|  |  |
| --- | --- |
|  |  |
| **Hüseyin Məmmədli** | **DevOps-2** |
| ArgoCD App Of Apps |  |
| **#**kubectl cluster-info  #git clone https://github.com/a1tan/appofappssample.git | Clone the sample App Of Apps implementation |
| **#**kubectl apply -k appofappssample/applications/argoc  #kubectl get pods -n argocd -w | Install the ArgoCD application |
| #kubectl apply -k appofappssample/parentapplication | Apply the parent application definition |
| #kubectl get application -n argocd -owide -w | Check if applications installed and synchronized |
| ArgoCD App Of Apps | ------------------------------------------------------ |
| #kubectl apply -k argoapplicationsets/managementstack/argocdapplications | Apply the applicationset definition |
| #kubectl get application -n argocd -owide -w | Check if applications installed and synchronized |
| ArgoCD Autopilot | ----------------------------------- |
| #kubectl cluster-info  #VERSION=$(curl --silent "https://api.github.com/repos/argoproj-labs/argocd-autopilot/releases/latest" | grep '"tag\_name"' | sed -E 's/.\*"([^"]+)".\*/\1/')  #curl -L --output - https://github.com/argoproj-labs/argocd-autopilot/releases/download/$VERSION/argocd-autopilot-linux-amd64.tar.gz | tar zx  #mv ./argocd-autopilot-\* /usr/local/bin/argocd-autopilot  #mv ./argocd-autopilot-\* /usr/local/bin/argocd-autopilot | Check if kubernetes is up and running |
| #PAT=XXX  #GITHUB\_ACCOUNT=XXX  #REPOSITORY=XXX  #export GIT\_TOKEN=$PAT  #export GIT\_TOKEN=$PAT  #git config --global user.name "test"  #export GIT\_REPO=https://github.com/$GITHUB\_ACCOUNT/$REPOSITORY |  |
| #argocd-autopilot repo bootstrap | create repository and bootstrap ArgoCD |
| #kubectl get pods -n argocd -w  #kubectl get application -n argocd -owide -w | Check if ArgoCD setup completed |
| Bootstrap flux With Bootstrap CLI |  |
| #kubectl cluster-info  #curl -s https://fluxcd.io/install.sh | sudo bash | Install flux cli |
| #kubectl get pods -n flux-system -owide -w  #kubectl get GitRepository -A -owide  #kubectl get Kustomization -A -owide | Check if sources for Kustomization applied |
| Bootstrap Flux With Terraform |  |
| #kubectl cluster-info  #curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add –  #sudo apt-get update && sudo apt-get install terraform | Install terraform cli |
| #git clone <https://github.com/a1tan/fluxterraformgithub.git>  #cd fluxterraformgithub | Download terraform resource to be applied |
| #terraform init  #terraform plan -out=tfplan  #terraform apply tfplan | Terraform init, plan and apply files |
| #kubectl get pods -n flux-system -owide -w  #kubectl get GitRepository -A -owide  #kubectl get Kustomization -A -owide |  |
| Pull Request Ephemeral Environments |  |
| #kubectl cluster-info  #git clone https://github.com/a1tan/ephemeralenvironments.git | Clone the sample App Of Apps implementation to current directory: |
| #kubectl apply -k ephemeralenvironments/managementstack/argocd  #kubectl get pods -n argocd -w | Install the ArgoCD application |
| #kubectl apply -k ephemeralenvironments/managementstack/argocdapplications | Apply the applicationset definition |
| #kubectl get application -n argocd -owide -w | Check if applications installed and synchronized |
| #kubectl port-forward svc/sampleapp-ns-f-ephemeralpullrequest-3 -n sampleapp-ns-f-ephemeralpullrequest-3-namespace 8080:80 & | Port forward service for the namespace environmen |
| #curl http://localhost:8080/Sample | Check the endpoint |
| #kubectl get secret vc-cluster-f-ephemeralpullrequest-3 -n cluster-f-ephemeralpullrequest-3-namespace -ojsonpath={.data.config} | base64 -d > config.yaml  #sed -i 's/cluster-f-ephemeralpullrequest-3.cluster-f-ephemeralpullrequest-3-namespace.svc.cluster.local/localhost:8081/g' config.yaml |  |
| #kubectl port-forward svc/cluster-f-ephemeralpullrequest-3 -n cluster-f-ephemeralpullrequest-3-namespace 8081:443 &  #kubectl port-forward svc/sampleapp-f-ephemeralpullrequest-3 -n sampleapp-f-ephemeralpullrequest-3-namespace --kubeconfig config.yaml 8082:80 &  #curl http://localhost:8082/Sample |  |