

## 2.Kubernetes task

### Step A

The Kubernetes Cluster was Created on Google Cloud (GKE) environment, the Cluster contains 3 nodes, the cluster with created following the below steps:

- Use GUI of Google cloud to create the cluster or by using Cloud shell commands (**restrict access to certain IPs option is added in the below command**):

```
gcloud beta container --project "ardent-kite-354709" clusters create "ocb-task" --zone "us-central1-c" --no-enable-basic-auth --cluster-version "1.22.8-gke.202" --release-channel "regular" --machine-type "e2-medium" --image-type "COS_CONTAINERD" --disk-type "pd-standard" --disk-size "100" --metadata disable-legacy-endpoints=true --scopes "https://www.googleapis.com/auth/devstorage.read_only","https://www.googleapis.com/auth/logging.write","https://www.googleapis.com/auth/monitoring","https://www.googleapis.com/auth/servicecontrol","https://www.googleapis.com/auth/service.management.readonly","https://www.googleapis.com/auth/trace.append" --max-pods-per-node "110" --num-nodes "3" --logging=SYSTEM,WORKLOAD --monitoring=SYSTEM --enable-ip-alias --network "projects/ardent-kite-354709/global/networks/default" --subnetwork "projects/ardent-kite-354709/regions/us-central1/subnetworks/default" --no-enable-intra-node-visibility --default-max-pods-per-node "110" --no-enable-master-authorized-networks --addons HorizontalPodAutoscaling,HttpLoadBalancing,GcePersistentDiskCsiDriver --enable-autoupgrade --enable-autorepair --max-surge-upgrade 1 --max-unavailable-upgrade 0 --enable-shielded-nodes --node-locations "us-central1-c" --master-ipv4-cidr=Needed IPs
```

- Connect to the cluster from Cloud Shell:

```
gcloud container clusters get-credentials ocb-task --zone us-central1-c --project ardent-kite-354709
```

### Step B:

- Deploy Nginx Ingress:

First, user needs to have cluster-admin permissions on the cluster. This can be done with the following command:

```
kubectl create clusterrolebinding cluster-admin-binding \
--clusterrole cluster-admin \
--user $(gcloud config get-value account)
```

- Then, install Nginx ingress:

```
kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/controller-v1.2.1/deploy/static/provider/cloud/deploy.yaml
```

- Create Deployment, service, and ingress files as attached here , then deploy them using the below command

Kubectl apply -f "filename"



juice-shop-deployment.yaml



juice-shop-service.yaml



juice-shop-ingress.yaml

- The application is accessible through this link: [OWASP Juice Shop](#)