**Setting up Airflow for GKE**

This document assumes that you have installed the required software including the following. Also it assumes that the GKE cluster is created using terraform

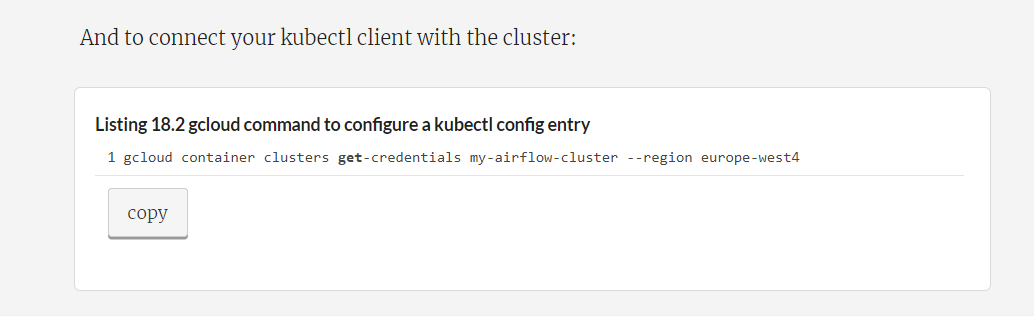
1.Git for windows

2.Kubectl

3.Terraform

4.Google SDK (gcloud)

**First Approach:**





**Second Approach:**

The other way is to bind the existing git Repo and use it



This will give the following output similar to the above

Setting up Airflow installation on the existing cluster options as shown below. The assumption is that the GKE cluster is already created and you have installed the following

C:\gkeclusterwithterraformhelmchart>gcloud container clusters get-credentials airflow-cluster --region us-central1-f

Fetching cluster endpoint and auth data.

kubeconfig entry generated for airflow-cluster.

C:\gkeclusterwithterraformhelmchart>helm install myreleaseforairflow bitnami/airflow --set airflow.cloneDagFilesFromGit.enabled=true --set airflow.cloneDagFilesFromGit.repository=https://github.com/bitnami/tutorials/tree/master/airflow-dag-examples --set airflow.cloneDagFilesFromGit.branch=master --set airflow.baseUrl=http://127.0.0.1:8080

coalesce.go:203: warning: destination for tls is a table. Ignoring non-table value

NAME: myreleaseforairflow

LAST DEPLOYED: Wed Sep 29 17:45:55 2021

NAMESPACE: default

STATUS: deployed

REVISION: 1

TEST SUITE: None

NOTES:

1. Get the Airflow URL by running:

echo URL : http://127.0.0.1:8080

kubectl port-forward --namespace default svc/myreleaseforairflow 8080:8080

2. Get your Airflow login credentials by running:

export AIRFLOW\_PASSWORD=$(kubectl get secret --namespace "default" myreleaseforairflow -o jsonpath="{.data.airflow-password}" | base64 --decode)

echo User: user

echo Password: $AIRFLOW\_PASSWORD

**Third Approach:**

**Deploying Airflow on Google Kubernetes Engine**

Install Apache Airflow framework (Celery Executer, Cloud Sql database, Redis) to GKE using Helm and Terraform. For details and examples visit [my blog](https://act-labs.github.io/posts/airflow-gke/)

**Installation**

1. Create cloud infrastructure: Cloud Sql instance, GKE cluster, SSD disk. Run the following command from the root folder

terraform init

terraform apply

1. Create service accounts and permissions, enable APIs and configure Workload Identity:

cd k8s

terraform init

terraform apply

cd ..

1. Create Docker image and publish it to Google Container Registry. Docker image is created in two steps. First, we build image with libraries and necessary utilities. Secondly we build image containing which adds sample DAGs:

# add libraries and third party packages

cd src

docker build -t airflow-python .

# add dags

cd ..

docker build -t airflow-gke:latest .

# push to Google Container repository

docker tag airflow-gke gcr.io/google\_project\_id/airflow-gke:latest

docker push gcr.io/google\_project\_id/airflow-gke

1. Finally, deploy your application using Helm:

helm upgrade --install airflow . --set projectId=google\_project\_id