

In order to achieve redundancy for your Kubernetes cluster, you will need to load balance usage of the Kubernetes API across multiple control nodes. In this lesson, you will learn how to create a simple nginx server to perform this balancing. After completing this lesson, you will be able to interact with both control nodes of your kubernetes cluster using the nginx load balancer.

Here are the commands you can use to set up the nginx load balancer. Run these on the server that you have designated as your load balancer server:

```
sudo apt-get install -y nginx
sudo systemctl enable nginx
sudo mkdir -p /etc/nginx/tcpconf.d
sudo vi /etc/nginx/nginx.conf
```

Add the following to the end of nginx.conf:

```
include /etc/nginx/tcpconf.d/*;
```

Set up some environment variables for the lead balancer config file:

```
CONTROLLER0_IP=<controller 0 private ip>
CONTROLLER1_IP=<controller 1 private ip>
```

Create the load balancer nginx config file:

```
cat << EOF | sudo tee /etc/nginx/tcpconf.d/kubernetes.conf
stream {
    upstream kubernetes {
        server $CONTROLLER0_IP:6443;
        server $CONTROLLER1_IP:6443;
    }

    server {
        listen 6443;
        listen 443;
        proxy_pass kubernetes;
    }
}
EOF
```

Reload the nginx configuration:

```
sudo nginx -s reload
```

You can verify that the load balancer is working like so:

```
curl -k https://localhost:6443/version
```

You should get back some json containing version information for your Kubernetes cluster.