**Setting up Prometheus on Kubernetes**

* <https://cloud.google.com/architecture/monitoring-apps-running-on-multiple-gke-clusters-using-prometheus-and-stackdriver#configure_cloud_identity_and_access_management_(iam)> for the article for setting up prometheus on gke
* Steps to execute from windows terminal /powershell

git clone https://github.com/GoogleCloudPlatform/prometheus-stackdriver-gke

cd prometheus-stackdriver-gke

gcloud iam service-accounts create prometheus --display-name prometheus-service-account

$PROJECT\_ID=$(gcloud info --format='value(config.project)')

$PROMETHEUS\_SA\_EMAIL=$(gcloud iam service-accounts list --filter="displayName:prometheus-service-account" --format='value(email)')

gcloud projects add-iam-policy-binding $PROJECT\_ID --role roles/monitoring.metricWriter --member serviceAccount:$PROMETHEUS\_SA\_EMAIL

gcloud iam service-accounts keys create prometheus-service-account.json --iam-account $PROMETHEUS\_SA\_EMAIL

gcloud container clusters create hello-cluster --num-nodes=1

gcloud container clusters get-credentials hello-cluster

kubectl create namespace prometheus

* Make changes in gke-prometheus-deployment.yaml

#Copyright 2019 Google LLC

#Licensed under the Apache License, Version 2.0 (the "License");

#you may not use this file except in compliance with the License.

#You may obtain a copy of the License at

#https://www.apache.org/licenses/LICENSE-2.0

#Unless required by applicable law or agreed to in writing, software

#distributed under the License is distributed on an "AS IS" BASIS,

#WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

#See the License for the specific language governing permissions and

#limitations under the License.

apiVersion: apps/v1

kind: Deployment

metadata:

name: prometheus-deployment

namespace: prometheus

labels:

app: prometheus-server

spec:

replicas: 1

selector:

matchLabels:

app: prometheus-server

template:

metadata:

labels:

app: prometheus-server

spec:

containers:

- name: prometheus

image: prom/prometheus:v2.19.3

args:

- "--config.file=/etc/prometheus/prometheus.yml"

- "--storage.tsdb.path=/prometheus/"

ports:

- containerPort: 9090

volumeMounts:

- name: prometheus-config-volume

mountPath: /etc/prometheus/

- name: prometheus-storage-volume

mountPath: /prometheus/

- name: sidecar

image: gcr.io/stackdriver-prometheus/stackdriver-prometheus-sidecar:0.8.2

imagePullPolicy: Always

args:

- --stackdriver.project-id=expertkubernetes

- --prometheus.wal-directory=/prometheus/wal

- --stackdriver.kubernetes.location=us-central1-a

- --stackdriver.kubernetes.cluster-name=hello-cluster

#- \"--stackdriver.generic.location=${GCP\_LOCATION}\"

#- \"--stackdriver.generic.namespace=${KUBE\_CLUSTER}\"

ports:

- name: sidecar

containerPort: 9091

volumeMounts:

- name: prometheus-storage-volume

mountPath: /prometheus

volumes:

- name: prometheus-config-volume

configMap:

defaultMode: 420

name: prometheus-server-conf

- name: prometheus-storage-volume

emptyDir: {}

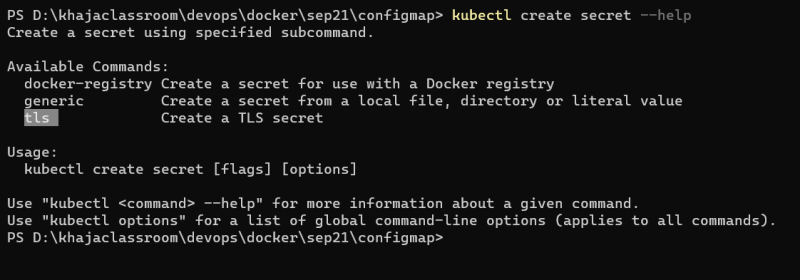
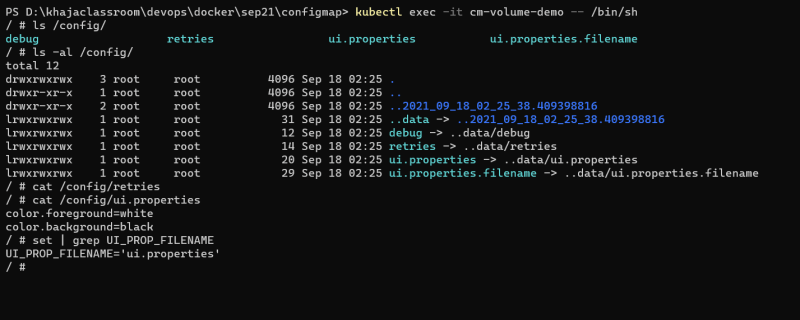
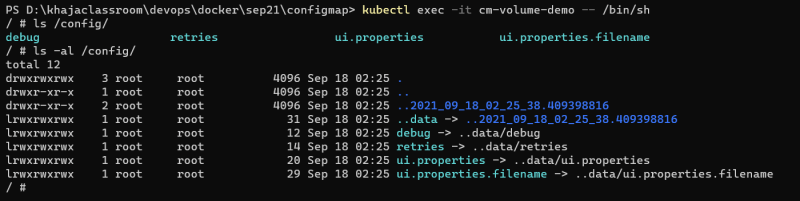
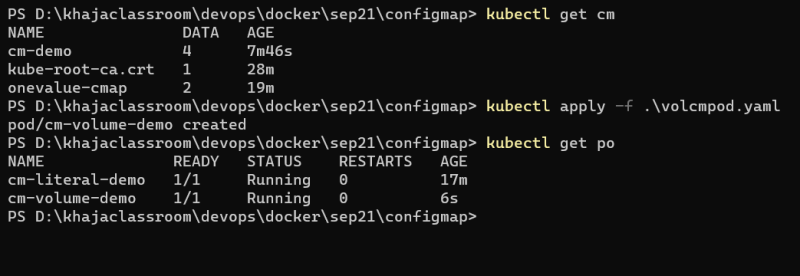
* Now apply

kubectl apply -f .\gke-prometheus-deployment.yaml

kubectl get pods -n prometheus

$PROMETHEUS\_POD\_GKE=$(kubectl get pods --namespace prometheus -l "app=prometheus-server" -o jsonpath="{.items[0].metadata.name}" )

kubectl port-forward --namespace prometheus $PROMETHEUS\_POD\_GKE 9090:9090

* Now work with Prometheus Expression Browser 
* Now lets create a postgres deployment into the k8s cluster

helm repo add bitnami https://charts.bitnami.com/bitnami

helm repo update

helm install gke bitnami/postgresql --set metrics.enabled=true --set postgresqlDatabase=pro