**LAB1**

**I.Requirement:**

1. Architecture

Diagram

Description automatically generated

2.Server Requires

|  |  |  |
| --- | --- | --- |
| **No** | **Function** | **Type** |
| 1 | Jenkins container,Gitlab container | Ubuntu18 |
| 2 | K8S cluster | AKS,ACR |

3.Knowledge

- Know how to manage a resource , a resources group

- Know how to deploy app in Azure VM

- Know how to create AKS

**II.Step by step:**

\* Step in Azure :

- Login to Azure account then create resources group :

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

- Then we must create a virtual machine under above resources group

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

1.Gitlab container :

- Install gitlab by docker as below pic :

Text

Description automatically generated

- After gitlab is up , we exec to container and get generated password :

A computer screen capture

Description automatically generated with medium confidence

- Access to <http://PublicIP> then login by account root/ and change password

Graphical user interface, text, application

Description automatically generated

2.Jenkins server :

- Define variable for jenkins :

Text

Description automatically generated

* Build Jenkins image and create Jenkins container :

Text

Description automatically generated

* Automation create Jenkins agent in host machine

Text

Description automatically generated

* Automation for configuring credentials

Text

Description automatically generated

* Automation for creating a job

Text

Description automatically generated

Result :

Graphical user interface

Description automatically generated

3. K8S cluster (master & worker) :

- Install tool needed for setup AKS cluster :

Text

Description automatically generated

- Next configure azure cli by cmd : az login

Text

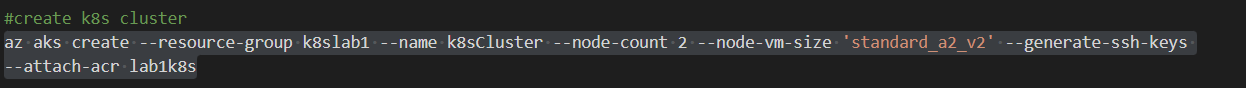
Description automatically generated

- Setup Azure container registry

Text

Description automatically generated

- Setup aks cluster :



Result :

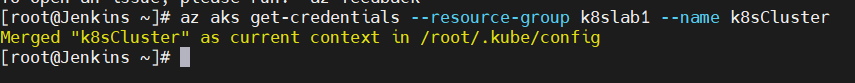
Text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

-connect to cluster

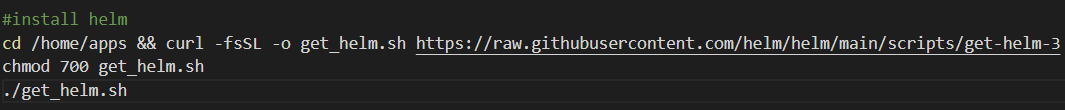


Graphical user interface, text

Description automatically generated

4.Setup app as the requirements

- Install helm in master node by below cmd (need to use helm version 3.8.2):

****

* Define helm chart:

.Run this command for creating a new chart :

*helm create <chart-name>*

.Go to the directory chart and define chart :

1.Chart.yaml

Text

Description automatically generated

2. \_\_helpers.tpl

Text

Description automatically generated

3. postgres-config.yml

Text

Description automatically generated

4. postgres-pvc.yaml

Text

Description automatically generated

5. service-be.yaml

Text

Description automatically generated

6.service-fe.yaml

Text

Description automatically generated

7. deployment-fe.yaml

Text

Description automatically generated

8. deployment-be.yaml

Text

Description automatically generated

.Create chart follow this format :

Text

Description automatically generated

.Run this command to check your helm chart :

*helm lint ./<chart-name>*

Text

Description automatically generated

.Run this command to see the templates with all values:

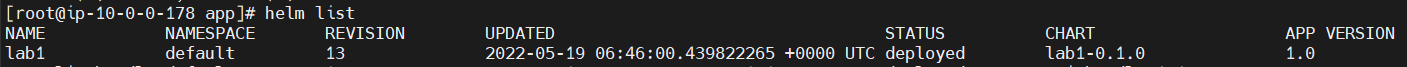
*helm template ./<chart-name>*

.Run this command to install app by helm chart :

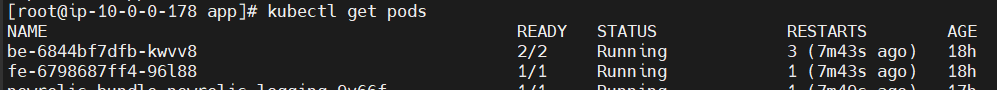
*helm install <name> <chart-name>/ --values <chart-name>/values.yaml*

.Run this command to see status chart:

*helm list*

**

.Run this command to see status pod and deploy:



*Text

Description automatically generated*

.Run this command to upgrade chart:

*helm upgrade <name> ./<chart-name>*

.After all pods backend is running , execute query sql to postgres db :

.exec to pods and run cmd:

* kubectl exec -i -t <pod-be> --container postgres -- /bin/bash
* psql -h localhost -U admin --password -p 5432 postgresdb
* execute query to postgress

Graphical user interface, text, application

Description automatically generated

4.Define app configuration and Jenkins pipeline :

.Edit DB information in queries.js in source be as below pic :

Text

Description automatically generated

.Edit BEURL in source FE

Text

Description automatically generated

.add agent in jenkins and then define Jenkins pipeline :

Text

Description automatically generated

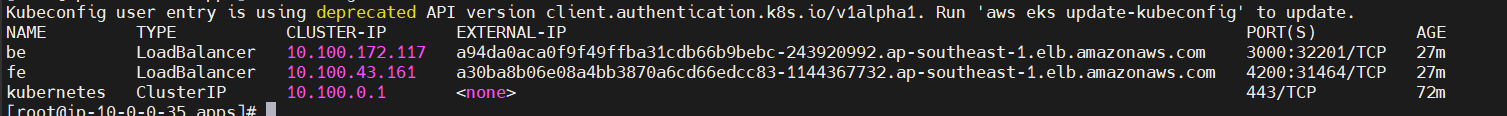
Text

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

We will use LoadBalancer type instead of ClusterIP type in svc . Result :



**Result** :

Graphical user interface, table

Description automatically generated

Graphical user interface

Description automatically generated

Setup new relic for host :

* Register free account
* Copy License
* Create file yaml by below cmd:

echo "license\_key: YOUR\_LICENSE\_KEY" | sudo tee -a /etc/newrelic-infra.yml

* Add repository:

sudo curl -o /etc/yum.repos.d/newrelic-infra.repo <https://download.newrelic.com/infrastructure_agent/linux/yum/amazonlinux/2/x86_64/newrelic-infra.repo>

* Refresh repository :

sudo yum -q makecache -y --disablerepo='\*' --enablerepo='newrelic-infra'

* Install :

sudo yum install newrelic-infra -y

Setup new relic for container by helm :

* Create file values and define as below pic

Text

Description automatically generated

* Add repo to helm chart :

helm repo add newrelic <https://helm-charts.newrelic.com>

* Install by helm chart :

helm upgrade --install newrelic-bundle newrelic/nri-bundle -f your-custom-values.yaml