

WireGuard

Keygen

Make private and public keys for client and for server (run twice).

Fish:

```
wg genkey | read F && echo $F && echo $F | wg pubkey
```

Bash:

```
wg genkey | while read F; do echo $F && echo $F | wg pubkey; done
```

Server

```
[Interface]
PrivateKey = <server: wg genkey>
MTU = 1340
ListenPort = <server port>
Address = 172.17.1.1/24
PostUp = iptables -A FORWARD -o vpn -i enpls0 -j ACCEPT && iptables -A FORWARD -i vpn -o
enpls0 -j ACCEPT && iptables -t nat -A POSTROUTING -o enpls0 -j MASQUERADE
PreDown = iptables -D FORWARD -o vpn -i enpls0 -j ACCEPT && iptables -D FORWARD -i vpn -o
enpls0 -j ACCEPT && iptables -t nat -A POSTROUTING -o enpls0 -j MASQUERADE

[Peer]
PublicKey = <client: wg pubkey>
AllowedIPs = 172.17.1.10/32
PersistentKeepalive = 300
```

Notes:

- Write that in file `/etc/wireguard/vpn.conf`.
- Using `enp1s0` as example default interface, edit as needed.
- MTU setting depends on your network.

Client

```
[Interface]
PrivateKey = <client: wg genkey>
MTU = 1340
Address = 172.17.1.10/24
DNS = 172.17.1.1

[Peer]
PublicKey = <server: wg pubkey>
AllowedIPs = 0.0.0.0/0,::0/0
Endpoint = <server IP>:<server port>
```

Notes:

- Write that in file `/etc/wireguard/vpn.conf`

Allowed IPs to keep local network access

- <https://www.procustodibus.com/blog/2021/03/wireguard-allowedips-calculator/>

```
# VPN all traffic except 192.168.0.0/16
AllowedIPs = 0.0.0.0/1, 128.0.0.0/2, 192.0.0.0/9, 192.128.0.0/11, 192.160.0.0/13,
192.169.0.0/16, 192.170.0.0/15, 192.172.0.0/14, 192.176.0.0/12, 192.192.0.0/10, 193.0.0.0/8,
194.0.0.0/7, 196.0.0.0/6, 200.0.0.0/5, 208.0.0.0/4, 224.0.0.0/3
```

Start server/client

```
# From user (will ask for sudo password)
wg-quick up vpn

# As a service Void Linux
cd /var/service/
```

```
ln -s /etc/sv/wireguard
```

```
# As a service SystemD (e.g. Debian/Ubuntu)
```

```
sudo systemctl enable --now wg-quick@vpn
```

Revision #5

Created 2026-08-30 18:50:38 IST by hxd

Updated 2026-08-30 19:06:10 IST by hxd