**Creating a Kubernetes Ingress**

**Note replace <your-name> with your name throughout the lab**

1. Create three Deployments by running the below commands.

2. SSH to the **haproxyVM on Azure or any of the master NODE** of the 3 AWS Clusters

3. Deployment 1

|  |
| --- |
| # kubectl run <your-name>-1 --image=lovescloud/nginxdemo:v1 --port=80 |

4. Deployment 2

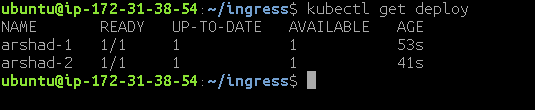
|  |
| --- |
| # kubectl run <your-name>-2 --image=lovescloud/nginxdemo:v2 --port=80 |

5. Deployment 3 - The default Backend echo server

|  |
| --- |
| # kubectl run echoserver-<your-name> --image=gcr.io/google\_containers/echoserver:1.4 --port=8080 |

6. Check if your deployments are running

|  |
| --- |
| # kubectl get deploy |

****

**Expose** your deployments over **NodePort**

7. Deployment 1

|  |
| --- |
| # kubectl expose deployment <your-name>-1 --target-port=80 --type=NodePort |

****

8. Deployment 2

|  |
| --- |
| # kubectl expose deployment <your-name>-2 --target-port=80 --type=NodePort |

****

9. Deployment 3

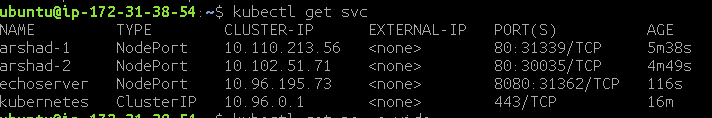
|  |
| --- |
| # kubectl expose deployment echoserver-<your-name> --target-port=8080 --type=NodePort |

****

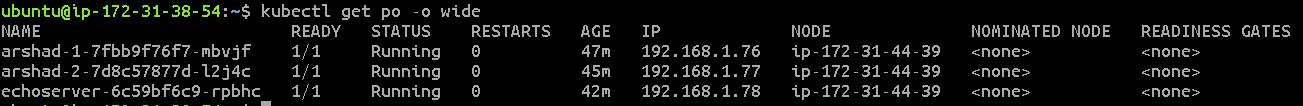
Accessing Deployments Over NodePort

10. Check the NODEPORT

|  |
| --- |
| # kubectl get svc |

****11. Check the Worker NODE where the POD’s are deployed.

|  |
| --- |
| # kubectl get po -o wide |



**Deploying the Ingress resouce.**

12. Now, create the ingress deployment yaml file and update the below fields.

|  |
| --- |
| # vim ingress-<your-name>.yaml |

**Replace <your-service1-name>** with your service name for the first deployment.

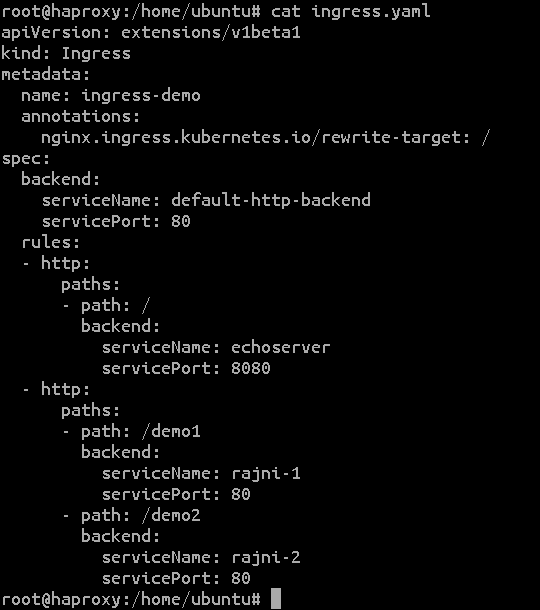
**Replace <your-service2-name>** with your service name for the second deployment

13. Paste the below script to ingress.yaml

|  |
| --- |
| apiVersion: extensions/v1beta1 kind: Ingress metadata:  name: ingress-demo--<your-name>  annotations:  nginx.ingress.kubernetes.io/rewrite-target: / spec:  backend:  serviceName: default-http-backend  servicePort: 80  rules:  - http:  paths:  - path: /  backend:  serviceName: echoserver-<your-name>  servicePort: 8080  - http:  paths:  - path: /<path1>  backend:  serviceName: <your-service1-name>  servicePort: 80  - path: /<path2>  backend:  serviceName: <your-service2-name>  servicePort: 80 |

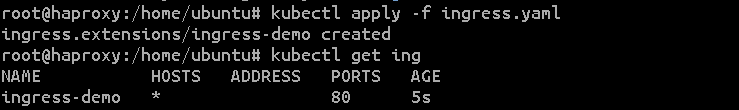
To save and exit once the changes have been made

Example of the deployment ingress.yaml file



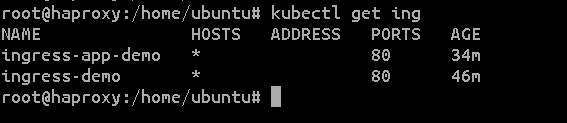
14. Deploy the Ingress.yaml

|  |
| --- |
| # kubectl apply -f ingress-<your-name>.yaml |



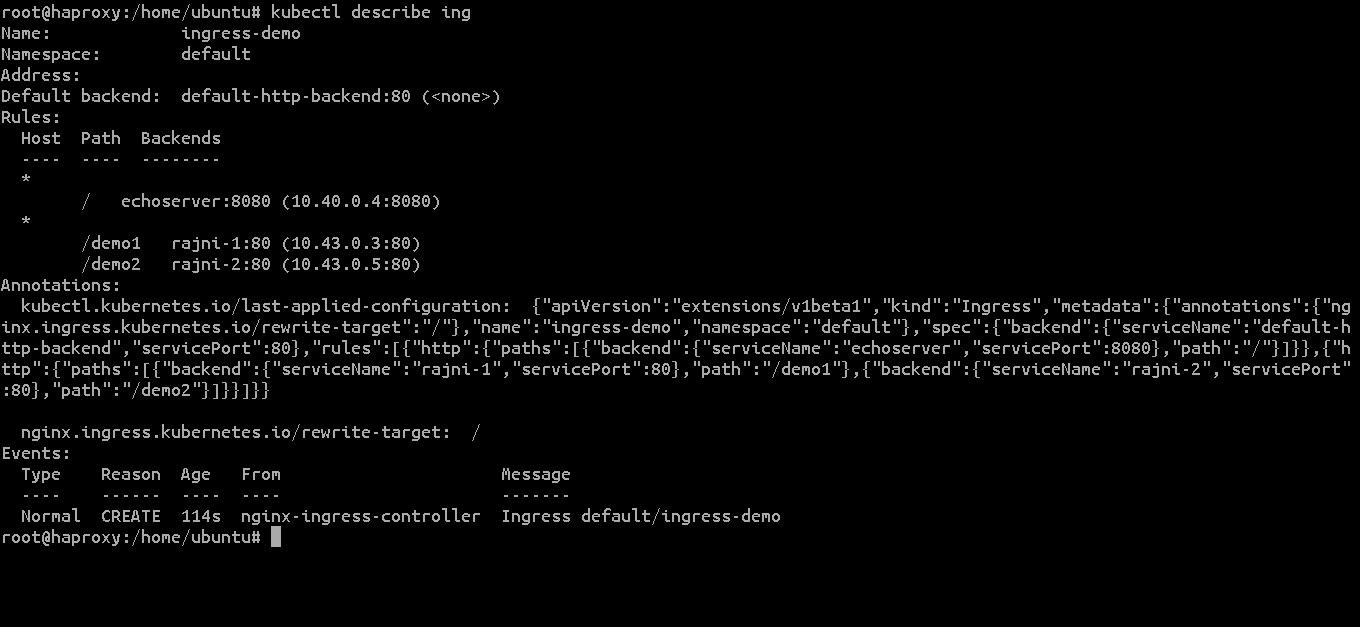
15. Check the Deployment

|  |
| --- |
| # kubectl get ing |



16. Describe the Ingress

|  |
| --- |
| # kubectl describe ing ingress-demo--<your-name> |



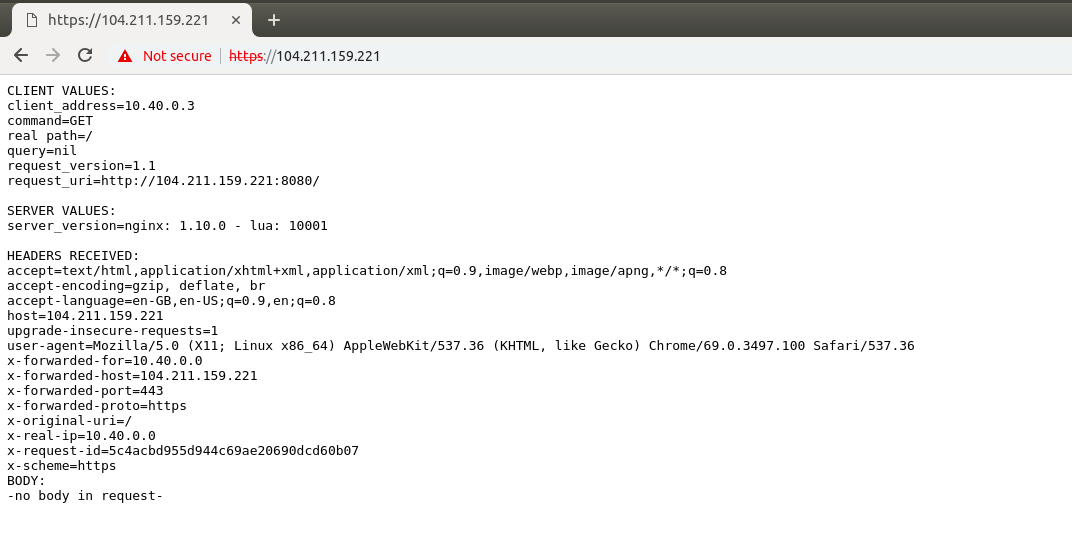
**17. Accessing the Deployments over Ingress.**

**The ingress in working behind the Haproxy-ingress VM.**

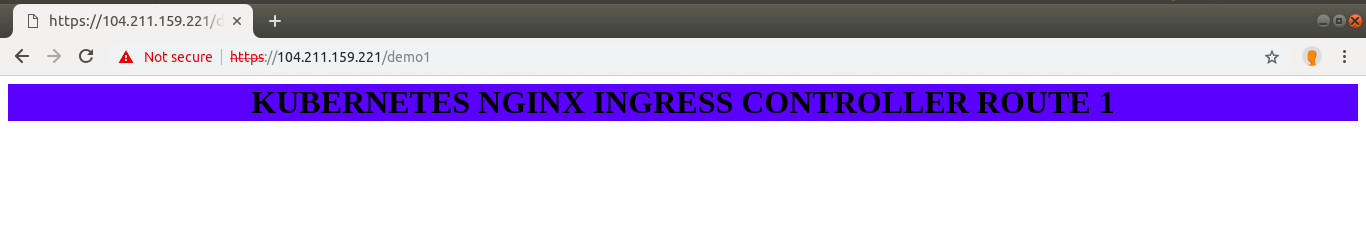
**To access the applications, try to access the public IP of the haproxy-ingress VM over port 80, followed by the /path you specified in the ingress.yaml**

**The default backend i.e the echoserver**

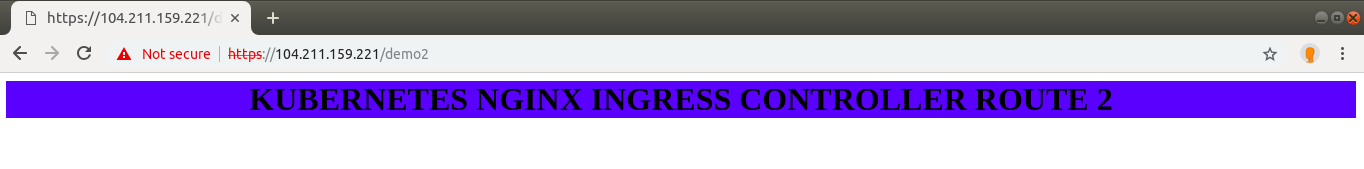
**https://<haproxy-ingress-public-ip>**

****

**https://<haproxy-ingress-public-ip>/demo1**

****

**https://<haproxy-ingress-public-ip>/demo2**

****