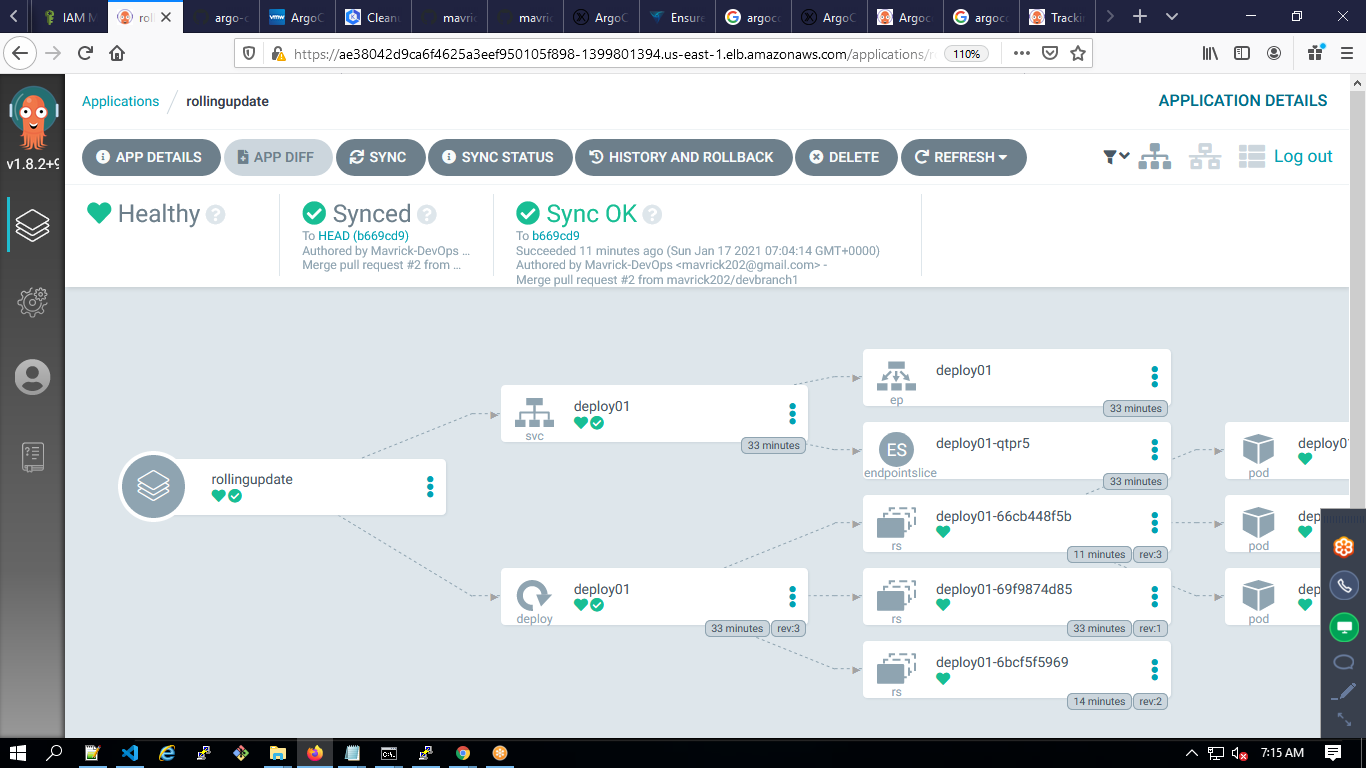
1. Deploy a 2-Node K8S Cluster Using Kops
2. Clone or Fork the repo <https://github.com/mavrick202/argocdtesting.git>
3. Visit following URL for step-by-step deployment of ArgoCD

<https://www.eksworkshop.com/intermediate/290_argocd/install/>

1. #Deploying ArgoCD and Deploying application
2. kubectl create namespace argocd
3. kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml
4. VERSION=$(curl --silent "https://api.github.com/repos/argoproj/argo-cd/releases/latest" | grep '"tag\_name"' | sed -E 's/.\*"([^"]+)".\*/\1/')
5. sudo curl --silent --location -o /usr/local/bin/argocd https://github.com/argoproj/argo-cd/releases/download/$VERSION/argocd-linux-amd64
6. sudo chmod +x /usr/local/bin/argocd
7. kubectl patch svc argocd-server -n argocd -p '{"spec": {"type": "LoadBalancer"}}'
8. export ARGOCD\_SERVER=`kubectl get svc argocd-server -n argocd -o json | jq --raw-output .status.loadBalancer.ingress[0].hostname`
9. ARGO\_PWD=`kubectl get pods -n argocd -l app.kubernetes.io/name=argocd-server -o name | cut -d'/' -f 2`
10. argocd login $ARGOCD\_SERVER --username admin --password $ARGO\_PWD --insecure
11. (or)
12. Access Load Balancer on Port 80
13. CONTEXT\_NAME=`kubectl config view -o jsonpath='{.contexts[].name}'`
14. argocd cluster add $CONTEXT\_NAME
15. argocd app create rollingupdate --repo https://github.com/mavrick202/argocdtesting.git --path deploy --dest-server https://api.trainingk8s.xyz(Replace-with-ur-cluster) --dest-namespace default

argocd app create votingapp --repo https://github.com/mavrick202/argocdtesting.git --path votingapp --dest-server https://api.trainingk8s.xyz --dest-namespace default

1. argocd app get rollingupdate (or) Can be done fom the GUI
2. argocd app sync rollingupdate (or) Can be done fom the GUI



Using Azure Devops:

1. Create SP using Kubeconfig.
2. Use the repo with a folder deploy and inside create deployment.yaml
3. Kubectl apply -f ./\_azureb13terraformaws/deploy/