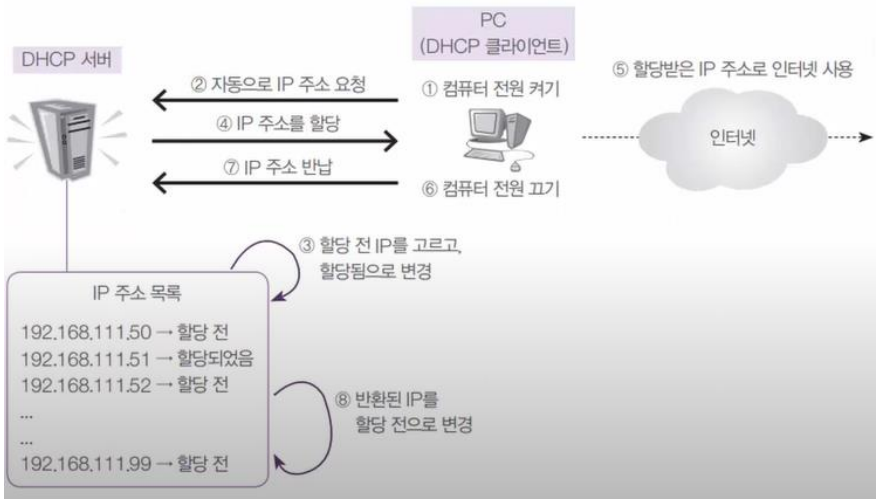
- Ch_1 _ DHCP 서버 구축
- Ch_2 _ 프록시 서버 구축
- Ch_3 _ 방화벽 컴퓨터 개요와 실습 환경
- Ch_4 _ 방화벽 컴퓨터 구축
- Ch_5 _ PXE 서버 구축
- Ch_6 _ 도커 개념정리 및 실습 구축

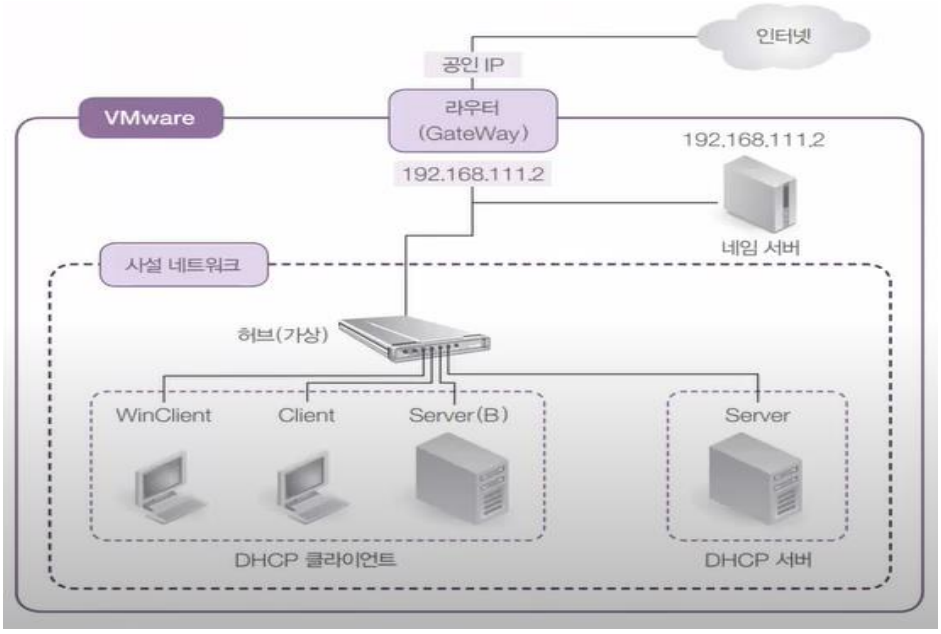
- DHCP 개념
  - DHCP(Dynamic Host Configuration Protocol) 서버가 하는 역할은 자신의 네트워크 안에 있는 클라이언트 컴퓨터가 부팅될 때 자동으로 IP 주소, 서브넷 마스크, 게이트웨이 주소, DNS 서버 주소를 할당해 주는 것임
  - 일반 사용자는 IP 에 관련된 어려운 정보를 알지 못해도, 인터넷을 사용하는 데는 더 이상 아무런 문제가 없어짐
  - DHCP 서버의 가장 큰 장점은 관리하기 편하고 이용자가 편하다는 것
  - 또한 한정된 IP 주소를 가지고 더 많은 IP 주소가 있는 것처럼 활용할 수 있음. 즉, 적은 개수의 IP 주소로 여러 명의 사용자가 사용할 수 있다는 의미
  - DHCP 서버의 작동원리



➢ DHCP 클라이언트로 설정 방법

➔ 우분투를 텍스트 모드에서 DHCP 클라이언트로 설정하려면 /etc/netplan/*.yaml 파일의 'dhcp4:no'부분을 파일의 'dhcp4:true'로 수정하면 된다

➢ VMware 내부에서 구현할 DHCP 서버 구성도

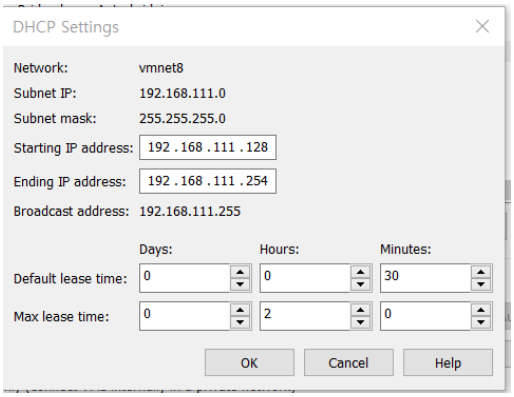


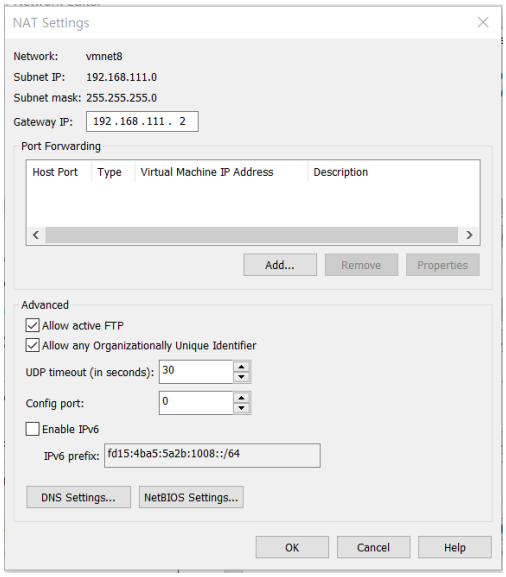
➔ VMware 가 제공하는 DHCP 서버의 기능은 중지시켜야 한다

- DHCP 서버 구축

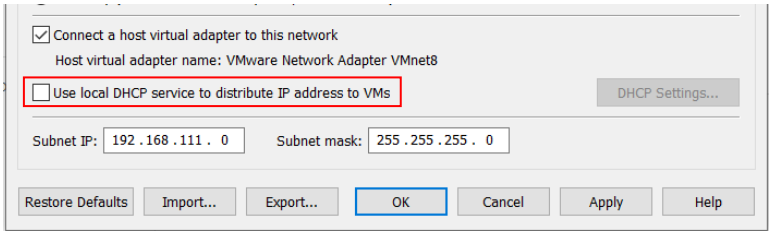
➢ Vmware pro 에서 서버 4 대 초기화 (스냅샷)

➢ server > Edit > Virtual Network Editor... > 우측 하단 Change Settings



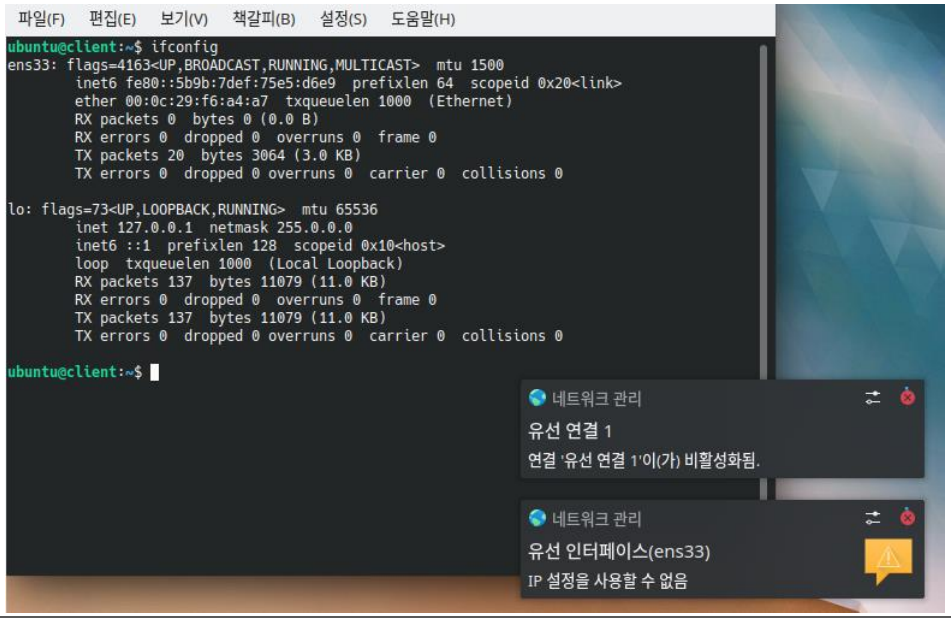


➔ DHCP Setttings 와 NAT Settings 확인 가능

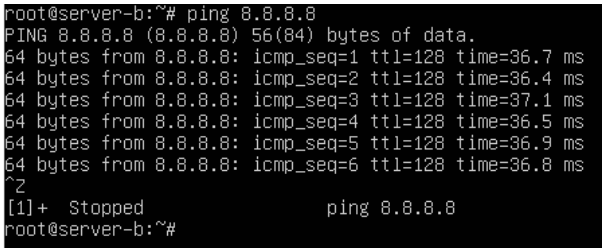


➔ 항목 체크를 풀고 Apply > ok // 더 이상 DHCP 서버를 사용하지 않음

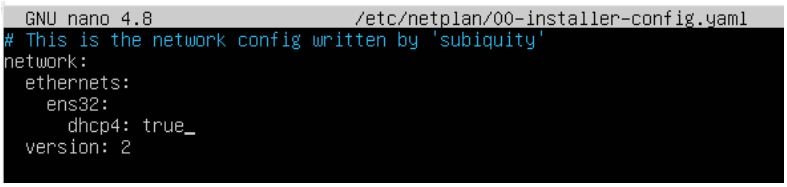
➢ server 와 server(b)는 ip 를 직접 할당하였기 때문에 지장이 없지만 client 와 Winclient 는 자동 ip 이였기 때문에 인터넷 사용 불가



➔ Client 를 ifconfig 로 확인해 본 결과 // 인터넷 비활성화

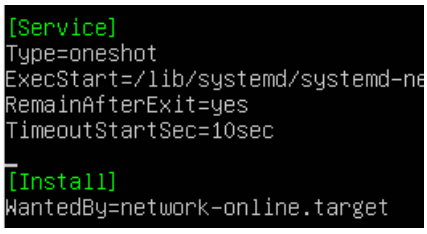


➔ server(b) ping 8.8.8.8 정상 작동 확인



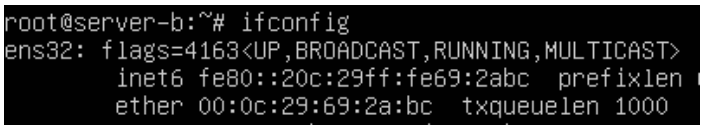
➔ /etc/netplan/00-installer-config.yaml 파일 수정

- dhcp4:no 를 true 로 변경하고 하단 addresses 등 ip 내용 삭제



➔ /etc/systemd/system/network-onlien.target.wants/systemd-networkd-wait-onlien.servie 파일 수정

- 재부팅 후 확인하려는데 이 재부팅이 오래걸린다 (계속해서 ip 를 찾기 때문) 따라서 이 재부팅 시 ip 찾는 시간을 제한하는 내용을 추가한다 (10sec)



➔ ip 없음 확인 가능

```
root@Server:~/바탕화면# apt install isc-dhcp-server -y
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다
```

➔ server dhcp-server 패키지 설치 (apt install)

```
114 subnet 192.168.111.0 netmask 255.255.255.0 {
115         option routers 192.168.111.2;
116         option subnet-mask 255.255.255.0;
117         range dynamic-bootp 192.168.111.55 192.168.111.99;
118         option domain-name-servers 8.8.8.8;
119         default-lease-time 10000;
120         max-lease-time 50000;
```

➔ /etc/dhcp/dhcpd.conf 파일 내용 추가

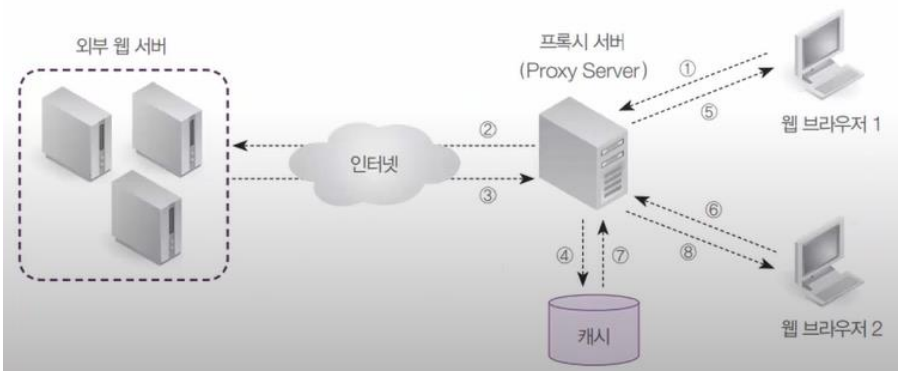
- 맨 하단 subnet, netmask 주소, 라우터와 서브넷마스크, ip 범위 등을 추가한다
- dhcp 서버는 빌려준 내역을 기록하는데 그 파일 내용은 /var/lib/dhcp/dhcpd.leases 서 확인 가능하다

```
root@server-b:~# ifconfig
ens32: flags=4163<UP,BROADCAST
        inet 192.168.111.55  n
        inet6 fe80::20c:29ff:f
        ether 00:0c:29:69:2a:b
```

```
ubuntu@client:~$ ifconfig
ens33: flags=4163<UP,BROADCAS
        inet 192.168.111.56
        inet6 fe80::5b9b:7def
        ether 00:0c:29:f6:a4:
```

➔ server(b), client ifconfig 로 ip 할당 확인

- 프록시 서버 개념
  - 프록시(Proxy)란 단어가 뜻하듯 '대리인'의 역할을 하는 서버
  - 웹 환경에서 프록시 서버는 웹 클라이언트와 웹 서버 사이에서 요청한 데이터를 전달하는 역할
  - 한번 전송한 데이터를 캐시에 저장한 후, 같은 데이터를 또 요청할 경우에 캐시에 저장된 것을 보내줌




```
root@Server:~/바탕화면# apt install squid -y
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다
상태 정보를 읽는 중입니다... 완료
```

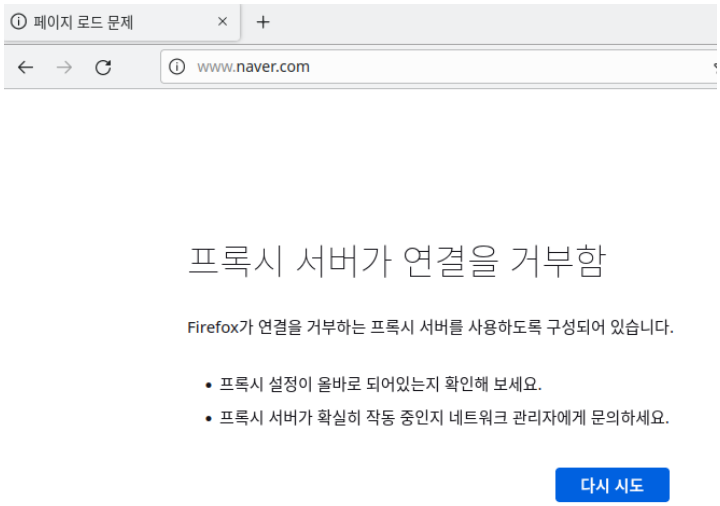
➔ squid 패키지 설치 (프록시 서버)

```
1 acl myserver src 192.168.111.0/255.255.255.0
2 http_access allow myserver
3 cache_dir ufs /var/spool/squid 1000 16 256
4 visible_hostname myserver
```

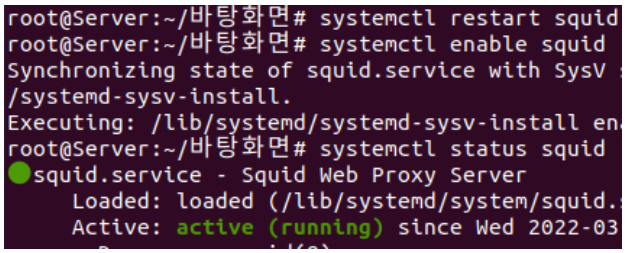
➔ /etc/squid/squid.conf 파일 내용 추가 (맨 상단)

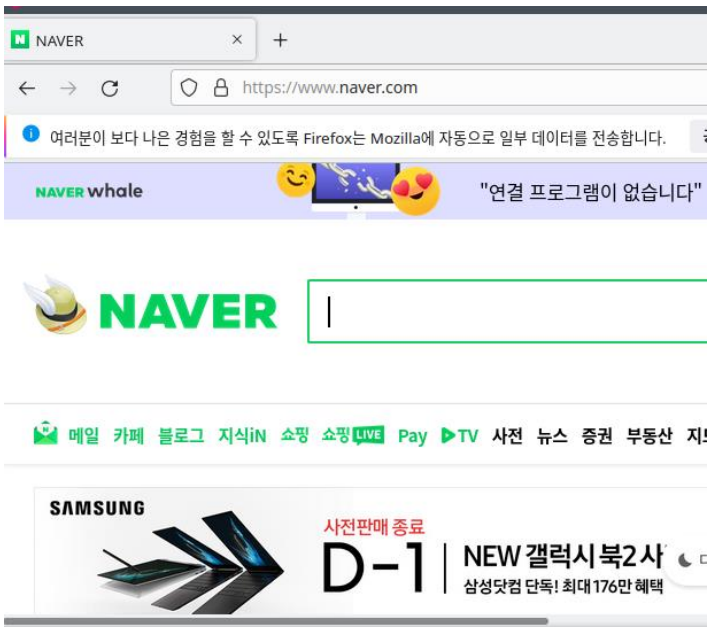
➔ 이후 방화역 비활성화/ systemctl stop squid

➔ 웹 브라우저 > 메뉴 > 네트워크설정 > 연결설정에서 프록시,포트 설정

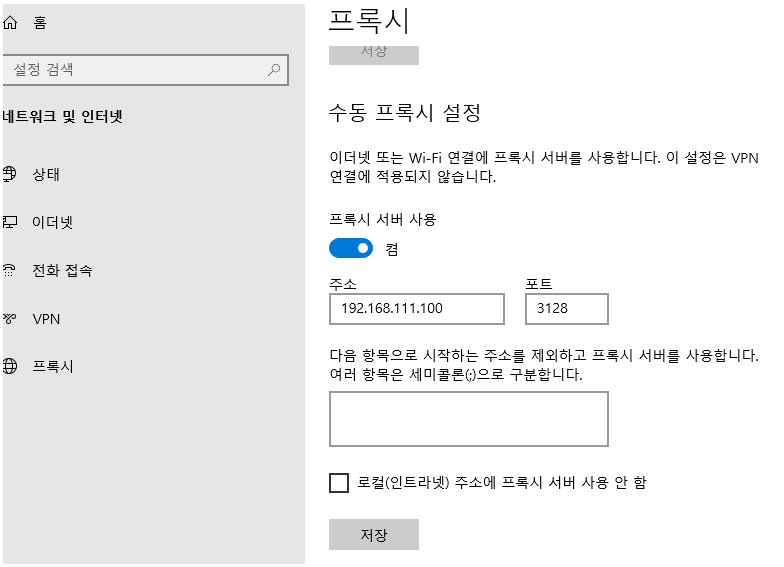


➔ 연결 거부 확인 (squid 를 정지시켰기 때문에 정상 반응임)



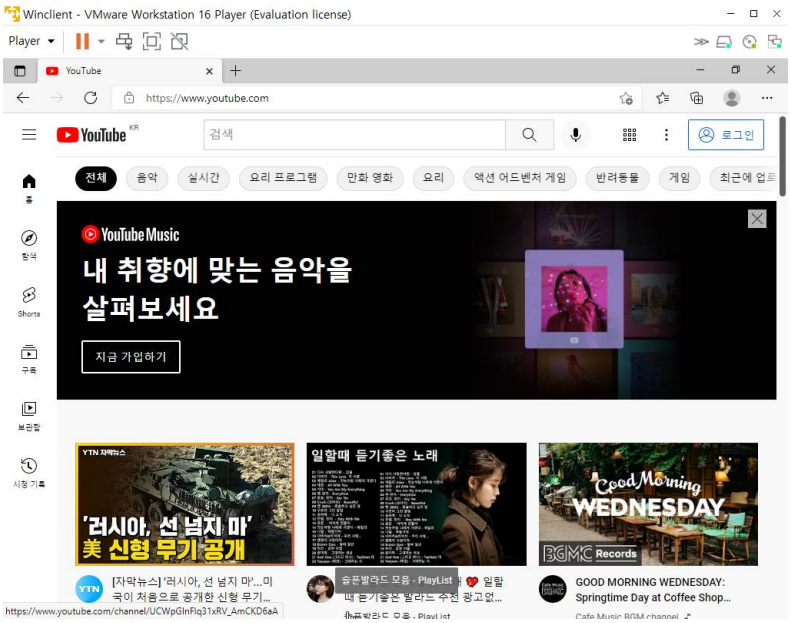


➔ server squid 시스템 재시작 후 client 인터넷 연결 확인



➔ 시작 > 설정 > 네트워크 및 인터넷 > 프록시

➢ 자동으로 설정 검색 off / 프록시 서버 사용 on 내용추가



➔ winclient 연결 성공 확인

- 보안을 위한 네트워크 설계

➢ 방화벽이란

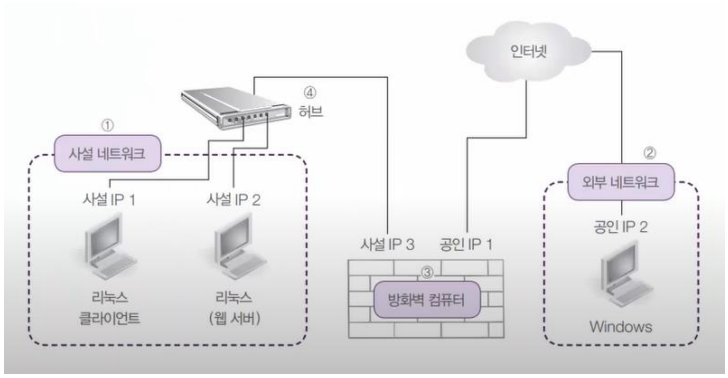
✓ 외부의 공개된 네트워크와 내부의 사설 네트워크 사이에 자리잡고, 외부와 내부에 전달되는 트래픽을 '정책(Policy)'에 의해서 허용/거부하는 역할을 하는 컴퓨터나 장치를 말함

- 내부의 사용자는 외부의 인터넷을 이용하면서, 외부에서는 내부로 침입할 수 없게 하는 방법 중 가장 보편적으로 많이 사용하는 방법이 사설 IP(Private IP)라고 흔히 불리는 nonroutable IP 주소를 이용함

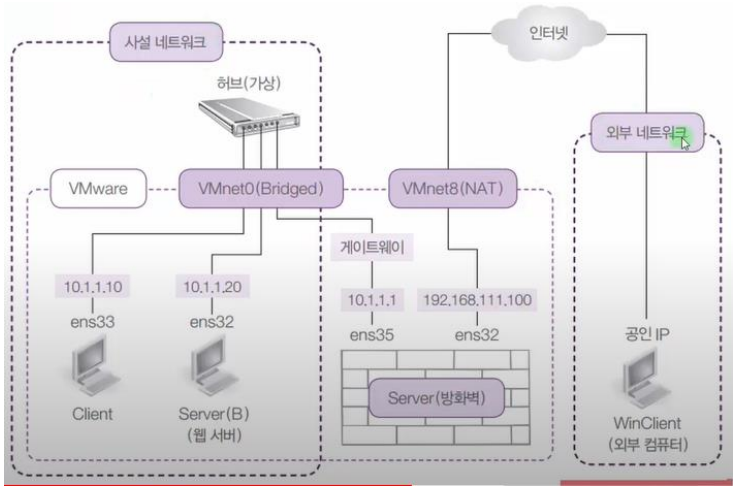
- 사설 IP 의 주소 범위는

  - ✓ 10.0.0.0 ~ 10.255.255.255, 172.16.0.0 ~ 172.31.255.255, 192.168.0.0~192.168.255.255 세 범위가 있음

- 사설 IP 주소의 컴퓨터가 외부의 인터넷으로 접속할 수 있도록 해주는 방법을 IP 마스커레잉딩(Masquerading)이라고 함

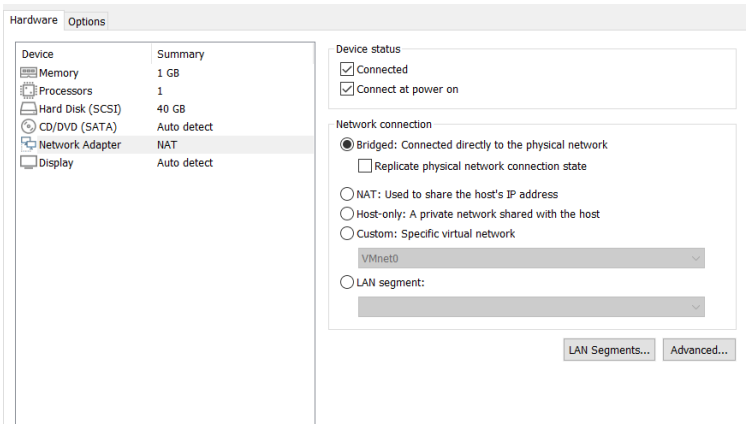


➔ 보편적인 회사 네트워크 구성

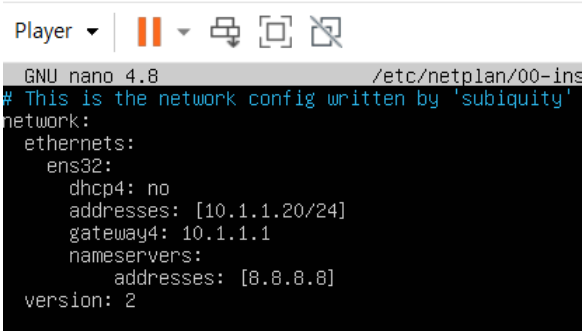


➔ 실습에서 구현할 네트워크 구성

- 방화벽 컴퓨터 구축 실습



➔ server(b) 네트워크어뎁터 > 기존 NET 에서 변경 후 확인



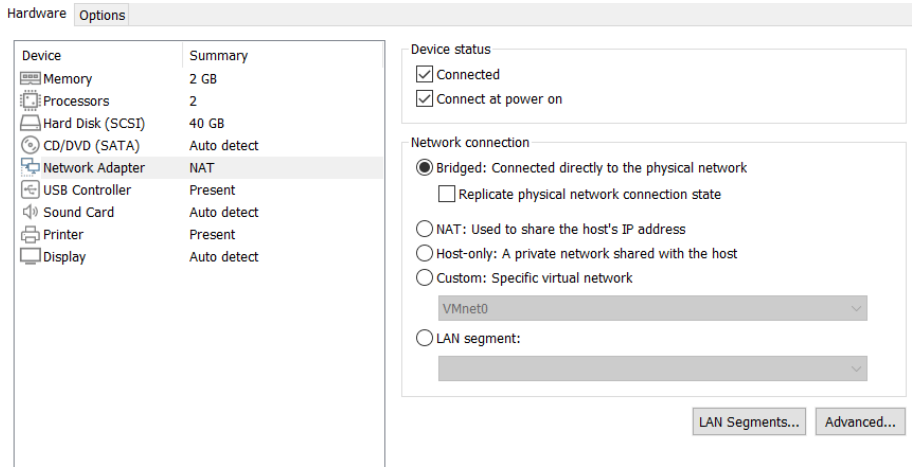
➔ 현재 고정된 ip(192.168.111.100)의 변경을 위해 /etc/netplan/00-installer-config.yaml 파일 수정

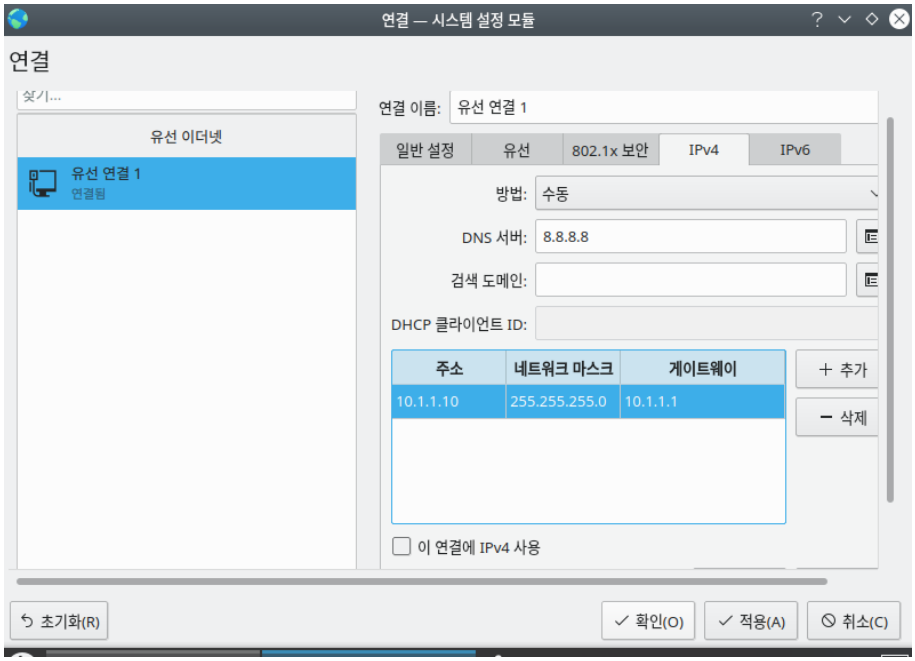
➢ addresses, gateway, namesevers 수정

➔ ip addr 명령어로 주소 확인

```
root@server-b:~# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
From 10.1.1.20 icmp_seq=1 Destination Host Unreachable
From 10.1.1.20 icmp_seq=2 Destination Host Unreachable
From 10.1.1.20 icmp_seq=3 Destination Host Unreachable
From 10.1.1.20 icmp_seq=4 Destination Host Unreachable
From 10.1.1.20 icmp_seq=5 Destination Host Unreachable
^Z
[1]+  Stopped                 ping 8.8.8.8
root@server-b:~#
```

➔ ping 명령어로 인터넷 연결 확인 (접속 불가) // 게이트웨이를 아직 생성하지 않음



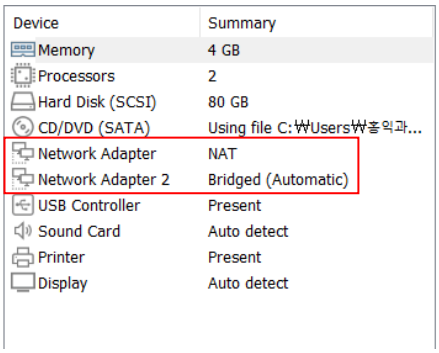


➔ Client 네트워크 어뎁터 설정 변경 후 인터넷 > 연결 – 시스템 설정 모듈 > ipv4 방법 수동 / 주소 추가

```
ubuntu@client:~$ ifconfig
ens33: flags=4163<UP,BROAD
        inet 10.1.1.10  ne
```

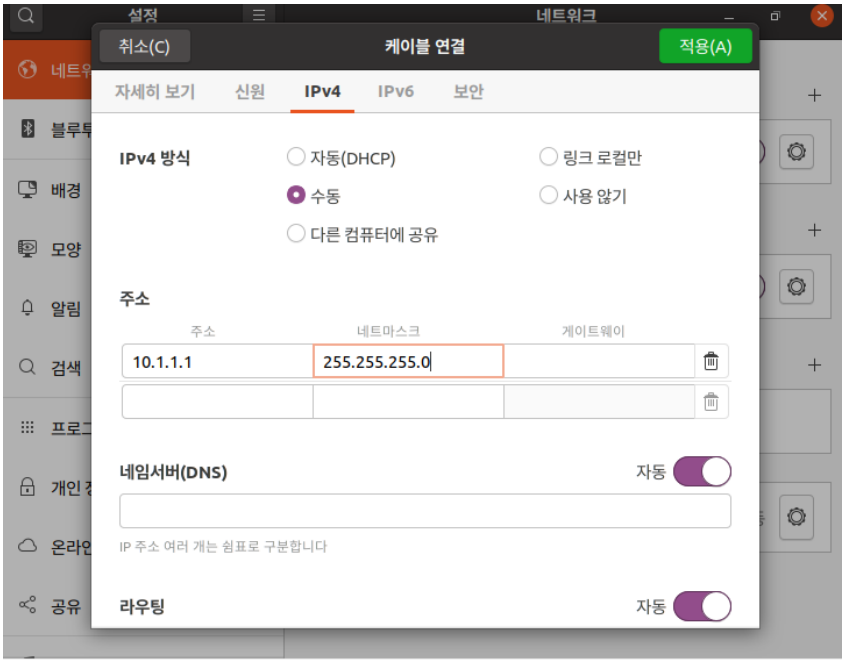
➔ 재부팅 후 client 주소 확인 가능 (정상적으로 적용)

➢ ping 명령어로 연결 확인 (연결 x)



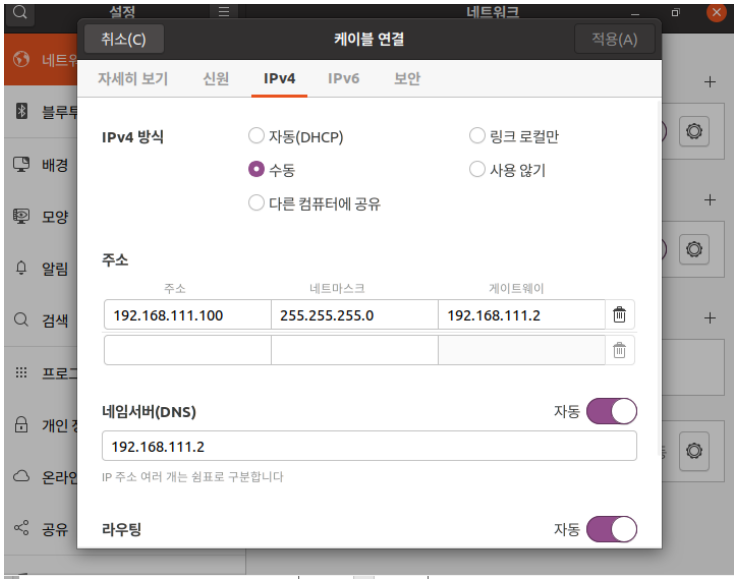
➔ server 에 랜카드(네트워크 어댑터) 추가 생성

➢ 기존에 사용하던 192.168.111.100, 추가할 10.1.1.1 총 2 개의 어댑터



➔ ens37 ipv4 설정

➢ 게이트웨이는 ens 33 이 실행하기 때문에 필요가 없음 (ens37 자신이 게이트웨이)



➔ ens33 의 설정 (유지) 외부 인터넷과 연결상태 양호

```
root@Server:~/바탕화면# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.111.100  netmask 255.255.255.0  broadcast 192.168.111.255
        inet6 fe80::358e:fcd6:24d:3047  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:e3:5f:10  txqueuelen 1000  (Ethernet)
        RX packets 129  bytes 64302 (64.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 142  bytes 14910 (14.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ens37: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.1.1.1  netmask 255.255.255.0  broadcast 10.1.1.255
        inet6 fe80::74af:3f2a:ec40:c0d8  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:e3:5f:1a  txqueuelen 1000  (Ethernet)
        RX packets 735  bytes 65808 (65.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 55  bytes 7408 (7.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

➔ 재부팅 후 ifconfig 로 주소 확인

```
26
27 # Uncomment the next line
28 net.ipv4.ip_forward=1
29
```

➔ /etc/sysctl.conf 파일 28 행 주석 제거

```
root@Server:~/바탕화면# echo   1   >   /proc/sys/net/ipv4/ip_forward
root@Server:~/바탕화면# cat /proc/sys/net/ipv4/ip_forward
1
```

➔ 설정한 ip 가 포워딩될 수 있도록 파일 생성 후 확인

```
root@Server:~/바탕화면# iptables  --policy  FORWARD  DROP
root@Server:~/바탕화면# iptables  --policy  INPUT  DROP
root@Server:~/바탕화면# iptables  --policy  OUTPUT  DROP
root@Server:~/바탕화면#
```

➔ iptable 초기화

```
root@Server:~/바탕화면# iptables --append  OUTPUT  --out-interface  ens37  --source  10.1.1.0/24  --dest
ination 0.0.0.0/0  --match  state  --state  NEW,ESTABLISHED  --jump  ACCEPT
root@Server:~/바탕화면# iptables --append  FORWARD  --in-interface  ens37  --source  10.1.1.0/24  --dest
ination 0.0.0.0/0  --match  state  --state  NEW,ESTABLISHED  --jump  ACCEPT
root@Server:~/바탕화면# iptables --append  FORWARD  --in-interface  ens33  --source  10.1.1.0/24  --dest
ination 0.0.0.0/0  --match  state  --state  NEW,ESTABLISHED  --jump  ACCEPT
root@Server:~/바탕화면#
```

➔ 사설네트워크가 server 를 통해서 외부에 접촉 가능을 위한 명령어 추가

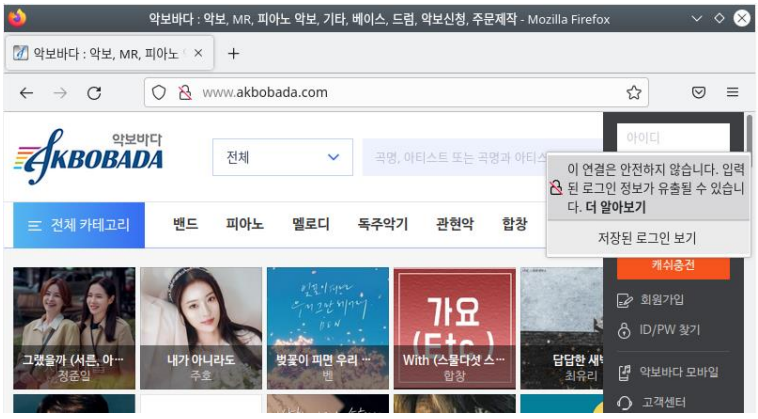
```
root@Server:~/바탕화면# iptables --table  nat  --append  POSTROUTING  --out-interface  ens33  --jump  MA
SQUERADE
root@Server:~/바탕화면#
```

➔ 위 명령어로 외부 접촉을 허용

```
root@Server:~/바탕화면# iptables-save  >  /etc/iptables.rules
root@Server:~/바탕화면#
```

➔ 변경 내용 저장

```
ubuntu@client:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=127 time=36.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=127 time=37.3 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=127 time=37.4 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=127 time=37.6 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=127 time=36.6 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=127 time=37.2 ms
^C
--- 8.8.8.8 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5010ms
rtt min/avg/max/mdev = 36.562/37.148/37.556/0.338 ms
ubuntu@client:~$
```



➔ ping, 웹 브라우저 등 연결 확인

- ➢ 외부 컴퓨터는 Client 가 아닌 방화벽 컴퓨터인 server 가 접속했다고 인지

■ server(b) 웹서버를 만들어 외부에서 허용가능하게 만드는 실습

- ➢ apt install apapche2
- ➢ allow http

- cd /var/www/html > 기존에 index.html 삭제 후 다시 생성
- 작성할 내용 추가한 뒤 시스템 재시작 (systemctl restart apache2)
- server(b)웹서버 생성 완료

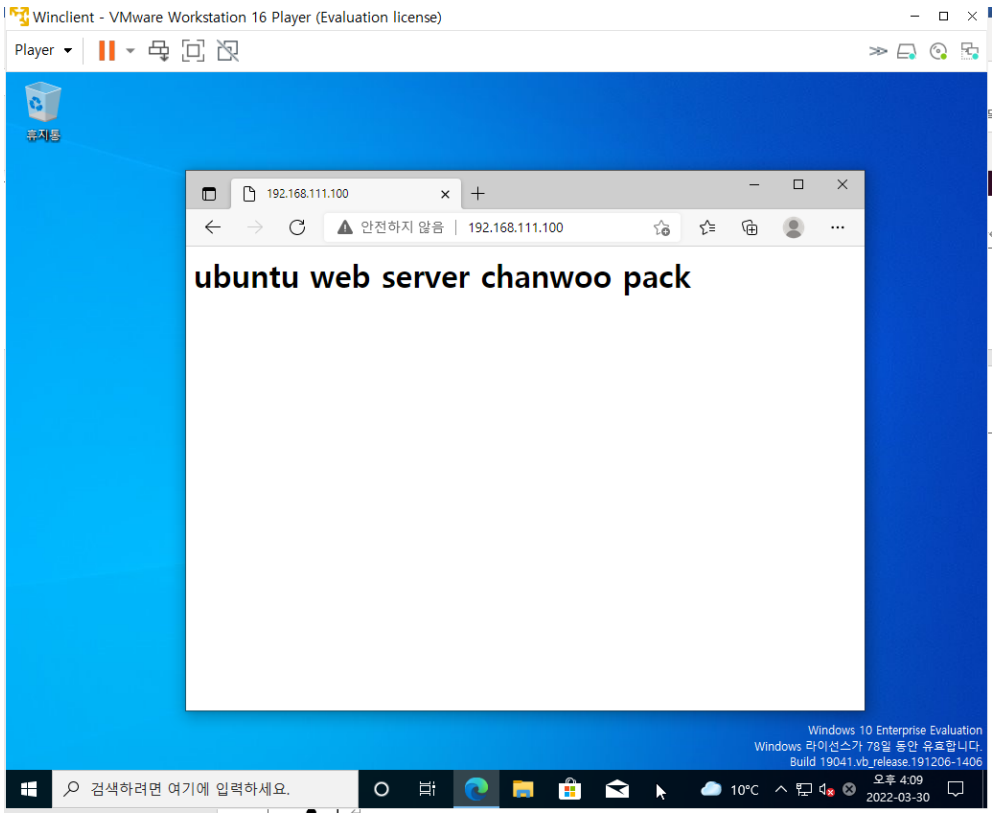
```
root@Server:~/바탕화면# iptables  --table  nat  --append  PREROUTING  --proto  tcp  --in-interface  ens3
3  --dport  80  --jump  DNAT  --to-destination  10.1.1.20
root@Server:~/바탕화면#
```

➔ 정책 추가

- ens33 으로 80port 요청이 오면 10.1.1.20 (serverb webserver) 로 연결해 줌

```
root@Server:~/바탕화면# iptables-save  >  /etc/iptalbe.rules
root@Server:~/바탕화면# ufw disable
방화벽이 비활성 되었으며 시스템이 시작할 때 사용되지 않습니다
root@Server:~/바탕화면#
```

➔ 저장 후 방화벽 비활성화



➔ 외부 사용자인 Winclient 에서 192.168.111.100 접속하니 serverB 로 접속시킨 모습