Deploying Google Nginx Controller with RBAC

This doc will help you out in installing Google Nginx ingress controller with RBAC alsong with default http backend. This Deployment is only for AWS cause it will launch ELB for exposing the services outside.

- Create a namespace

This will create a namespace called "`ingress-nginx`"

```
curl
https://raw.githubusercontent.com/kubernetes/ingress-nginx/master/deploy/namespace.yaml \
    | kubectl apply -f -
```

- Create a RBAC rule

This will create a service account called "`nginx-ingress-serviceaccount`"

```
curl
https://raw.githubusercontent.com/kubernetes/ingress-nginx/master/deploy/without-rbac.yaml \
    | kubectl apply -f -
```

- Create default-backend

This will create default-backend pod

```
curl
https://raw.githubusercontent.com/kubernetes/ingress-nginx/master/deploy/default-backend.yaml \
    | kubectl apply -f -
```

- Create Configmap for later modification in main nginx config file

```
curl
https://raw.githubusercontent.com/kubernetes/ingress-nginx/master/deploy/configmap.yaml \
    | kubectl apply -f -
```

- Create Google Nginx ingress controller

This will create nginx controller pod.

```
curl
https://raw.githubusercontent.com/kubernetes/ingress-nginx/master/deploy/with-rbac.yaml \
    | kubectl apply -f -
```

- Create Service with Type Loadbalancer

This will create a service and parallely launch an ELB.

```
kubectl apply -f
https://raw.githubusercontent.com/kubernetes/ingress-nginx/master/deploy/provide
r/aws/service-l4.yaml
```

- Now modify your configmap and add one data set.

```
# vi configmap.yml
kind: ConfigMap
apiVersion: v1
metadata:
  name: nginx-configuration
  namespace: ingress-nginx
  labels:
    app: ingress-nginx
data:
  use-proxy-protocol: "true"
```

Post that hit `kubectly apply -f configmap.yml`

At this point we have deployed nginx controller. If you will hit the ELB DNS then you will get

“default backend - 404” as output

Note: You won’t get Nginx Default page