

HAProxy 설치 & 설정

- 시나리오
- 설치
- 설정
- 참고

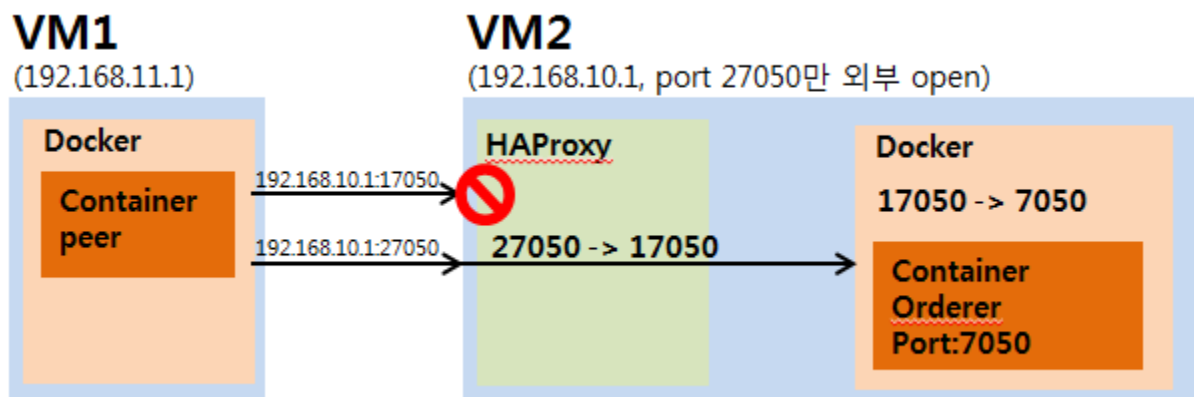


haproxy는 grpc를 1.9.2 버전 이상부터 지원

- 참고: <https://www.haproxy.com/blog/haproxy-1-9-2-adds-grpc-support/>

시나리오

시나리오.pptx



설치

- apt install

```
$ sudo apt-get install software-properties-common
```

- repository 추가

```
$ sudo add-apt-repository ppa:vbernat/haproxy-1.9
```

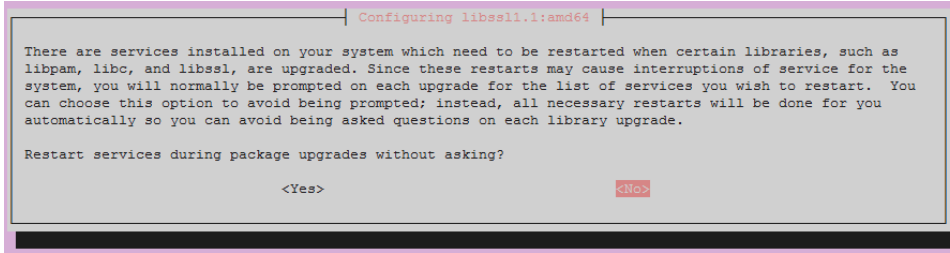
- apt update

```
$ sudo apt-get update
```

- haproxy 설치

```
$ sudo apt-get install haproxy=1.9.*
```

- 다음과 같은 알람 뜨는데? 어떻게 함?



haproxy apt install 과정에서 아래와 같은 에러 발생할 경우

```
The following packages have unmet dependencies:  
haproxy : Depends: libpcre2-8-0 (>= 10.31) but it is not installable  
E: Unable to correct problems, you have held broken packages.
```

- libpcre2-8-0 download
 - https://ubuntu.pkgs.org/18.04/ubuntu-universe-amd64/libpcre2-8-0_10.31-2_amd64.deb.html
- 다운로드 받은 파일 server로 전송(filezilla등 이용)
- libpcre2-8-0 설치

```
$ sudo dpkg -i libpcre2-8-0_10.31-2_amd64.deb
```

- 다시 haproxy 설치

```
$ sudo apt-get install haproxy=1.9.*
```

설정

- 설정 파일 open

```
$ sudo vi /etc/haproxy/haproxy.cfg
```

haproxy.cf

```
global
    log /dev/log      local0
    log /dev/log      local1 notice
    chroot /var/lib/haproxy
    stats socket /run/haproxy/admin.sock mode 660 level admin expose-fd listeners
    stats timeout 30s
    user haproxy
    group haproxy
    daemon

    # Default SSL material locations
    ca-base /etc/ssl/certs
    crt-base /etc/ssl/private

    # Default ciphers to use on SSL-enabled listening sockets.
    # For more information, see ciphers(1SSL). This list is from:
    # https://hynek.me/articles/hardening-your-web-servers-ssl-ciphers/
    # An alternative list with additional directives can be obtained from
    # https://mozilla.github.io/server-side-tls/ssl-config-generator/?server=haproxy
    ssl-default-bind-ciphers ECDH+AESGCM:DH+AESGCM:ECDH+AES256:DH+AES256:ECDH+AES128:DH+AES:
RSA+AESGCM:RSA+AES:!aNULL:!MD5:!DSS
    ssl-default-bind-options no-ssl3

defaults
    log          global
    mode         http
    option        httplog
    option        dontlognull
    timeout connect 5000
    timeout client  50000
    timeout server  50000
    errorfile 400 /etc/haproxy/errors/400.http
    errorfile 403 /etc/haproxy/errors/403.http
    errorfile 408 /etc/haproxy/errors/408.http
```

- 하단에 아래 내용 추가 (balance 설정은 상황에 따라 필요 없음)

haproxy.cf

```
# -----
# 7050      17050
# port 17050 docker-compose orderer container 7050(orderer coninaer  )
# host(vm) 17050 Host 7050  .
# host(vm) 7050
# 7050 host -> haproxy 7050 localhost:17050
# -> host 17050 port orderer.example.com 7050
# -----
listen orderer-test
    mode tcp
    bind *:27050
    balance roundrobin
    server orderer localhost:17050 check
```



TLS를 사용하는 경우 mode tcp를 사용해야함

- haproxy 재시작

```
$ sudo systemctl restart haproxy
```

- haproxy 상태 확인

```
$ sudo systemctl status haproxy

haproxy.service - HAProxy Load Balancer
  Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; vendor preset: enabled)
  Active: active (running) since Fri 2019-08-02 06:09:28 UTC; 1s ago
    Docs: man:haproxy(1)
           file:/usr/share/doc/haproxy/configuration.txt.gz
  Process: 4741 ExecStartPre=/usr/sbin/haproxy -f $CONFIG -c -q $EXTRA_OPTS (code=exited, status=0/SUCCESS)
 Main PID: 4751 (haproxy)
   Tasks: 2 (limit: 1109)
  CGroup: /system.slice/haproxy.service
          4751 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid
          4753 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid

Aug 02 06:09:28 edu systemd[1]: Starting HAProxy Load Balancer...
Aug 02 06:09:28 edu haproxy[4751]: [WARNING] 213/060928 (4751) : parsing [/etc/haproxy/haproxy.cfg:26] :
'option ht
Aug 02 06:09:28 edu haproxy[4751]: Proxy orderer-test started.
Aug 02 06:09:28 edu haproxy[4751]: Proxy orderer-test started.
Aug 02 06:09:28 edu haproxy[4751]: [NOTICE] 213/060928 (4751) : New worker #1 (4753) forked
Aug 02 06:09:28 edu systemd[1]: Started HAProxy Load Balancer.
lines 1-18/18 (END)
```

- 외부에서 HAProxy를 통해서 내부 container로 채널 create 명령어 날려보자(이전까지의 과정은 생략: ex. 인증서 설정, channel artifacts 생성, container up 등)

```
$ docker exec -e "CORE_PEER_MSPCONFIGPATH=/var/hyperledger/users/Admin@org1.example.com/msp" peer0.org1.example.com peer channel create -o orderer.example.com:27050 -c channelall -f /var/hyperledger/configs/channelall.tx
```

- 블록 생성되는지 확인(or 통신)

참고

- haproxy 설치 : <https://haproxy.debian.net/#?distribution=Ubuntu&release=bionic&version=1.9>