**Create a Nginx LB for React Application(s)**

**Revision History**:

|  |  |  |  |
| --- | --- | --- | --- |
| Author | Description | Date | Version |
| Robert Chin | Initial Document Creation | 3/24/20 | V0.1 |

**Prerequisites**:

1. 2 x Existing Nginx web servers VMs hosting React Applications
2. A new VM to act as Nginx LB

**Procedures**:

1. Install Nginx

apt-get install nginx

cd /etc/nginx

1. Navigate to Nginx web server via browser to see the default page

http://12.132.231.240

1. Reconfigure the Nginx web server for Load Balancing (By default, Round Robin)

Reference: [**https://docs.nginx.com/nginx/admin-guide/load-balancer/http-load-balancer/**](https://docs.nginx.com/nginx/admin-guide/load-balancer/http-load-balancer/)

cp /etc/nginx/sites-available/default /etc/nginx/sites-available/default.old

vi /etc/nginx/sites-available/default

Replace with this below:

upstream www {

server 192.168.1.6:80; #react server 1

server 192.168.1.3:80; #react server 2

}

server {

listen 80;

server\_name 12.132.231.240;

location / {

proxy\_pass http://www;

}

}

1. Bounce the Nginx service

service nginx restart

1. Enable Nginx with SSL certificates (i.e. LB SSL Termination)

[**https://www.digitalocean.com/community/tutorials/how-to-set-up-nginx-load-balancing-with-ssl-termination**](https://www.digitalocean.com/community/tutorials/how-to-set-up-nginx-load-balancing-with-ssl-termination)

* 1. Create SSL certificate directory

mkdir -p /etc/nginx/ssl/example

cd /etc/nginx/ssl/example

* 1. Create Private Key

openssl genrsa -des3 -out server.key 2048

* 1. Remove its passphrase (so you don’t have to enter password everytime server is started)

openssl rsa -in server.key -out server.key

* 1. Create a CSR (Certificate Signing request)

openssl req -new -key server.key -out server.csr

* 1. Create Self-Signed Certificate

openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt

* 1. Edit nginx configuration (to enable SSL)

NOTE: ssl\_trusted\_certificate field is for ca-certs.pem which is a bundle of certificates (i.e. Leaf, Intermediate, Root)

root@bob-reverse-proxy:/etc/nginx/sites-enabled# vi default

upstream www {

server 192.168.1.6:443; #react server 1

server 192.168.1.3:443; #react server 2

}

server {

listen 80;

listen 443;

ssl on;

ssl\_certificate /etc/nginx/ssl/example/server.crt;

ssl\_certificate\_key /etc/nginx/ssl/example/server.key;

# ssl\_trusted\_certificate /etc/nginx/ssl/example/ca-certs.pem;

server\_name 12.132.231.240;

location /server/1 {

proxy\_pass https://192.168.1.6:443;

}

location /server/2 {

proxy\_pass https://192.168.1.6:443;

}

location / {

proxy\_pass https://www;

}

}