export INSTANCE\_NAME=nucleus-jumphost-371

export ZONE=us-west4-b

export REGION=us-west4

export PORT=8083

export FIREWALL\_NAME=grant-tcp-rule-901

gcloud compute instances create $INSTANCE\_NAME \

--network nucleus-vpc \

--zone $ZONE \

--machine-type f1-micro \

--image-family debian-10 \

--image-project debian-cloud

gcloud container clusters create nucleus-backend \

--num-nodes 1 \

--network nucleus-vpc \

--zone $ZONE

gcloud container clusters get-credentials nucleus-backend \

--zone $ZONE

kubectl create deployment hello-server \

--image=gcr.io/google-samples/hello-app:2.0

kubectl expose deployment hello-server \

--type=LoadBalancer \

--port $PORT

cat << EOF > startup.sh

#! /bin/bash

apt-get update

apt-get install -y nginx

service nginx start

sed -i -- 's/nginx/Google Cloud Platform - '"\$HOSTNAME"'/' /var/www/html/index.nginx-debian.html

EOF

gcloud compute instance-templates create web-server-template \

--metadata-from-file startup-script=startup.sh \

--network nucleus-vpc \

--machine-type g1-small \

--region $ZONE

gcloud compute target-pools create nginx-pool --region=$REGION

gcloud compute instance-groups managed create web-server-group \

--base-instance-name web-server \

--size 2 \

--template web-server-template \

--region $REGION

gcloud compute firewall-rules create $FIREWALL\_NAME \

--allow tcp:80 \

--network nucleus-vpc

gcloud compute http-health-checks create http-basic-check

gcloud compute instance-groups managed \

set-named-ports web-server-group \

--named-ports http:80 \

--region $REGION

gcloud compute backend-services create web-server-backend \

--protocol HTTP \

--http-health-checks http-basic-check \

--global

gcloud compute backend-services add-backend web-server-backend \

--instance-group web-server-group \

--instance-group-region $REGION \

--global

gcloud compute url-maps create web-server-map \

--default-service web-server-backend

gcloud compute target-http-proxies create http-lb-proxy \

--url-map web-server-map

gcloud compute forwarding-rules create http-content-rule \

--global \

--target-http-proxy http-lb-proxy \

--ports 80

gcloud compute forwarding-rules create $FIREWALL\_NAME \

--global \

--target-http-proxy http-lb-proxy \

--ports 80

gcloud compute forwarding-rules list