============================================

K8S POD & Service Manifest in Single YML

============================================

---

apiVersion: v1

kind: Pod

metadata:

name: javawebapppod

labels:

app: javawebapp

spec:

containers:

- name: javawebappcontainer

image: vinodses/my-web-app

ports:

- containerPort: 8080

---

apiVersion: v1

kind: Service

metadata:

name: javawebappsvc

spec:

type: LoadBalancer

selector:

app: javawebapp

ports:

- port: 80

targetPort: 8080

...

To execute manifest yml

# kubectl apply -f <manifest-yml>

Once Pod and service got created, we can see in EC2 dashboard Load balancer creation.

We can access our application using Load balancer DNS url

URL: LBR-DNS-URL/java-web-app/

To Get the pods details

# kubectl get pods

To Get service details

# kubectl get service

To delete all the resources, we have created

# kubectl delete all --all

===================================

k8s Node Port service manifest yml

====================================

---

apiVersion: v1

kind: Pod

metadata:

name: javawebapppod

labels:

app: javawebapp

spec:

containers:

- name: javawebappcontainer

image: vinodses/my-web-app

ports:

- containerPort: 8080

---

apiVersion: v1

kind: Service

metadata:

name: javawebappsvc

spec:

type: NodePort

selector:

app: javawebapp

ports:

- port: 80

targetPort: 8080

...

# kubectl apply -f k8s-node-manifest.yml

Your pod and service resource will created in the cluster and service type is nodeport

application can be accessed using the below url

http://public ip of the worker node:<random port number>/<application name>

============================

What is the node port number

============================

If we do not specify the node port in the service manifest yml file then k8s will assign a random port number in between 30000 and 32767

In this range it can take any port number and it will be use as node port number

The issue is every time we need allow the port number in our worker node NSG To overcome this we need to specify the node port in the manifest yml file

---

apiVersion: v1

kind: Pod

metadata:

name: javawebapppod

labels:

app: javawebapp

spec:

containers:

- name: javawebappcontainer

image: vinodses/my-web-app

ports:

- containerPort: 8080

---

apiVersion: v1

kind: Service

metadata:

name: javawebappsvc

spec:

type: NodePort

selector:

app: javawebapp

ports:

- port: 80

targetPort: 8080

nodePort: 31153

...

# kubectl apply -f <yml file>

To create my pod and services using the manifest yml file

# kubectl get pods

It will display whatever pods running in our k8s cluster

# kubectl get svc

It will display whatever services running in our k8s cluster

# kubectl get pods -o wide

It is used to display the pods are running on which worker node

# kubectl delete all --all

it will delete all the resources which is created in our k8s cluster

We can access our application using below url

http://workd-node-public ip:<worker node-port>/application name

e.g. : http://18.204.35.196:32153:30898/my-web-app/