Exercise 9.2: Configure a NodePort

In a previous exercise we deployed a `LoadBalancer` which deployed a `ClusterIP` and`NodePort` automatically. In this exercise we will deploy a `NodePort`. While you can access a container from within the cluster, one can use a `NodePort` to NAT traffic from outside the cluster. One reason to deploy a `NodePort` instead, is that a `LoadBalancer` is also a load balancer resource from cloud providers like GKE and AWS.

1. In a previous step we were able to view the **nginx** page using the internal Pod IP address. Now expose the deployment using the `--type=NodePort`. We will also give it an easy to remember name and place it in the `accounting` namespace. We could pass the port as well, which could help with opening ports in the firewall.

```
student@cp:~$ kubectl -n accounting expose deployment nginx-one --type=NodePort --name=service-lab
service/service-lab exposed
```

2. View the details of the services in the `accounting` namespace. We are looking for the autogenerated port.

```
student@cp:~$ kubectl -n accounting describe services
....
NodePort:                 <unset>  32103/TCP
....
```

3. Locate the exterior facing hostname or IP address of the cluster. The lab assumes use of GCP nodes, which we access via a `FloatingIP`, we will first check the internal only public IP address. Look for the Kubernetes cp URL. Whichever way you access check access using both the internal and possible external IP address

```
student@cp:~$ kubectl cluster-info
Kubernetes control plane is running at https://k8scp:6443
CoreDNS is running at https://k8scp:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
```

4. Test access to the **nginx** web server using the combination of cp URL and `NodePort`.

```
student@cp:~$ curl http://k8scp:32103
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
```

5. Using the browser on your local system, use the public IP address you use to SSH into your node and the port. You should still see the **nginx** default page. You may be able to use **curl** to locate your public IP address.

```
student@cp:~$ curl ifconfig.io
104.198.192.84
```