

WireGuard Cheat Sheet

Commands, config directives, MTU and keepalive tables, first-line checks. Built from the `wg(8)` and `wg-quick(8)` man pages and the vpnsmith.com troubleshooting guides.

■ Commands

KEYS

COMMAND	WHAT IT DOES
<code>wg genkey</code>	Private key (base64, 44 chars)
<code>wg pubkey < private.key</code>	Derive the public key
<code>wg genpsk</code>	Pre-shared key

INSPECT

<code>wg show</code>	State of every interface and peer
<code>wg show wg0 latest-handshakes</code>	Epoch timestamp of last handshake per peer, 0 = never
<code>wg show wg0 transfer</code>	Bytes received and sent per peer
<code>wg show wg0 endpoints</code>	Current public IP:port of each peer
<code>wg show wg0 dump</code>	Everything, tab-separated, for scripts

CHANGE THE TUNNEL LIVE

<code>wg-quick up wg0 / down wg0</code>	Bring interface up/down from <code>/etc/wireguard/wg0.conf</code> , applies routes and DNS
<code>wg synccnf wg0 <(wg-quick strip wg0)</code>	Apply config file changes without dropping the tunnel
<code>wg set wg0 peer <pubkey> remove</code>	Revoke a peer live
<code>wg set wg0 peer <pubkey> allowed-ips 10.8.0.9/32</code>	Add a peer live (also endpoint, persistent-keepalive, preshared-key)

SERVICE AND DEBUG

<code>systemctl enable --now wg-quick@wg0</code>	Start at boot
<code>journalctl -u wg-quick@wg0</code>	Why wg-quick refused to start
<code>qrencode -t ansiutf8 < client.conf</code>	QR code in the terminal for the mobile app

■ [Interface] keys

KEY	SIDE	NOTES
PrivateKey	both	Never share, mode 600
Address	both	Tunnel address with prefix, for example 10.8.0.1/24 on the server, 10.8.0.2/32 on a client
ListenPort	server	UDP. Clients should omit it (random port, better behind NAT)
DNS	client	Resolver used while the tunnel is up (wg-quick only, needs resolvconf)
MTU	both	Default 1420. See table below
Table	both	off to manage routes yourself, otherwise wg-quick adds routes for every AllowedIPs
PreUp PostUp PreDown PostDown	both	Shell commands, %i = interface name. Typical: nftables/iptables masquerade
SaveConfig	both	Keep false
FwMark	both	Packet mark for policy routing; wg-quick uses it automatically for 0.0.0.0/0 clients

■ [Peer] keys

KEY	NOTES
PublicKey	The other side's public key
PresharedKey	Same value both sides, optional but recommended
AllowedIPs	Two jobs at once: which source addresses are accepted from this peer, and which destinations are routed to it. Server side: the client's /32. Client side: 0.0.0.0/0, ::/0 for a full tunnel, or the subnets you want
Endpoint	host : port. Only needed on the side that initiates (the client). A server learns client endpoints from incoming packets
PersistentKeepalive	25 on the side behind NAT. Omit on servers

■ MTU table

WireGuard overhead: 60 bytes on IPv4 endpoints, 80 bytes on IPv6 endpoints. Tunnel MTU = path MTU to the endpoint minus overhead.

YOUR PATH	PATH MTU	TUNNEL MTU =
Plain Ethernet, IPv4 endpoint	1500	1440 (default 1420 is safe)
Plain Ethernet, IPv6 endpoint	1500	1420
PPPoE DSL/fibre (8 bytes less)	1492	1432 (IPv4) / 1412 (IPv6)
LTE/5G, Starlink, many hotel networks	1400 to 1472, unpredictable	1380
Double tunnel (WireGuard inside another VPN)	1420	1360
Anything that still hangs		1280 (IPv6 minimum, always works)

Test from inside the tunnel: `ping -M do -s <MTU minus 28> <peer tunnel IP>`. The largest size that passes, plus 28, is your usable MTU.

■ Keepalive

SITUATION	PERSISTENTKEEPALIVE
Phone/laptop behind home NAT, server with public IP	25 on the client
Both sides public IPs	omit
Home box behind CGNAT dialling out to a VPS	25 on the home box
Site-to-site, both routers behind NAT	25 on both, server-less (each is the other's endpoint through a VPS relay)
Battery concern	keepalive costs about 32 bytes every 25 s; on mobile use the app's on-demand setting instead of forcing the tunnel to stay up

■ Port choice

PORT	WHEN
51820/udp	Default. Fine everywhere except restrictive networks
443/udp	Networks that block "unknown" UDP but allow QUIC. Some DPI expects QUIC and drops it anyway
53/udp	Captive portals that leak DNS. Fragile, and some ISPs redirect port 53 to their own resolver
123/udp	NTP is rarely blocked, rarely inspected. Works on more networks than 443 in practice

PORT	WHEN
any TCP	Not possible natively. Use <code>udp2raw</code> , <code>ws-tunnel</code> or <code>phantun</code> to wrap WireGuard in TCP/WebSocket on the networks that need it

■ Addressing conventions used in the templates

NETWORK	PURPOSE
10.8.0.0/24	Road-warrior clients. Server .1, clients from .2
10.9.0.0/24	Site-to-site transit
10.10.0.0/24	CGNAT reverse tunnel between the home box and the VPS
fd00:8::/64	IPv6 inside the tunnel if you want it; otherwise delete the IPv6 lines

■ One-line diagnostics

```
# age in seconds of the last handshake per peer
sudo wg show wg0 latest-handshakes | awk -v now=$(date +%s) '{print $1, ($2==0)?"never":now-$2 " s ago"}'
```

```
# is UDP reaching the server at all
sudo tcpdump -ni WAN_IF udp port SERVER_PORT -c 5
```

```
# is forwarding on and masqueraded
sysctl net.ipv4.ip_forward && sudo nft list ruleset | grep -A2 postrouting
```

```
# largest packet that passes without fragmentation (run on the client)
for s in 1472 1412 1392 1372 1352 1252; do ping -c1 -W1 -M do -s $s 10.8.0.1 >/dev/null 2>&1 && echo "pass $((s+28))" && break; done
```

■ This page is free. The full kit is not.

This cheat sheet is one of the four documents of the **WireGuard Self-Hosting Kit** (9 USD). The kit adds what a reference card cannot give you:

- **Troubleshooting checklist:** the 12 causes of a tunnel that will not come up, in diagnostic order, each with the test command and the fix.
- **Hardening checklist:** 21 points, including 4 specific to `wg-easy`.
- **11 ready-to-edit templates:** server `wg0`, full and split tunnel clients, site-to-site A and B, CGNAT reverse tunnel (VPS and home side), `wg-easy` compose, `nftables.conf` in policy drop, `sysctl`.
- **2 scripts:** `wg-healthcheck.sh` and `wg-newpeer.sh` (also free on GitHub: github.com/ricco020/wireguard-ops-scripts).

Get the kit: ericgerard.gumroad.com/l/wireguard-kit

■ Need a server to run this on?

Every command above assumes a Linux box with a public IPv4 and UDP open. The cheapest one we point readers to is a Contabo Cloud VPS 4 (4 vCPU, 8 GB RAM, from about 5.50 EUR/month excl. VAT on a 24-month term, list price checked on contabo.com on 20 October 2026): [see the Contabo offer](#). This is an affiliate link: [vpnsmith](#) earns a commission if you buy, at no extra cost to you. For a WireGuard server serving a few devices, a smaller plan is enough: HostPapa Unmanaged VPS Lite (1 core, 1 GB RAM, 25 GB NVMe, 4 TB transfer, root SSH, no control panel) is listed at 2.95 USD/month on a 3-year term, then the regular rate, with a 30-day money-back guarantee (hostpapa.com, read on 28 October 2026): [see the HostPapa VPS offer](#). Also an affiliate link. A free Oracle Cloud Always Free VM also works if your region has capacity.

Guides behind each table: vpnsmith.com [WireGuard guides](#)

No multi-device test bench was used to write this material: it is built from the WireGuard documentation, the `wg` and `wg-quick` man pages and the troubleshooting guides published on vpnsmith.com.