keepalived 구축

구축 목적

: Proxy 서버에 문제가 생겼을때 이중화를 통하여, BACKUP서버로 바로 트래픽을 넘겨 받아 정상적인 서버 운영을 하기 위하여 (failover)

IP

```
VIP : 172.17.124.240 - Keepalived
HA 1 : 172.17.124.241
HA 2 : 172.17.124.242
WEB 1 : 172.17.124.243
WEB 2 : 172.17.124.244
```

keepalived 구축

```
# yum -y install keepalived
# yum -y install psmisc ( killall 명령어를 위해 )

# vi /etc/selinux/config
SELINUX=disabled

# systemctl stop firewalld

# vi /etc/sysctl.conf
net.ipv4.ip_forward = 1
net.ipv4.ip_nonlocal_bind = 1

# sysctl -p

# cp keepalived.conf keepalived.conf.bak
# vi /etc/keepalived/keepalived.conf

<MASTER>
----------------------------------------------------------
! Configuration File for keepalived

global_defs {
    router_id HAproxy1
    script_user root
    enable_script_security
}

vrrp_script chk_haproxy {
    script "killall -0 haproxy"
```

```
        interval 2
        weight 2
        fall 2
        rise 2
    }

vrrp_instance VI_1 {
    state MASTER
    interface eth0
    virtual_router_id 51
    priority 101
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        172.17.124.240
    }
    track_script {
        chk_haproxy
    }
}
----------------------------------------------------------

<BACKUP>
----------------------------------------------------------
! Configuration File for keepalived

global_defs {
    router_id HAProxy2
    script_user root
    enable_script_security
}

vrrp_script chk_haproxy {
    #script "systemctl is-active haproxy"
    script "killall -0 haproxy"
    interval 2
    weight 2
    rise 2
    fall 2
}

vrrp_instance VI_1 {
#    track_interface {
#        eth0
#    }
    state BACKUP
    interface eth0
    virtual_router_id 51
    priority 100
    advert_int 1
    authentication {
```

```
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        172.17.124.240
    }
    track_script {
        chk_haproxy
    }
}
---------------------------------------------------------
# systemctl start keepalived
# systemctl enable keepalived

* 오류 발생시 참고
# vi /var/log/messages
# systemctl status keepalived
```

TEST

- ip a 명령어를 통해 HAproxy 서버와 VIP가 잘 들어와있는지 확인

```
# ip a

<MASTER>
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
    link/ether 50:6b:8d:95:24:5d brd ff:ff:ff:ff:ff:ff
    inet 172.17.124.241/24 brd 172.17.124.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet 172.17.124.240/32 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::526b:8dff:fe95:245d/64 scope link
       valid_lft forever preferred_lft forever

<BACKUP>
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
    link/ether 50:6b:8d:ed:eb:88 brd ff:ff:ff:ff:ff:ff
    inet 172.17.124.242/24 brd 172.17.124.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::526b:8dff:feed:eb88/64 scope link
       valid_lft forever preferred_lft forever
```

- 서비스를 내렸을때 정상적으로 BACKUP서버 쪽으로 Failover 되는지 확인 (HAproxy)

```
# systemctl stop keepallived
```

```
<MASTER>
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
    link/ether 50:6b:8d:95:24:5d brd ff:ff:ff:ff:ff:ff
    inet 172.17.124.241/24 brd 172.17.124.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::526b:8dff:fe95:245d/64 scope link
       valid_lft forever preferred_lft forever

<BACKUP>
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
    link/ether 50:6b:8d:ed:eb:88 brd ff:ff:ff:ff:ff:ff
    inet 172.17.124.242/24 brd 172.17.124.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet 172.17.124.240/32 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::526b:8dff:feed:eb88/64 scope link
       valid_lft forever preferred_lft forever
```

- 위와 동일하게 확인 (keepalived)

```
# systemctl stop haproxy
: 위와 동일하게 확인
```

- WEB에서 Proxy서버가 정상적으로 작동되는지, 내렸을때도 VIP로 접속했을때 잘 올라오는지 확인

★ 정상적으로 작동된다면 Keepalive 구축 완료 ! ★

참조

- https://www.nakjunizm.com/2020/02/20/haproxy/
- https://springboot.cloud/24