gcloud config set compute/zone us-east1-b

git clone https://source.developers.google.com/p/$DEVSHELL\_PROJECT\_ID/r/sample-app

gcloud container clusters get-credentials jenkins-cd

kubectl create clusterrolebinding cluster-admin-binding --clusterrole=cluster-admin --user=$(gcloud config get-value account)

helm repo add stable https://charts.helm.sh/stable

helm repo update

helm install cd stable/jenkins

kubectl get pods

export POD\_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/component=jenkins-master" -l "app.kubernetes.io/instance=cd" -o jsonpath="{.items[0].metadata.name}")

kubectl port-forward $POD\_NAME 8080:8080 >> /dev/null &

printf $(kubectl get secret cd-jenkins -o jsonpath="{.data.jenkins-admin-password}" | base64 --decode);echo

In the Jenkins URL field, enter the following value: http://cd-jenkins:8080

In the Jenkins tunnel field, enter the following value: cd-jenkins-agent:50000

Click Save.

cd sample-app

kubectl create ns production

kubectl apply -f k8s/production -n production

kubectl apply -f k8s/canary -n production

kubectl apply -f k8s/services -n production

Branch Sources: Git

Project Repository: https://source.developers.google.com/p/[PROJECT\_ID]/r/sample-app

click Periodically if not...... 1 min

kubectl get svc

kubectl get service gceme-frontend -n production

git init

git config credential.helper gcloud.sh

git remote add origin https://source.developers.google.com/p/$DEVSHELL\_PROJECT\_ID/r/sample-app

git config --global user.email "<user email>"

git config --global user.name "<user name>"

git add .

git commit -m "initial commit"

git push origin master

open editor

SAVE

git checkout -b new-feature

git add Jenkinsfile html.go main.go

git commit -m "Version change\_version"

git push origin new-feature

curl http://localhost:8001/api/v1/namespaces/new-feature/services/gceme-frontend:80/proxy/version

kubectl get service gceme-frontend -n production

git checkout -b canary

git push origin canary

export FRONTEND\_SERVICE\_IP=$(kubectl get -o \

jsonpath="{.status.loadBalancer.ingress[0].ip}" --namespace=production services gceme-frontend)

git checkout master

git push origin master

export FRONTEND\_SERVICE\_IP=$(kubectl get -o \

jsonpath="{.status.loadBalancer.ingress[0].ip}" --namespace=production services gceme-frontend)

while true; do curl http://$FRONTEND\_SERVICE\_IP/version; sleep 1; done

kubectl get service gceme-frontend -n production

git merge canary

git push origin master

export FRONTEND\_SERVICE\_IP=$(kubectl get -o \

jsonpath="{.status.loadBalancer.ingress[0].ip}" --namespace=production services gceme-frontend)

Deployment:-

gcloud auth list

gsutil cat gs://cloud-training/gsp318/marking/setup\_marking\_v2.sh | bash

gcloud source repos clone valkyrie-app

cd valkyrie-app

cat > Dockerfile <<EOF

FROM golang:1.10

WORKDIR /go/src/app

COPY source .

RUN go install -v

ENTRYPOINT ["app","-single=true","-port=8080"]

EOF

docker build -t <Docker Image>:<Tag Name> .

cd ..

cd marking

./step1\_v2.sh

cd ..

cd valkyrie-app

docker run -p 8080:8080 <Docker Image>:<Tag Name> &

cd ..

cd marking

./step2\_v2.sh

cd ..

cd valkyrie-app

docker tag <Docker Image>:<Tag Name> gcr.io/$GOOGLE\_CLOUD\_PROJECT/<Docker Image>:<Tag Name>

docker push gcr.io/$GOOGLE\_CLOUD\_PROJECT/<Docker Image>:<Tag Name>

sed -i s#IMAGE\_HERE#gcr.io/$GOOGLE\_CLOUD\_PROJECT/<Docker Image>:<Tag Name>#g k8s/deployment.yaml

gcloud container clusters get-credentials valkyrie-dev --zone us-east1-d

kubectl create -f k8s/deployment.yaml

kubectl create -f k8s/service.yaml

git merge origin/kurt-dev

kubectl edit deployment valkyrie-dev

### change replicas from 1 to <Replicas Count>

### change <Tag Name> to <Updated Version> in two places

docker build -t gcr.io/$GOOGLE\_CLOUD\_PROJECT/<Docker Image>:<Updated Version> .

docker push gcr.io/$GOOGLE\_CLOUD\_PROJECT/<Docker Image>:<Updated Version>

docker ps

docker kill <take container\_id from above command>

export POD\_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/component=jenkins-master" -l "app.kubernetes.io/instance=cd" -o jsonpath="{.items[0].metadata.name}")

kubectl port-forward $POD\_NAME 8080:8080 >> /dev/null &

printf $(kubectl get secret cd-jenkins -o jsonpath="{.data.jenkins-admin-password}" | base64 --decode);echo

1 Username : admin

2 Password : {Code output from previous command}

3

Go through the following: -> Manage Jenkins -> Manage Credentials -> Jenkins -> Global credentials (unrestricted) -> Add credentials -> Kind: Google Service Account from metadata -> OK

-> Jenkins -> New Item -> Name : valkyrie-app -> Pipeline -> Pipeline script from SCM -> Set SCM to git -> OK -> Pipeline -> Script: Pipeline script from SCM -> SCM: Git -> Repository URL: {find it using command: gcloud source repos list} -> Credentials: {Project id} -> Apply -> Save

*YAML*

sed -i "s/green/orange/g" source/html.go

sed -i "s/YOUR\_PROJECT/$GOOGLE\_CLOUD\_PROJECT/g" Jenkinsfile

git config --global user.email "you@example.com" //Email

git config --global user.name "student..." // Username

git add .

git commit -m "built pipeline init"

git push

*BASH*