|  |
| --- |
| GSP330 : Implement DevOps in Google Cloud: Challenge Lab :- |
|  |  |
|  | ---------------------------------------------------------------------------------------------------------------------------------------------- |
|  |  |
|  | 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://kubernetes-charts.storage.googleapis.com/ → this might not work… use the next line of code instead… |
|  |  |
|  | 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 $(kubectl get secret cd-jenkins -o jsonpath="{.data.jenkins-admin-password}" | base64 --decode);echo |
|  |  |
|  |  |
|  | cd sample-app |
|  | kubectl create ns production |
|  | kubectl apply -f k8s/production -n production |
|  | kubectl apply -f k8s/canary -n production |
|  | kubectl apply -f k8s/services -n production |
|  |  |
|  | kubectl get svc |
|  | kubectl get service gceme-frontend -n production |
|  |  |
|  |  |
|  | ##Branch Sources: Git |
|  | - Project Repository: https://source.developers.google.com/p/[PROJECT\_ID]/r/sample-app |
|  | - Credentials: qwiklabs service account |
|  |  |
|  | click Periodically if not...... 1 min |
|  |  |
|  | git init |
|  | git config credential.helper gcloud.sh |
|  | git remote add origin https://source.developers.google.com/p/$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 |
|  |  |
|  |  |
|  | #save |
|  |  |
|  |  |
|  | git checkout -b new-feature |
|  |  |
|  | git add Jenkinsfile html.go main.go |
|  | git commit -m "Version 2.0.0" |
|  | 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) |