**Deploying Microservices using Docker and Minikube Cluster**

Kubernetes works in a cluster. There is high chance have multiple master and worker nodes in a Kubernetes’s cluster .

To simulate a cluster is difficult in local , so we use minkube

Minikube is a tool that makes it easy to run Kubernetes locally. Minkube runs a single node Kubernetes cluster inside a Virtual machine on you laptop for users looking to try out Kubernetes.

So we install Minkube and CubeCtl . Cubectl is a cockpit for Kubernetes. It allows all configs and operations to manage Kubernetes, it’s a command tool.

We use follwoing to install minkube and cubectl in wnidows.

https://devopscube.com/minikube-mac/

macOS:

To start:

minikube start --driver qemu --network socket\_vmnet

Minkube status

Minkube dashboard

Minkube stop

Minkube delete

Use qemu driver to run minkube and cubectl

Don’t use docker driver as it has some issues.

A diagram of a computer

Description automatically generated

Now to make Kubernestes ready , we have to run all components in conatiner , to create containers we implement dockers .we will eb addin config in application so app can create docker image. We use docker file maven plugin to create docker images put of docker file and we will be pushing all the images to docker hub registry.

We have to use those images in our Kubernetes config files , so we create it for all components so we can apply all those components in Kubernetes cluster.

For user service and dpet service we wil be using deployment component and Service component. User serivec will be deployed using deployment component and fo that we will be creatin service component , it iwl allows to conec tot other applications like cluster Ip type-internal ip.

We will create Load balacer(deployment and load balancing service) for API gatweway . For hysterix we create nodeport service , it will enable particular node/VM to access all traffic.

For service resigstry we use deployment and stateful set .both are similar for component management to manage pods , but stateful set is for stateful application. Like db , registry to maintain host information or state .Bcoz we don’t want pod with hostname, we need particular hostname so all clients can connect to service registry , to get it uniquely we use stateful set using headless service . When we creta service we have cluster IP address but when you create headless service there isno IP but instead there is internal endpoints to expose pod to Kubernetes cluster to assign that value to clients so they can register k\* cluster.like

Podname-{replica-index}.{serviecName}.default.svc.cluster.local

A screenshot of a computer

Description automatically generated

Go to microservices project, add docker file plugin in pom.xml.Create docker repo and add it in pm under plugin also.

Create a Dcokerfile in project and add this .

FROM openjdk:11  
ARG JAR\_FILE=target/\*.jar  
COPY ${JAR\_FILE} user-service.jar  
ENTRYPOINT ["java","-jar","/user-service.jar"]  
EXPOSE 9002

Now go to termainal and execute this command

mvn clean package dockerfile:push

Do that to all projects and see them in docker hub

Now we add config so that service registry can connect to all services .

Podname-{replica-index}.{serviecName}.default.svc.cluster.local

Go to service registery application.yml file , we gice config about default zone and config

eureka:

instance:

preferIpAddress: true

hostname: eureka-0

client:

registerWithEureka: true

fetchRegistry: true

serviceUrl:

defaultZone: <http://eureka-0.eureka.default.svc.cluster.local:8761/eureka>

In cloud gareway applciaiton,ymal file

eureka:

instance:

preferIpAddress: true

hostname: eureka-0

client:

registerWithEureka: true

fetchRegistry: true

serviceUrl:

defaultZone: <http://eureka-0.eureka.default.svc.cluster.local:8761/eureka>

we add above in all services.

We now build and push to docker hub again now all of services.

Use command docker images to see images

Minkube status

Start minukube cluster by using

Minikube start

We create Kubernetes files in K8S folder. Install Kubernetes snippet plugin in sVS code fpr ease

User-service.yml we define dpeloyemtn and service

This can be found here

<https://github.com/shabbirdwd53/Springboot-k8s>

Add Kubernetes snippet plugin to trype easy for K8 files in VS code.Type dploeyment simple , beow will be coming on editor.After that we eill be creating serice , type serice complete below will be autopupulated.

apiVersion: apps/v1

kind: Deployment

metadata:

name: user-service-app

labels:

app: user-service-app

spec:

replicas: 1

selector:

matchLabels:

app : user-service-app

template:

metadata:

labels:

app: user-service-app

spec:

containers:

- name: user-service-app

image: dailycodebuffer/user-service:0.0.1

imagePullPolicy: Always

ports:

- containerPort: 9002

---

kind: Service

apiVersion: v1

metadata:

name: user-service-svc

spec:

selector:

app: user-service-app

ports:

- port: 80

targetPort: 9002

if you don’t add type in service. Default will be CLusterIp.

We wil be doing this for all microservices expect service registry snce it’s a statefule set.

For Hystric dashboard w euse Nodeport , we can use lcusterip , loadbalancer or ingress , just for use case

apiVersion: apps/v1

kind: Deployment

metadata:

name: hystrix-dashboard-app

labels:

app: hystrix-dashboard-app

spec:

replicas: 1

selector:

matchLabels:

app: hystrix-dashboard-app

template:

metadata:

labels:

app: hystrix-dashboard-app

spec:

containers:

- name: hystrix-dashboard-app

image: dailycodebuffer/hystrix-dashboard:0.0.1

imagePullPolicy: Always

ports:

- containerPort: 9295

---

apiVersion: v1

kind: Service

metadata:

name: hystrix-dashboard-svc

spec:

type: NodePort

ports:

- targetPort: 9295

port: 80

selector:

app: hystrix-dashboard-app

For Service-registry microservice , we create configmap component to define the variable to use that in environment (because in yml file we have defaultoone:${EUEKA SERVER ADDRESS}) ,and then for deployment we will create stateful set to get partialur hostname to use it in eureka clients , then we will create headless service(service without IP, needed for stateful set) and top of that we use service to connect to the pod.

We use secrets or credetnails or db details to store important details.

WE use <http://eureka-0.eureka.default.svc.cluster.local:8761/eureka> in format to reflect Podname-{replica-index}.{serviecName}.default.svc.cluster.local

Firs one below is configmap , headless service , stateful set, service

apiVersion: v1

kind: ConfigMap

metadata:

name: eureka-cm

data:

eureka\_service\_address: http://eureka-0.eureka:8761/eureka

---

apiVersion: v1

kind: Service

metadata:

name: eureka

labels:

app: eureka

spec:

clusterIP: None

ports:

- port: 8761

name: eureka

selector:

app: eureka

---

apiVersion: apps/v1

kind: StatefulSet

metadata:

name: eureka

spec:

serviceName: "eureka"

replicas: 1

selector:

matchLabels:

app: eureka

template:

metadata:

labels:

app: eureka

spec:

containers:

- name: eureka

image: dailycodebuffer/service-registry:0.0.1

imagePullPolicy: Always

ports:

- containerPort: 8761

env:

- name: EUREKA\_SERVER\_ADDRESS

valueFrom:

configMapKeyRef:

name: eureka-cm

key: eureka\_service\_address

---

apiVersion: v1

kind: Service

metadata:

name: eureka-lb

labels:

app: eureka

spec:

selector:

app: eureka

type: NodePort

ports:

- port: 80

targetPort: 8761

minikube status

kubectl get all

kubectl get nodes

kubectl cluster-info

minikube service list

cd k8s—go tot folder

we deploy by following command in k8s folder

kubectl apply -f ./

Now they are created

Go to terminal kubetcl get all

You will see services and pods getting created.

To see whats happening in pod , you can use

Kubectl describe pod/name of resource

Kubectl get pods :to get all pods

Kubectl get svc :to get all services

Kubectl get dpeloyments to get all deployemnts

Minikube service list to see all services we created . If no port then its are cluster ips, I f posrts to connect to other services externally.

As we deployed everything we need to do port forwarding to connect to localhost to k\* cluster.

Kubectl port-forward but we found pixelpoint to do it easily , in ideal PROD we will not be using port forwarding be using loadbalancer or ingress service. Install pixel point . <https://github.com/pixel-point/kube-forwarder>

Select minkube. And Namespace default , kind service , Name eureka-lb , local port 8761 , resourceport 80 click Add Resrouce and start click start ., you wil be redirect to eureka once you click link.

We do same for clodgateay serice , Name wil be clod gatway serice , local port 9191 resourceport 80,s tart port forward to access api gateway.

Only api servie is exposed so we can use postman to test the application with localhost:9191/departments/ url

Now caling: If we replicate these servicesreqeusts are going to different nodes ., when we scle up , someimts we get data and sometimes we don’t seince depts and users serviceis releated , thus we know wload balancer is working.

We use command to scale to see If lb is working or not

Kubectl scale –replicas=3 -f .\user-service.yml

You wil see info that its scaled.

Kubectl get all

When we use postman ,we get data sometimes or don’t. Thius lb is working.

Kubernetes dashboard to see in browser.

Yu can do port forwarding for hystrix dashboard to see it also

All this can be automated circle ci , Jenkins or github actions, github ci cd

To delete all in K8 cluster we can use one command.

Kubectl delete -f ./

You can add health checks , liveness probe , readiness probe , resource configuration s, graceful shutdown, dofferent namespaces for fronte end , nnd or backend or on team basis etc from docs.

End.