CI/CD Framework for Infra Automation

1. Overall Infra Automation setup
   1. Artifacts

All artifacts are in GitEA repo: <https://gitea.rogers.lab/klukashin/cicd>

Concourse pipelines folder: <https://gitea.rogers.lab/klukashin/cicd/src/branch/master/pipelines>

Ansible playbooks folder:

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible>

Values.yaml for ECS folder:

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ECS_Values>

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ECS_XXX_Values>, where XXX is environment(QA1/2/3, DEV1/2/3)

Artifacts for Upgrade folder:

https://gitea.rogers.lab/klukashin/cicd/src/branch/master/Upgrade

* 1. Pre-requisites

The following are pre-requisites to make it work:

1. Concourse is installed on Jumphost. Concourse CLI(FLY) is installed
2. Concourse pipelines are pushed to Concourse using Fly CLI
   1. How it works

Concourse pipelines check out GitEA repo and run relevant Ansible Playbook. Ansible playbook calls relevant APIs(TCA, NSX-T, AVI, NetBox)

1. Pipelines
   1. TKG Cluster creation pipeline

Pipeline name: create-cluster

What it does: Pipeline deploy TKG cluster

Pipeline input in GitEA: <https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/mycluster.j2>

* 1. Deploy AKO pipeline

Pipeline name: deploy-ako

What it does: Pipeline deploy AKO(AVI Kubernetes Operator) Csar on TKG Cluster

Pipeline inputs in GitEA:

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/cnf_resource.json>

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/cnf_instantiate.j2>

https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/values.yaml

* 1. Configure NSXT pipeline

Pipeline name: config-nsxt

What it does: Pipeline adds static route on T1 GW for Cluster VIP and adds SNAT Rules for all workers/masters of the TKG cluster

Pipeline inputs in GitEA:

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/clusterIPs.yaml>

https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/values.yaml

* 1. Deploy ECS Network Service pipeline

Pipeline name: deploy-network-service

What it does: Pipeline deploy full ECS as a network service(includes instantiating of 17 CNFs and tainting/kernel params workflows) on the TKG Cluster

Pipeline inputs:

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/ecsNSParams.yaml> (network service config file)

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/ns_resource.json>

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/ns_instantiate.j2>

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ECS_XXX_Values> (folder for Values.yaml for each component)

* 1. Upgrade ECS network function pipeline

Pipeline name: upgrade-nf

What it does: Pipeline automate Upgrade scenario for CNF. It consists of 4 steps: push new helm chart to repo, push new images to repo, build a new csar file and push it TCA Catalogue, run Upgrade against existing CNF instance.

Pipeline inputs:

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/Upgrade> (folder for new artifacts)

<https://gitea.rogers.lab/klukashin/cicd/src/branch/master/ansible/ecsUpgradeParams.yaml> (Upgrade config file)