtesvoght:	[preflight] Some	fatal er
NAME	TYPE	CLUŞTER Ω İ PFileA	VEXTERNAL-IP	-k port(s) es-kube	let AGE nf]
kubernetes	ClusterIP	10.957001 Port-	1≼none> Port	10 443/TCP in use	27h
nginx-nodeport	NodePort	10.108 ₹28.234⊝A	v andne ≯eetc	-k8080:30111/TCP	:.a.3≝t]



Step 3 - Test Node Port Service

kubectl get nodes -o wide

```
-o wide
· INTERNAL-IP
                                                                       EXTERNAL-IP OS-IMAGE
                                                                                                           KERNEL-VERSION
                                                                                                                             CONTAINER-RUNTIME
NAME
                    STATUSUPOROLES are AGE!
                                              VERSIONS
                                                         10.1.0.109
                                                                                      Ubuntu 18.04.3 LTS
                                                                                                           4.15.0-1057-aws
kubernetes-master
                                                                                                                             docker://19.3.8
                    Ready
                             master
                                      27h
                                              v1.18.1
                                                                       ⊲none>
                                                                                      Ubuntu 18.04.3 LTS
                                                                                                           4.15.0-1057-aws
                    Ready
                                      3h15m
                                                         10.1.0.188
                                                                                                                             docker://19.3.8
worker-1
                              <none>
                                              v1.18.1
                                                                        ⊲none>
                                               V1.18.1
                                                                                      Ubuntu 18.04.3 LTS
                                                                                                                             docker://19.3.8
worker-2
                                      26h
                                                         10.1.0.85
                                                                                                           4.15.0-1057-aws
                                                                        <none>
```

curl <Kubernetes-master_IP_Addr>:30111

curl <Worker-1_IP_Addr>:30111

curl <Worker-2_IP_Addr>:30111

```
root@kubernetes-master:/home/ubuntu#fourl 10.110.85:30111ape.canonic
<!DOCTYPE html>
⊲html>
⊲head>
<title>Welcome to nginx!</title>
⊲style>
        width: 35em;
       margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
</style>
</head>
-dody>
<h1>Welcome to nginx!</h1>
If you see this page, the inginx web server is successfully installed and whind enterprise
working. Further configuration is required totyping for cloud operations just got easier
For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.dbr/>
Commercial support is available at

√a href="http://nginx.com/">nginx.com</a>.√a>.√p> updates.

<em>Thank you for using nginx.</em>
</body>
</html>
```



Step 4 - Create a Service Cluster-IP

Use the same Pod created in the last steps

Create a Service Cluster-IP YAML File

vi cluster-ip-service.yaml

```
apiVersion: v1
kind: Service
metadata:
    name: nginx
spec:
    ports:
    - port: 8080
        targetPort: 80
    selector:
        app: nginx
```

kubectl create -f cluster-ip-service.yaml

root@kubernetes-master:/home/ubuntu# kubectl create -f cluster-ip-service.yaml service/nginix created | 34 packages can be updated.

Validate the Cluster-IP Service and the IP address

Nginx - Cluster IP - IP Addres

root@kubernetes-master:/home/ubuntu# kubectl get svc s								
NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE			
kubernetes	ClusterIP	10.96.0.1	⊲none>	443/TCP	27h			
nginix	ClusterIP	S101109182147t r	eanones ***	8080/TCP	10s			



Test the access to the Pod with the Cluster-IP Address

Example:

curl 10.109.82.47:8080

```
root@kubernetes-master:/home/ubuntu# curl http://10.109.82.47
۸Ċ
root@kubernetes-master:/home/ubuntu#"curl=10.109.82.47:8080"annel=1.18
<!DOCTYPE html>
⊲html>
⊲head>
<title>Welcome to nginx!</title>
⊲style>|
   body {
      width: 35em;
      margin: 0 auto; 0 updates are security update
       font-family: Tahoma, Verdana, Arial, sans-serif;
</style>
</head>
dody>
<h1>Welcome to nginx!</h1>ot@worker=1:/home/ubuntu#
√p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.
For online documentation and support please refer to
⊲a href="http://nginx.org/">nginx.org</a>.⊲br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>, 
</body>
</html>
```

Step 5 - Delete both services NodePort and Cluster-IP

kubectl delete svc nginx-nodeport

kubectl delete svc nginx

[root@kubernetes=master:/home/ubuntu# kubectl=deletelsvclnginx=nodeportes of dat service "nginx=nodeport" deleted from kubernetes=master (10.1.0.109): icmp_seq=