Levy Andrew Documentation EKS assignment

1. Create a Dockerfile with the content below:

   From python:3.10
   WORKDIR /calculator

   COPY . .
   run pip install -r requirements.txt
   ENV FLASK_APP=application.py
   EXPOSE 5000
   CMD flask run --host=0.0.0.0

2. Run CMD and cd to the calculator folder location in my case it was in my document

   cd Document/calculator

3. Run this command to build the image name calculator with the docker file well use above.

   docker build –t calculator .

4. To check if the image was build run this command

   docker images

5. To test this image and open port 8080 and redirect 5000 to it: run this command

   docker run -ti -p 8080:5000 calculator

6. Open a browser and type in the address

   localhost:8080

We have created a private Repositories in AWS so we are going to push the builded images into AWS with the following commands

7. Retagging the image to send to AWS Repositories. Run the command below

   docker tag calculator:latest 278144774015.dkr.ecr.us-east-2.amazonaws.com/calculator:latest

8. To check and to login in docker-hub. Run the command below

   aws ecr get-login-password --region us-east-2 | docker login --username AWS --password-stdin 278144774015.dkr.ecr.us-east-2.amazonaws.com

9. To push the image to docker-hub. Run the command below

   docker push 278144774015.dkr.ecr.us-east-2.amazonaws.com/calculator:latest

Task 2

Task 2: Deploying Your Flask app in Kubernetes.

1. Open a Command prompt

2. Type these commands to create a folder for the project and use it as the working directory

   mkdir calculator

   cd calculator

3. Type this command to create the cal deployment yaml file and click on yes

   notepad cal-deployment.yaml

4. Paste the following code in the file

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: cal-app
spec:
  replicas: 2
  selector:
    matchLabels:
      app: cal-app
  template:
    metadata:
      labels:
        app: cal-app
    spec:
      containers:
        - name: cal-app
          image: 278144774015.dkr.ecr.us-
east-2.amazonaws.com/calculator:latest
          ports:
            - name: http
              containerPort: 5000


---
apiVersion: v1
kind: Service
metadata:
  name: cal-app-service-nodeport
spec:
  type: NodePort
```

```
  selector:
    app: cal-app
  ports:
    - protocol: TCP
      port: 80
      targetPort: 5000
```

5. Type this command to create the ingress-controller yaml file and click on yes

   notepad ingress-controller.yaml

6. Paste the following code in the file

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: simple-ingress
  annotations:
    kubernetes.io/ingress.class: alb
    alb.ingress.kubernetes.io/scheme: internet-facing
    alb.ingress.kubernetes.io/target-type: instance
spec:
  rules:
    - http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: cal-app-service-nodeport
                port:
                  number: 80
```

7. Type this commands to create a cluster on EKS (Elastic Kubernetes Service)

   eksctl create cluster --name calculator

8. Type this commands to see the all node

   kubectl get nodes

9. Type this command to see the cluster and the loadbalancer

   eksctl get cluster

10. Type this command to view your cluster OpenID Connect provider URL

    aws eks describe-cluster --name calculator ^ --query "cluster.identity.oidc.issuer" --output text

11. Here is the output

https://oidc.eks.us-east-2.amazonaws.com/id/358F2FCF44FD690E8DA4E2862AF1F353

12. Type this command to list the IAM OIDC providers in your account
    aws iam list-open-id-connect-providers

13. Type this command to create an IAM OIDC identity provider for your cluster
    eksctl utils associate-iam-oidc-provider --cluster calculator –approve

14. Type this command to list the IAM OIDC providers in your account
    aws iam list-open-id-connect-providers

15. Type this command to download the (Role Base Access Control) rbac-role yaml (it contain the ingress controller and services) file from github and naming it rbac-role.yaml
    curl -o rbac-role.yaml ^ https://raw.githubusercontent.com/RobinNagpal/kubernetes-tutorials/master/06_tools/007_alb_ingress/01_eks/rbac-role.yaml

16. Type this command to apply the rbac-role yaml file
    kubectl apply -f rbac-role.yaml

17. Type this command to see the cluster that you just create
    kuber get alb-ingress-controller

18. Type this command to download iam policy json file
    curl -o iam_policy.json https://raw.githubusercontent.com/kubernetes-sigs/aws-load-balancer-controller/v2.3.0/docs/install/iam_policy.json

19. Type this command to create the AWS policy
    aws iam create-policy ^ --policy-name AWSLoadBalancerControllerIAMPolicy ^ --policy-document file://iam_policy.json

20. Type this command to create the service account
    eksctl create iamserviceaccount ^ --cluster= calculator ^ --namespace=kube-system ^ --name=aws-load-balancer-controller ^ --attach-policy-arn=arn:aws:iam::278144774015:policy/AWSLoadBalancerControllerIAMPolicy ^ --override-existing-serviceaccounts ^ --approve

21. Type this command to create certificate manager for the ingress controller
    kubectl apply ^ --validate=false ^ -f https://github.com/jetstack/cert-manager/releases/download/v1.5.4/cert-manager.yaml

22. Type this command to make the load balancer controller by downloading the file from GitHub
    curl -Lo v2_3_0_full.yaml https://github.com/kubernetes-sigs/aws-load-balancer-controller/releases/download/v2.3.0/v2_3_0_full.yaml

Levy Andrew Documentation EKS assignment

23. Edit the file that was downloaded v2_3_0_full.yaml (replace {cluster-name=*calculator*}

    notepad v2_3_0_full.yaml

24. Type this command to apply the v2_3_0_full yaml file

    kubectl apply -f v2_3_0_full.yaml

25. Type this command to view the controller

    kubectl get deployment -n kube-system aws-load-balancer-controller

26. Type this command to create the ngnix-deployment yaml file

    kubectl apply -f cal-deployment.yaml

27. Type this command to create the ingress-controller yaml file

    kubectl apply -f ingress-controller.yaml

28. Type this command to see the status of the pod

    kubectl get {name}

29. Go to AWS EC2 and on loadbalancer to get the DNS record and paste it in a web browser.

30. Type this command to delete the cluster

    eksctl delete cluster --name calculator

Levy Andrew Documentation EKS assignment