SWE Homework-2

Sai Venkata Sucheta Pallempati

G#01211890

URL of the AWS Survey Form: https://psvsucheta.s3.us-east-2.amazonaws.com/demo.html

URL of the AWS Error page : https://psvsucheta.s3.us-east-2.amazonaws.com/error.html

I have done the second part of the homework with a team of 5 members

Sai Ruchietha Maanvi - #g01224053

Sai Venkata Sucheta Pallempati - #g01211890

Aditya Varma Vetukuri - #g01213246

Kaushik Gedela - #g01166902

Jeevan Reddy Mamillapalli - #g01187250

My role : I was involved in creating the clusters. I was also involved in installations and configurations e.t.c.

Tools used-

- Docker Desktop
- Docker Hub
- Amazon EKS

Docker for Windows

Creation of a docker account:

- Navigate to docker hub using a web browser
- Sign up for a new account in docker hub
- Login with your new account and create a new repository and set it to public
- Download and install docker application.

To push containerized application into the repository

- Open docker desktop application and deploy war file
- Follow the below steps.
    - cd desktop
    - cd sucheta1 (war file which needs to be containerized)
    - touch Docker file
    - vim docker file
    - docker build -t hw2 (hw2 is the name of the war file which is assigned randomly) [to build the docker]
    - docker run -d -it -rm -p 60 :6060 hw2 (to run the docker)
    - docker ps
- Inside the vim Dockerfile, enter the following code:

```
From tomcst
MAINTAINER xyz
ADD sucheta1/usrr/locsl/tomcat/webapps
EXPOSE 7070
CMD ["cataline.sh" , "run"]
```

Tools required:

- Kubectl
- Dockerhub
- Amazon eks
- AWS CLI

Some other requirements : Windows 10, Python 3.8, Amazon aws version.

Team Project

Deploying the containerized application on Amazon eks :-

- create Amazon EKS service role in the IAM console
- create Amazon EKS Cluster VPC
- Create my own Amazon EKS Cluster on AWS console
- Configured the cluster with all the requirements given by the aws documentation.
- Create a Kubeconfig file and configured the AWS with my access key and secret keys.
- Tested using kubectl get svc
- Launch the worker nodes
- Enable worker nodes to join cluster, curl -o aws-auth-cm.yaml
- Downloaded the content of a sample yaml file and created a new yaml file and have configured it with my image names and target port.
- Configuration using the command kubectl apply -f awsauth-cm.yaml