WireGuard Fast VPN tunnels

CONCEPTS

Each peer is a public key plus the IPs it may use (AllowedIPs). The same list routes outgoing packets and filters incoming ones.

Packet to 10.0.0.2
↓ AllowedIPs
Peer matching key
↓ encrypt, UDP
Endpoint ip:port

There is no server or client in the protocol; any peer with a reachable Endpoint can accept connections.

KEYS

wg genkey > privatekey
Generate a private key

wg pubkey *< privatekey >* **publickey**
Derive the public key

wg genpsk > psk
Generate a preshared key

wg genkey | tee privatekey | wg pubkey
Both keys in one go

BRING UP AND DOWN

sudo wg-quick up *<iface>*
Start tunnel from /etc/wireguard/<iface>.conf

sudo wg-quick down *<iface>*
Stop tunnel and undo routes

sudo wg-quick up *<path>*.conf
Use a config file by path

sudo systemctl enable --now wg-quick@*<iface>*
Start now and at boot

sudo systemctl restart wg-quick@*<iface>*
Apply config changes (drops sessions)

INSPECT

sudo wg
Show all interfaces and peers

INSPECT (cont.)

sudo wg show *<iface>*
Show one interface

sudo wg show *<iface>* **latest-handshakes**
Last handshake per peer (epoch)

sudo wg show *<iface>* **transfer**
Bytes received and sent per peer

sudo wg show *<iface>* **allowed-ips**
AllowedIPs per peer

sudo wg showconf *<iface>*
Print live config in file format

CHANGE LIVE

sudo wg set *<iface>* **peer** *<pubkey>* **allowed-ips** *<cidr>* **endpoint** *<ip:port>*
Add or update a peer

sudo wg set *<iface>* **peer** *<pubkey>* **remove**
Remove a peer

sudo wg set *<iface>* **peer** *<pubkey>* **persistent-keepalive 25**
Keep NAT mapping alive

CONFIG FILE

[Interface]

Field	Meaning
PrivateKey	Own secret key
Address	Tunnel IP, e.g. 10.0.0.1/24
ListenPort	UDP port (51820 usual)
DNS	Resolver for the tunnel
MTU	Override detected MTU
Table	off stops route creation
PostUp / PostDown	Shell hooks; %i is iface
SaveConfig	Rewrite file on down

[Peer]

Field	Meaning
PublicKey	Peer's public key
AllowedIPs	IPs sent to peer
Endpoint	host:port to dial
PersistentKeepalive	Secs; 25 if NAT
PresharedKey	Optional extra secret

CONFIG FILE (cont.)

Minimal /etc/wireguard/wg0.conf

```
[Interface]
PrivateKey = <private key>
Address = 10.0.0.2/24

[Peer]
PublicKey = <peer public key>
Endpoint = vpn.example.com:51820
AllowedIPs = 10.0.0.0/24
PersistentKeepalive = 25
```

chmod 600 the config: it holds the private key, and wg-quick warns if it is readable by others.

ROUTING

AllowedIPs choices

Value	Effect
10.0.0.2/32	Only that one host
10.0.0.0/24	A whole subnet
0.0.0.0/0, ::/0	All traffic via peer

0.0.0.0/0 makes wg-quick set up policy routing so the default route goes through the tunnel; the endpoint itself stays reachable.

Overlapping AllowedIPs are allowed: the most specific match wins, and an identical range is moved to the later peer.

ip route get *<ip>*
Check which path a destination uses

sudo sysctl -w net.ipv4.ip_forward=1
Let this host forward packets

TROUBLESHOOT

sudo wg show
No handshake line means no contact yet

ping *<peer tunnel ip>*
Test the tunnel

Handshake but no traffic: check AllowedIPs on both sides, IP forwarding and the firewall.

No handshake: check the UDP port is open, the Endpoint, and that each side has the other's public key, not its own.