6. High-Available Reverse Proxy

Introduction

Please ensure that you have already set up web servers in BOTH web01 and web02 as haproxy is dependent on them.

**Objective:**

At the end of this guide, you will learn how to configure ha-prx01 and ha-prx02 as high-available reverse proxies using haproxy and keepalived.

**Login**

The login credential for all server and client machines:

Username: root / user

Password: Skill39@Lyon

**Network Topology**

|  |
| --- |
| This will be the network topology that will be referenced for setting up the infrastructure. |
|  |

We will be using keepalived and haproxy for high-available reverse proxy.

**Install haproxy and keepalived on both ha-prx01 and ha-prx02:**

apt install keepalived haproxy -y

## Keepalived

**Do the following on ha-prx01**

apt install keepalived -y

cp /etc/keepalived/keepalived.conf.sample /etc/keepalived/keepalived.conf

vim /etc/keepalived/keepalived.conf

systemctl restart keepalived

**Copy the file to ha-prx02. You can also redo the above steps in ha-prx02 but copying will be faster.**

scp /etc/keepalived/keepalived.conf

**Do note that changes need to be made after copying the file to ha-prx02.**

In ha-prx02, MASTER needs to be changed to BACKUP as shown below:

**Restart the service in both ha-prx01 and ha-prx02.**

systemctl restart keepalived -y

## Haproxy

**Do the following in ha-prx01**

scp root@10.1.10.10:/etc/ssl/CA/certs/web.crt /etc/haproxy

scp /etc/haproxy/

scp /etc/haproxy

cat /etc/haproxy/web.key >> /etc/haproxy/web.crt

cat /etc/haproxy/ca.crt >> /etc/haproxy/web.crt

vim /etc/haproxy/haproxy.cfg

**Copy the file to ha-prx02**

scp /etc/haproxy/haproxy.cfg

**After copying the file to ha-prx02, change the header.**

**Restart the service on both ha-prx01 and ha-prx02**

systemctl restart haproxy

## NAT

After configuring haproxy and keepalived, we configure NAT on the firewall to forward HTTP, HTTPS and DNS to 10.1.20.20 (virtual IP address which is ha-prx01)

**Do the following in fw**

vim /etc/nftables.conf

table ip nat {

chain postrouting {

type nat hook postrouting priority srcnat;

#interface with public IP

iif “ens33” tcp dport 80 dnat to 10.1.20.20:80;

iif “ens33” tcp dport 443 dnat to 10.1.20.20:443;

iif “ens33” tcp dport 53 dnat to 10.1.20.20:53;

iif “ens33” udp dport 53 dnat to 10.1.20.20:53;

}

chain prerouting {

type nat hook prerouting priority dstnat;

oif “ens33” masquerade;

}

}

systemctl restart nftables.service

|  |
| --- |
| Test |
|  |

|  |
| --- |
| Access the web to test if the reverse proxy and port forwarding works |
|  |