

# 리눅스 간 VPN

📅 날짜	@2024년 10월 11일
🌟 상태	완료



## 환경 구성

- 서버 (원)
  - 우분투 10.10.10.10
  - VPN IP : 192.168.10.1
- 클라이언트 (오)
  - 우분투 10.10.10.13
  - VPN IP : 192.168.10.2

## 0. 키 생성

### a. 서버

```
umask 077
```

```
wg genkey | tee /etc/wireguard/private.key | wg pubkey >  
/etc/wireguard/public.key
```

```
root@sumin-VM:/etc/wireguard# cat private.key  
wN+x3PUvRvVZeQd4cmICBajU1LzH24gx3tZaDEFvXW4=  
root@sumin-VM:/etc/wireguard# cat  
public.key  
vxP5k8DH032NIDdJaWZZVgMhpIDfYx+E0+J/rA8SASI=
```

### b. 클라이언트

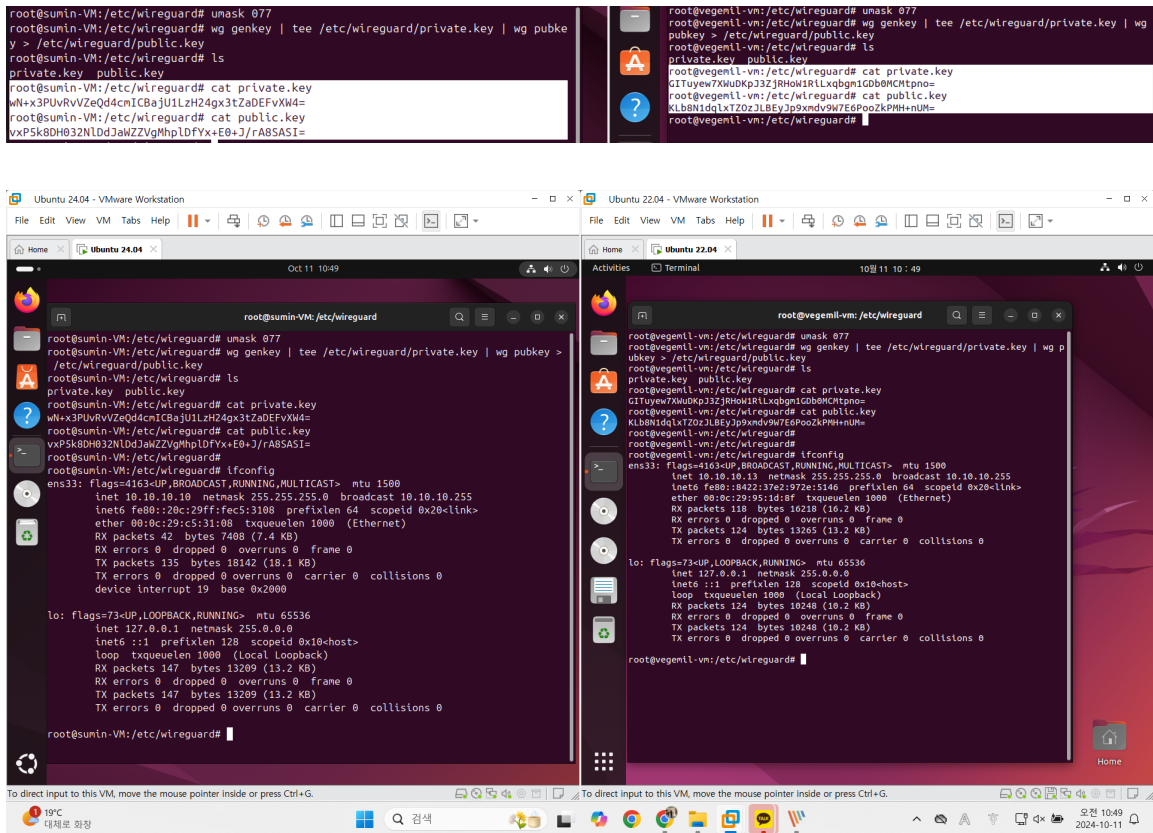
```
umask 077
```

```
wg genkey | tee /etc/wireguard/private.key | wg pubkey >  
/etc/wireguard/public.key
```

```
root@vegemil-vm:/etc/wireguard# cat private.key  
GI Tuyew7XWuDkPj3ZjRHoW1RiLxqbgm1GDb0MCMtpno=  
root@vegemil-vm:/etc/wireguard# cat
```

public.key

KLb8N1dqlxTZOzJLBEyJp9xmdv9W7E6PooZkPMH+nUM=



## umask 077을 하는 이유

- 파일을 생성할 때 기본 권한에서 특정 권한을 제거하는 것
  - 디렉터리 777과 파일 666에서 077을 하면 사용자 이외의 권한(그룹/기타 사용자)를 없앤다.
  - umask를 적용하면 경로와 무관하게 적용 후 생성되는 모든 파일과 디렉토리에 권한이 적용됨 (터미널을 닫거나 새로운 셸을 열면 기본 umask로 돌아감)
- WireGuard에서 **비밀키는 보안이 중요** → 다른 사용자가 접근하지 못하도록 제한
  - 비밀키가 유출되면 VPN 서버나 클라이언트에 대한 접근 권한을 잃을 수도 있음
  - 따라서 비밀키를 생성할 때는 umask 077을 설정하여 파일을 생성해야 함

각 설정파일은 /etc/wireguard/wg0.conf

### 1. 서버 설정 파일 구성

```

[Interface]
Address = 10.8.0.1/24                # 서버의 VPN IP 주소
SaveConfig = true                    # WireGuard 종료 시
PrivateKey = <서버의 비밀키>        # 서버의 비밀키 (wg genkey로 생성)
ListenPort = 51820                  # 서버가 수신 대기할 포트

# NAT 설정 및 트래픽 포워딩을 위한 iptables 규칙
PostUp = iptables -A FORWARD -i %i -j ACCEPT; iptables -t nat -A POSTROUTING -o ens33 -j MASQUERADE
PostDown = iptables -D FORWARD -i %i -j ACCEPT; iptables -t nat -D POSTROUTING -o ens33 -j MASQUERADE

[Peer]
PublicKey = <클라이언트의 공개키>    # 클라이언트의 공개키
AllowedIPs = 10.8.0.2/32              # 클라이언트의 VPN IP 주소

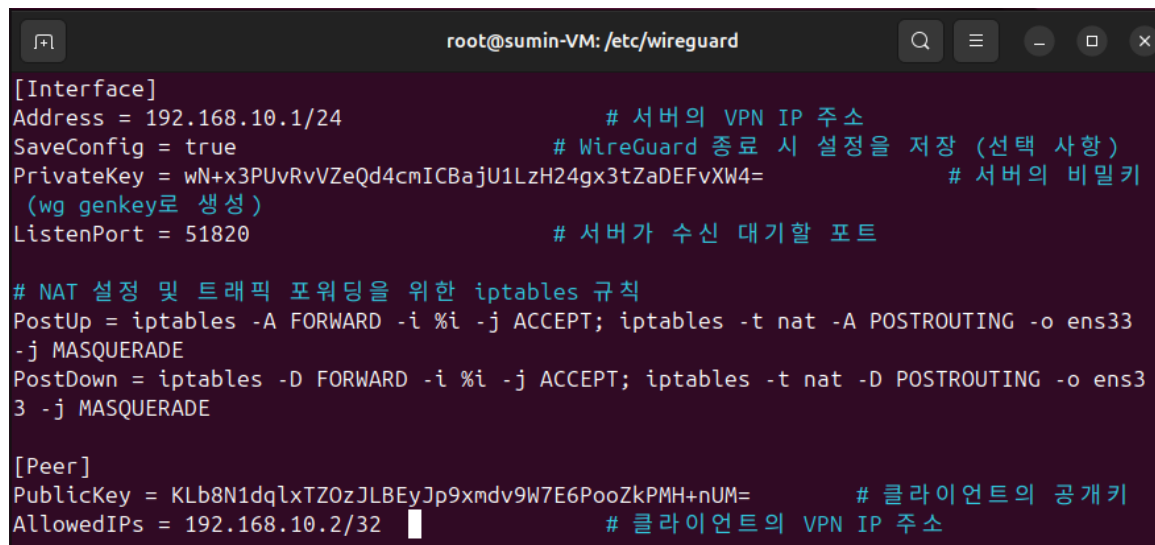
```

**[Interface]**는 서버의 IP 주소, 비밀키, 수신 대기 포트를 설정한다

이 때, Address는 VPN IP주소로 임의로 설정한다.

PostUP / PostDown은 wg0 인터페이스 활성화 시 실행할 iptables 규칙으로, eth0 enp0s3 등 서버의 네트워크 인터페이스 이름을 입력

**[Peer]**는 클라이언트와의 연결 정보 설정



```

[Interface]
Address = 192.168.10.1/24                # 서버의 VPN IP 주소
SaveConfig = true                        # WireGuard 종료 시 설정을 저장 (선택 사항)
PrivateKey = wN+x3PUvRvVZeQd4cmICBajU1LzH24gx3tZaDEFvXW4=    # 서버의 비밀키
(wg genkey로 생성)
ListenPort = 51820                      # 서버가 수신 대기할 포트

# NAT 설정 및 트래픽 포워딩을 위한 iptables 규칙
PostUp = iptables -A FORWARD -i %i -j ACCEPT; iptables -t nat -A POSTROUTING -o ens33 -j MASQUERADE
PostDown = iptables -D FORWARD -i %i -j ACCEPT; iptables -t nat -D POSTROUTING -o ens33 -j MASQUERADE

[Peer]
PublicKey = Klb8N1dqLxTZ0zJLBEyJp9xmdv9W7E6PooZkPMH+nUM=    # 클라이언트의 공개키
AllowedIPs = 192.168.10.2/32            # 클라이언트의 VPN IP 주소

```

## 2. 클라이언트 설정 파일

```

[Interface]
Address = 10.8.0.2/24                # 클라이언트의 VPN IP 주소
PrivateKey = <클라이언트의 비밀키>    # 클라이언트의 비밀키 (wg genkey로 생성)

```

```

DNS = 8.8.8.8 # (선택 사항) VPN 연결을 위한 DNS 서버 주소

[Peer]
PublicKey = <서버의 공개키> # 서버의 공개키
Endpoint = <서버의 공인 IP 주소>:51820 # 서버의 IP 주소 및 포트 번호
AllowedIPs = 0.0.0.0/0, :::/0 # 전체 트래픽을 VPN을 통해 전송 (라우팅 설정)
PersistentKeepalive = 25 # 연결이 유지되도록 keepalive 설정 (NAT 환경에서 유용)

```

서버와 마찬가지로 **[Interface]**에는 클라이언트의 IP주소와 비밀키 설정

**[Peer]**에는 연결할 서버의 정보와 허용된 IP 주소를 지정

Endpoint는 서버의 공인 IP주소와 포트번호를 설정한다.

AllowedIPs를 0.0.0.0/0으로 설정하면 모든 트래픽이 VPN을 통해 전송되도록 한다.

만약 특정 대역만 라우팅하고자 한다면 필요한 대역만 지정할 수 있다. ex) 10.8.0.0/24 이런 식으로 네트워크 대역 지정

```

root@vegemil-vm:/etc/wireguard# cat wg0.conf
[Interface]
Address = 192.168.10.2/24 # 클라이언트의 VPN IP 주소
PrivateKey = GITuyew7XWuDkPJ3ZjRHoW1RiLxqbgm1GDb0MCMtpno= # 클라이언트의 비밀키 (wg genkey로 생성)

[Peer]
PublicKey = vxP5k8DH032NlDdJaWZZVgMhplDfYx+E0+J/rA8SASI= # 서버의 공개키
Endpoint = 10.10.10.10:51820 # 서버의 IP 주소 및 포트 번호
AllowedIPs = 0.0.0.0/0, :::/0 # 전체 트래픽을 VPN을 통해 전송 (라우팅 설정)
PersistentKeepalive = 25 # 연결이 유지되도록 keepalive 설정 (NAT 환경에서 유용)

```

### 3. 설정 파일 적용

#### a. 서버

```
sudo wg-quick up wg0
```

```
sudo wg show
```

#### b. 클라이언트

```
sudo wg-quick up wg0
```

```
sudo wg show ]
```

```

root@sumin-VM:/etc/wireguard# wg
interface: wg0
  public key: vxP5k8DH032NldDJaWZZVgMhpLDfYx+E0+J/rA85ASI=
  private key: (hidden)
  listening port: 51820

peer: Klb8NldqLxTZ0zJLBEyJp9xmdv9W7E6PooZkPMH+nUM=
  endpoint: 10.10.10.13:59618
  allowed ips: 192.168.10.2/32
  latest handshake: 16 seconds ago
  transfer: 1.13 KiB received, 824 B sent
root@sumin-VM:/etc/wireguard#

root@vegemil-vm:/etc/wireguard# wg
interface: wg0
  public key: Klb8NldqLxTZ0zJLBEyJp9xmdv9W7E6PooZkPMH+nUM=
  private key: (hidden)
  listening port: 59618
  fwmark: 0xca6c

peer: vxP5k8DH032NldDJaWZZVgMhpLDfYx+E0+J/rA85ASI=
  endpoint: 10.10.10.10:51820
  allowed ips: 0.0.0.0/0, ::/0
  latest handshake: 15 seconds ago
  transfer: 824 B received, 1.13 KiB sent
  persistent keepalive: every 25 seconds
root@vegemil-vm:/etc/wireguard#

```

#### 4. 방화벽 설정 → 서버만 설정해도 충분!!

Wireguard 포트를 방화벽에서 허용해야 클라이언트가 서버에 접근할 수 있다.

```
sudo ufw allow 51820/udp
```

```
sudo ufw reload
```

```

root@sumin-VM:/etc/wireguard# ufw allow 51820/udp
Skipping adding existing rule
Skipping adding existing rule (v6)
root@sumin-VM:/etc/wireguard# ufw status
Status: active

To Action From
--
51820/udp ALLOW Anywhere
22/tcp ALLOW Anywhere
51820/udp (v6) ALLOW Anywhere (v6)
22/tcp (v6) ALLOW Anywhere (v6)

Anywhere on ens33 ALLOW FWD Anywhere on wg0
Anywhere (v6) on ens33 ALLOW FWD Anywhere (v6) on wg0

root@sumin-VM:/etc/wireguard# ufw reload
Firewall reloaded
root@sumin-VM:/etc/wireguard#

```

이제 서버의 VPN IP로 핑을 날려보자

```

root@vegemil-vm:/etc/wireguard# ping 192.168.10.1
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data:
64 bytes from 192.168.10.1: icmp_seq=1 ttl=64 time=1.99 ms
64 bytes from 192.168.10.1: icmp_seq=2 ttl=64 time=2.29 ms
64 bytes from 192.168.10.1: icmp_seq=3 ttl=64 time=3.69 ms
64 bytes from 192.168.10.1: icmp_seq=4 ttl=64 time=1.91 ms
64 bytes from 192.168.10.1: icmp_seq=5 ttl=64 time=2.33 ms
^C
--- 192.168.10.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 1.911/2.441/3.691/0.645 ms
root@vegemil-vm:/etc/wireguard#

```

ping test 성공