OpenWRT WireGuard Client Setup guide using LuCi

Introduction

These are my notes for setting up WireGuard as a *Client*.
In essence WireGuard is a peer -to-peer protocol but because of differences in setup we still make a distinction between setting it up as a Client or as a Server, but a WireGuard interface can be setup to function as a Client and Server at the same time.

This guide was made on a NetGear R7800 running OpenWRT 24.10.0, screenshots are made with OpenWRT2020 theme which is not much different from the default theme.

My notes are using the easy way with a simple setup using LuCi although the corresponding config files are also listed.
This simple setup is done by importing a config file (.conf) from your VPN provider with necessary settings (see: config file).
Importing a config file is possible if you installed the *wg-tools* package (see Install WireGuard).
But just adding the settings manually will also do the trick.

Other useful information can be found in the OpenWRT WireGuard wiki.

Index

Install WireGuard

LuCi > System > Software: click *Update Lists*
Install: *luci-proto-wireguard*   (wireguard-tools should be installed automatically)

Download configuration

If you already have a configuration file then go to <u>Create WireGuard interface</u>, otherwise download a WireGuard configuration file from your provider or WireGuard Server.
In this example we are going to download a WireGuard configuration file from Proton which is free but it will expire after a week or so:
Create an account on https://protonvpn.com/
Login
Go to Downloads and scroll to the bottom for the WireGuard configuration.

Give a name to your config and choose router for your Platform :

WireGuard configuration

These configurations are provided to work with WireGuard routers and official clients.

1. Give a name to the config to be generated

Device/certificate name ⓘ

| wg_proton_nl |

2. Select platform

○ Android   ○ iOS   ○ Windows   ○ macOS   ○ GNU/Linux   ● Router

3. Select VPN options

⬤ NAT-PMP (Port Forwarding) <u>Learn more</u>

✓ VPN Accelerator <u>Learn more</u>

4. Select a server to connect to

Use the best server according to current load and position: **NL-FREE#70**

[Create]

Or select a particular server:

○ Standard server configs   ● Free server configs   ○ Secure Core configs

Scroll down to the server you want to connect to and Choose Create:

∧ 🇳🇱 Netherlands

Name	Status	Action
NL-FREE#1	ⓘ 63%	Create

Download the config file to your computer, the config file (wg_proton_nl-NL-FREE-1.conf) looks like this:
[Interface]
 # Key for wg_proton_nl
 # Bouncing = 3
 # NAT-PMP (Port Forwarding) = off
 # VPN Accelerator = on
PrivateKey = UJmovcwC7KQ/vfgnradTHoHD30WJ6SonkvXYg23ex0A=
Address = 10.2.0.2/32
DNS = 10.2.0.1
[Peer]
 # NL-FREE#1
PublicKey = vH2i8RY1qc66XfqwrixBpvH4K9GYJatkugJj0GHgoUQ=
AllowedIPs = 0.0.0.0/0
Endpoint = 217.23.3.76:51820

Add the `PersistentKeepAlive` so that the connection stays open:
PersistentKeepalive = 25 and if you use IPv6 add `::0/0` to allowed IPs:
AllowedIPs = 0.0.0.0/0, ::/0

==The result:==
[Interface]
 # Key for wg_proton_nl
 # Bouncing = 3
 # NAT-PMP (Port Forwarding) = off
 # VPN Accelerator = on
PrivateKey = UJmovcwC7KQ/vfgnradTHoHD30WJ6SonkvXYg23ex0A=
Address = 10.2.0.2/32
DNS = 10.2.0.1
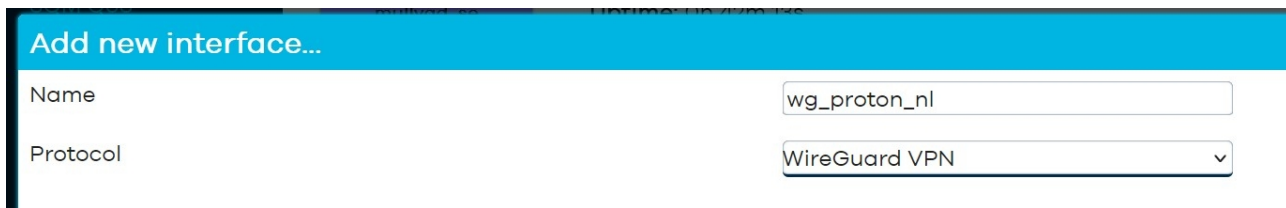*[Peer]*
 # NL-FREE#1
PublicKey = vH2i8RY1qc66XfqwrixBpvH4K9GYJatkugJj0GHgoUQ=
AllowedIPs = 0.0.0.0/0, ::0/0
Endpoint = 217.23.3.76:51820
PersistentKeepalive = 25

Note that there is no *listen port* in the Interface section, if you have a config file with a listen port then I recommend to delete the *listen port* so that the router can choose its own port which is not already taken.

Create WireGuard interface

Network > Interfaces on the bottom **click**: *Add New interface*

Add new interface...

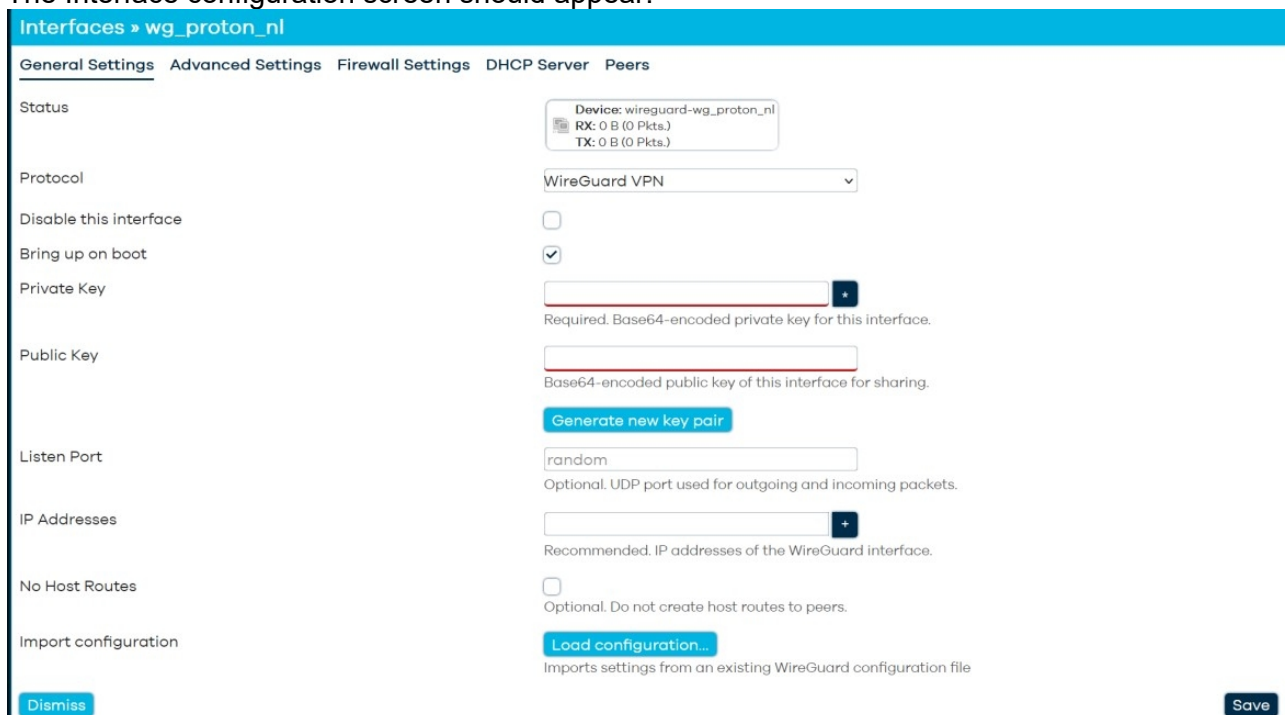| Name	wg_proton_nl
Protocol	WireGuard VPN

Name: give a descriptive name, hyphens are not allowed and the name has to be less than 15 characters!
Protocol: *WireGuard VPN*

Click: *Create interface*
The Interface configuration screen should appear:

Interfaces » wg_proton_nl

General Settings   Advanced Settings   Firewall Settings   DHCP Server   Peers

Status	Device: wireguard-wg_proton_nl RX: 0 B (0 Pkts.) TX: 0 B (0 Pkts.)
Protocol	WireGuard VPN
Disable this interface	☐
Bring up on boot	☑
Private Key	Required. Base64-encoded private key for this interface.
Public Key	Base64-encoded public key of this interface for sharing.
	Generate new key pair
Listen Port	random Optional. UDP port used for outgoing and incoming packets.
IP Addresses	Recommended. IP addresses of the WireGuard interface.
No Host Routes	☐ Optional. Do not create host routes to peers.
Import configuration	Load configuration... Imports settings from an existing WireGuard configuration file

Dismiss   Save

We can now import our configuration file by clicking the button *Load configuration*

Click: *Load configuration*
Drop the configuration file from the file manager into this box and automagically the settings should appear into the Interfaces configuration:

Interfaces » wg_proton_nl

General Settings   Advanced Settings   Firewall Settings   DHCP Server   Peers

Status	Device: wireguard-wg_proton_nl RX: 0 B (0 Pkts.) TX: 0 B (0 Pkts.)
Protocol	WireGuard VPN
Disable this interface	☐
Bring up on boot	☑
Private Key	●●●●●●●●●●●●●●●●●●●●●●●●●●●●●● * Required. Base64-encoded private key for this interface.
Public Key	1rMnp6/8iXg4uMdFNgkzWrSgLbl4uSqa6 Base64-encoded public key of this interface for sharing. Generate new key pair
Listen Port	random Optional. UDP port used for outgoing and incoming packets.
IP Addresses	10.2.0.2/32 - +

Advanced Settings:

Interfaces » wg_proton_nl

General Settings   Advanced Settings   Firewall Settings   DHCP Server   Peers

Force link	☐ Set interface properties regardless of the link carrier (If set handlers).
MTU	1420 Optional. Maximum Transmission Unit of tunnel interface.

MTU: can usually be left at its default setting (1420 or 1412 for PPPoE).
On occasion if you have slow or hanging connections especially when streaming, you have to lower the MTU, start lowering to 1280, but sometimes you have to go even lower. For some further explanation see MTU size problems at the end of this guide.

Note that you also have to enable MSS clamping (option mtu_fix '1') on the firewall zone the WireGuard interface is added to but this is covered in the firewall settings.

Create WireGuard Peers section

Network > Interfaces > wg_proton_nl: click *edit*
Go to *Peers* section:

Interfaces » wg_proton_nl

General Settings   Advanced Settings   Firewall Settings   DHCP Server   Peers

Further information about WireGuard interfaces and peers at wireguard.com.

Disabled	Description	Allowed IPs	Endpoint Host		
☐	wg_proton_nl-NL-FREE-1.conf vH2i8...HgoUQ=	0.0.0.0/0 ::/0	217.23.3.76:51820	≡	Edit

Add peer   Import configuration as peer...

Dismiss

Click: *Edit* and the Peers section will open:

Interfaces » wg_proton_nl » Edit peer

Disabled	☐ Enable / Disable peer. Restart wireguard interface to apply chang
Description	wg_proton_nl-NL-FREE-1.conf Optional. Description of peer.
Public Key	vH2i8RY1qc66XfqwrixBpvH4K9GYJatkug Required. Public key of the WireGuard peer.
Private Key	[ ] * Optional. Private key of the WireGuard peer. The key is not require allows generating a peer configuration or QR code if available. It c has been exported. Generate new key pair
Preshared Key	[ ] * Optional. Base64-encoded preshared key. Adds in an additional l post-quantum resistance. Generate preshared key
Allowed IPs	0.0.0.0/0 – ::/0 – [ ] + Optional. IP addresses and prefixes that this peer is allowed to us tunnel IP addresses and the networks the peer routes through the
Route Allowed IPs	☐ Optional. Create routes for Allowed IPs for this peer.
Endpoint Host	217.23.3.76 Optional. Host of peer. Names are resolved prior to bringing up th
Endpoint Port	51820 Optional. Port of peer.
Persistent Keep Alive	25 Optional. Seconds between keep alive messages. Default is 0 (disc is behind a NAT is 25.

Most settings should be automatically imported from your uploaded config but carefully check it:
- **Public key**: this is the public key of the server
- **Allowed IPs:** are set both for **IPv4** (*0.0.0.0/0*) and for **IPv6** (*::0/0*), for IPv6 only necessary if you have IPv6 implemented on your router.
- **Route Allowed IPs**: Enable (tick), this is not automatically set so must be done
- **Endpoint Host:** this is the IP address or Domain name of the server
- **Endpoint port**: this is the port the server is listening on, often this is *51820* but it could be different.
- **Persistent Keep Alive**: *25*, Check that this value is set, this keeps the connection alive

Click: *Save*
In the next window **Click:** *Save* again

In the Interface window click *Save & Apply*

/etc/config/network:
```
config interface 'wg_proton_nl'
        option proto 'wireguard'
        option private_key 'UJmovcwC7KQ/vfgnrasdfggdfgdfgdgddsgfdc='
        list dns '10.2.0.1'
        list addresses '10.2.0.2/24'

config wireguard_wg_proton_nl
        option description 'wg_proton_nl-NL-FREE-1.conf'
        option public_key 'vH2i8RY1qc66XfqwrixBpvH4K9dsfge4egdfgdfger='
        option endpoint_host '217.23.3.76'
        option endpoint_port '51820'
        list allowed_ips '0.0.0.0/0'
        list allowed_ips '::/0'    # leave in place for PBR
        list allowed_ips '::/1'
        list allowed_ips '8000::/1'
        option route_allowed_ips '1'
        option persistent_keepalive '25'
```

Note for IPv6 either
1. add `::/1` and `8000::/1` as Allowed IPs to create a default route, this has the "benefit" of reinstating default route via wan6 if the interface is disabled
or
2. Disable Source routing (Interfaces > Network > wan6 > Advanced tab: untick/disable IPv6 source routing

IPv6 source routing ☐
Automatically handle multiple uplink interfaces using source-based policy routing.

etc/config/network:
```
config interface 'wan6'
        option sourcefilter '0'`
```

Alternatively you can work with metrics and set appropriate metrics on WG interface and higher metrics on default route in wan and wan6, but this is more complicated as default metrics for IPv4 and IPv6 are not the same.

Next up Firewall

Firewall

Easy method

Easiest method is to just add the wg_proton_nl interface to the WAN zone

Network > Firewall > WAN zone > **Click**: *edit:*

Firewall - Zone Settings

General Settings   Advanced Settings   Conntrack Settings

This section defines common properties of "wan". The *input* and *output* options set the default policies for traffic entering and l
policy for forwarded traffic between different networks within the zone. *Covered networks* specifies which available networks a

Name	wan
Input	reject ⌄
Output	accept ⌄
Intra zone forward	reject ⌄
Masquerading	☑ Enable network address and port translation IPv4 typically enabled on the *wan* zone.
MSS clamping	☑
Covered networks	wan: 🖥️ wan6: 🖥️ ▾

Covered Networks: add *wg_proton_nl*

Covered networks	wan: 🖥️ wan6: 🖥️ wg_proton_nl: 🖥️ ▾

For IPv6 enable IPv6 Masquerading on the WireGuard firewall zone:
Advanced settings > Enable IPv6 Masquerading
   but restrict this to the IPv6 subnet of the WireGuard interface, this is the IPv6 `IP/List Addresses`
   in the WireGuard interface but with a /64 netmask

Firewall - Zone Settings

General Settings   Advanced Settings   Conntrack Settings

The options below control the forwarding policies between this zone (ovpn_client) and other zones. *Destination zones* cover forwarded traffic **originating from ovpn_client**. *Sour zones* match forwarded traffic from other zones **targeted at ovpn_client**. The forwarding rule is *unidirectional*, e.g. a forward from lan to wan does *not* imply a permission to forw from wan to lan as well.

Covered devices	unspecified ▾ Use this option to classify zone traffic by raw, non-*uci* managed network devices.
Covered subnets	[ ] + Use this option to classify zone traffic by source or destination subnet instead of networks or devices
IPv6 Masquerading	☑ Enable network address and port translation IPv6 (NAT6 or NAPT6) for outbound traffic on this zone.
Restrict to address family	IPv4 and IPv6 ⌄
Restrict Masquerading to given source subnets	fc00:bbbb:bbbb:bb01::6:4edd/64 -

```
/etc/config/firewall:
config zone
        option name 'wan'
        option input 'REJECT'
        option output 'ACCEPT'
        option forward 'REJECT'
        option masq '1'
        option mtu_fix '1'
        option masq6 '1'
        list masq_src 'fc00:bbbb:bbbb:bb01::6:4edd/64'
        list network 'wan'
        list network 'wan6'
        list network 'wg_proton_nl'
```

Click: *Save* and click *Save & Apply*

This should give you a working WireGuard Client

Check from the routers console with *curl ipinfo.io* and/or from your LAN clients with *ipleak.net*

Alternative Method

The Alternative method is to make a separate firewall zone for the VPN interface.
This can be useful if you want to make a killswitch (prevent traffic going out of the wan) or setup a Wireguard client on a Bridged AP.

Note that a killswitch is not really necessary as the WireGuard interface stays up even if there is no connection but it will add an extra layer of security and guards against mis-configuration.
Furthermore a killswitch is not compatible with Policy Based Routing (PBR).

Network > Firewall > **Click**: *Add*:
- **Name**: *vpn_client*
- **Input**: *reject*
- **Output**: *accept*
- **Intra zone forward**: *reject*
- **Masquerading**: *enabled*
- **MSS clamping**: *enabled*

- **Allow forward from source zone**: *lan*

Firewall - Zone Settings

General Settings   **Advanced Settings**   **Conntrack Settings**

This section defines common properties of "this new zone". The *input* and *output* options set the default policies for traffic entering describes the policy for forwarded traffic between different networks within the zone. *Covered networks* specifies which available

Name	vpn_client
Input	reject
Output	accept
Intra zone forward	reject
Masquerading	☑ Enable network address and port translation IPv4 (NA typically enabled on the *wan* zone.
MSS clamping	☑
Covered networks	wg_proton_nl:

The options below control the forwarding policies between this zone (this new zone) and other zones. *Destination zones* cover forw *zones* match forwarded traffic from other zones **targeted at this new zone**. The forwarding rule is *unidirectional*, e.g. a forward fro from wan to lan as well.

Allow forward to *destination zones*:	unspecified
Allow forward from *source zones*:	lan lan:

If your VPN provider also supports **IPv6** (with ULA addresses) then on
Advanced Settings:
- **IPv6 Masquerading**: *enable*

General Settings   **Advanced Settings**   Conntrack Settings

The options below control the forwarding policies between this zone (vpn_client) and other zones. *Destination zones* cover for
match forwarded traffic from other zones **targeted at vpn_client**. The forwarding rule is *unidirectional*, e.g. a forward from lar
to lan as well.

Covered devices	unspecified ▼
	Use this option to classify zone traffic by raw, non
Covered subnets	[ ] +
	Use this option to classify zone traffic by source c
IPv6 Masquerading	☑
	Enable network address and port translation IPv6
Restrict to address family	IPv4 and IPv6 ▼
Restrict Masquerading to given source subnets	0.0.0.0/0 +
Restrict Masquerading to given destination subnets	0.0.0.0/0 +

etc/config/firewall:
```
config zone
        option name 'vpn_client'
        option input 'REJECT'
        option output 'ACCEPT'
        option forward 'REJECT'
        option masq '1'
        option mtu_fix '1'
        list network 'wg_proton_nl'
        option masq6 '1'# only for IPv6
```

To prevent traffic going out of the wan (the Killswitch) **Edit** the lan firewall zone and disable forwarding to wan and only allow forwarding to the vpn_client zone.

Firewall – Zone Settings

General Settings   Advanced Settings   Conntrack Settings

This section defines common properties of "lan". The *input* and *output* options set the default policies for traffic entering and
policy for forwarded traffic between different networks within the zone. *Covered networks* specifies which available networks

Name	lan
Input	accept ▼
Output	accept ▼
Intra zone forward	accept ▼
Masquerading	☐
	Enable network address and port translation IP typically enabled on the *wan* zone.
MSS clamping	☐
Covered networks	lan: 🖧 ▼

The options below control the forwarding policies between this zone (lan) and other zones. *Destination zones* cover forwarde
forwarded traffic from other zones **targeted at lan**. The forwarding rule is *unidirectional*, e.g. a forward from lan to wan does

Allow forward to *destination zones*:	**vpn_client** wg_proton_nl: 🖧 ▼

/etc/config/firewall:
```
config forwarding
        option src 'lan'
        option dest 'vpn_client'
```

Allowing Specific client(s) WAN access

If you want to let just specific lan clients to have WAN access e.g. in case of Policy Based Routing you can make a traffic rule with the IP of MAC addresses of that specifc lan client(s) e.g.:
etc/config/firewall:
```
config rule
        option name 'allow_wan'
        option src 'lan'
        option src_ip '192.168.1.50/32'    # for IPv4 use the IPv4 address
        option src_mac '00:11:22:33:44:55' # for both IPv4 and IPv6 use the MAC address
        option dest 'wan'
        option target 'ACCEPT'
```

Make sure this rule is on top of the other rules, so add it before the other rules or use the GUI to drag the rule to the top.

Check

When everything is done **Reboot** the router and check that traffic flows:
Status > WireGuard:
You should see **Latest Handshake** with a new handshake every 3-4 minutes and traffic in both directions.
Or look at:
NetWork > Interface:
You should see traffic both for RX and TX indicating a working connection

Check your **lan clients** by browsing to *ipleak.net*

DNS Leak

On a typical phone (Android, iOS) or Windows the DNS is just set on the WireGuard interface and the DNS set is used after the tunnel is up.
On the OpenWRT router things are much more complicated (in contrast to other third party firmwares which handles this much better)
For some background reading: https://github.com/egc112/OpenWRT-egc-add-on/tree/main/stop-dns-leak

Be very careful with using the DNS server from your VPN provider as sole DNS server if that DNS server is not publicly available as you might end up in a catch 22 situation because the router must have the correct

time (more or less) before it can connect and to get the correct time it needs DNS resolving which is not available.
So in that case instead of a domain for time server use IP addresses (System > System > Time Synchronization)
Of course if you stop the tunnel you do not have DNS resolution in that case you need a scripting solution to use the VPN DNS server after the tunnel is up.

WireGuard Client on a BridgedAP

If you want to setup a WireGuard Client on a BridgedAP, then be aware that normal traffic from your clients just bypasses your BridgedAP, so will not use the WireGuard tunnel unless you point the gateway of your clients to the BridgedAP (by using DNSMasq tagging with option 3 or iptables redirect) or setup a Guest Wifi on the BridgedAP.
In case of using a Guest wifi all clients using your Guest wifi will automatically use the WireGuard tunnel, so this is the more easier option.

First double check that you have setup your BridgedAP correctly see:
https://openwrt.org/docs/guide-user/network/wifi/wifiextenders/bridgedap

For a Guest Wifi on a BridgeAP see:
https://openwrt.org/docs/guide-user/network/wifi/guestwifi/guestwifi_dumbap
Setup the Guest wifi and check that it is working without WireGuard!
Note: do not forget to Enable Masquerading on the LAN zone

Setup a WireGuard client the regular way as described, but for firewall settings I recommend to use a separate zone for the WireGuard Interface (Alternative Method) as you need to enable MSS Clamping and then make a Forward rule to Forward from *guest* zone to *vpn_client* zone.
If you then remove the forwarding from guest zone to lan zone you will have an effective killswitch.

For IPv6 make sure your lan has an IPv6 with prefix delegated, the Guest interface will then get its own IPv6 address from Lan

Clients directly connected to the main router will bypass this router anyway but you can deal with this by manually setting the gateway on the client to point to your BridgedAP.
A second method is to have DNSMasq hand out a different gateway, this can be done per client with DNSMasq tagging with option 3(gateway), etc/config/dhcp"

```
config tag 'tag1'
        option dhcp_option '3,ipaddress-of-BridgedAP'

config host
        option name 'client1'
        option mac '00:21:63:75:aa:17'
        option ip '10.11.12.14'
        option tag 'tag1'
```

The third method is by using policy based routing on the main router:
Create a routing table with default route via the BridgedAP and also local route for br-lan e.g.:

```
ip route add default via 192.168.0.9 table 2   # 192.168.0.9 is my BridgeAP
ip route add 192.168.0.0/24 dev br-lan table 2
```

For IPv6 routing you need the IPv6 address of the br-lan interface of the BridgedAP see: Setup IPv6 on a bridgedAP, you can get the address with `ifstatus lan 6 | grep address` or with ifconfig, you need the IPv6 address in the same subnet as the main router
This will get you a route like:

```
ip -6 route add default via 2001:8b11:234:1a33:a697:33ff:dcab:a3f2 dev br-lan table 2
```

Create a rule adding the lan clients involved to use this routing table 2 e.g.:

```
ip rule add 192.168.0.80 table 2   # 192.168.0.80 is the lan client I want to use the VPN
```

For IPv6 create the same rule with the IPv6 address of your client, unfortunately Windows uses its Temporary IPv6 address unless this is disabled and as it is temporary this is not a solid solution so disable the use of private IPv6 addresses e.g.:
```
ip -6 rule add from 2001:1ba2:236:a100::6f1 table 2
```

Make sure "Invalid traffic" is allowed or SNAT traffic from the LAN clients (e.g. 192.168.0.80) coming out of br-lan otherwise traffic can be blocked as there is asymmetric routing.

Policy Based Routing on the Bridged AP

When you have multiple Guest subnets on the router all will be routed via the WireGuard VPN. You can use Policy Based Routing just as if this was a regular router to route traffic to your liking via WireGuard or via the LAN to the main router.
Note: if you use the PBR app you have to set the lan interface as wan (option uplink_interface(6) 'lan(6)').
As `option lan_device` you might need to set the device which you want to route via the LAN instead of br-lan. See: https://forum.openwrt.org/t/guest-wifi-on-dumb-ap/242830/20?u=egc

Asking for Help

You can ask for help at the OpenWRT forum.

If you do, it helps if we can have a look at your configs, so please connect to your OpenWRT device using ssh and copy the output of the following commands and post it on the forum using the "Preformatted text </> " button

Remember to redact keys, passwords, MAC addresses and any public IP addresses you may have:
```
ubus call system board
cat /etc/config/network
cat /etc/config/wireless
cat /etc/config/firewall
wg show
```

To view the log for errors:
```
logread | grep -E -i 'netifd|wireguard'
```

References

https://openwrt.org/docs/guide-user/services/vpn/wireguard/start
https://openwrt.org/docs/guide-user/services/vpn/wireguard/basics
https://openwrt.org/docs/guide-user/services/vpn/wireguard/client

https://protonvpn.com/support/openwrt-wireguard

Miscellaneous

Setup IPv6 on a bridgedAP

/etc/config/network:
```
config interface 'lan6'
        option ifname '@lan'
        option proto 'dhcpv6'
        #option reqprefix 'no'
        option reqprefix '62'    #for ipv6 guest interface
        #option sourcefilter '0'  # disable source routing for WG server routing of IPv6
```

Prevent Mullvad from hijacking your DNS

https://schnerring.net/blog/use-custom-dns-servers-with-mullvad-and-any-wireguard-client/

Port forwarding via Proton VPN with natpmpc

https://protonvpn.com/support/port-forwarding-manual-setup/
https://forum.openwrt.org/t/openwrt-protonvpn-and-pmp-port-forwarding-for-remote-ssh-access/229367

MTU size problems (no traffic, hang, slow loading, no streaming media, bad VoiP, slow RDP)

The MTU (Maximum Transmission Units) is the maximum datagram size in bytes that can be sent unfragmented over a particular network path, so to have the highest throughput you want the highest value without fragmentation.

But if the MTU is too high and the packets fragment this will manifest as connections which hang during periods of active usage, or does not load the whole page when browsing. Or you can connect but not see or use streaming media (like an IP Camera) or your connection is unexpected slow, VoIP can also be affected.
In the log you can see messages like: read UDP [EMSGSIZE path-MTU=1388]: Message too long (code=90) but that is not always present.

To test for the highest value basically use ping from a client on your local LAN, search for the maximum packetsize which can be send unfragmented , add 28 and that is your MTU value, see:
https://hamy.io/post/000c/how-to-find-the-correct-mtu-and-mru-of-your-link/

ICMP blackhole test to see if PMTUD is working:
 http://icmpcheck.popcount.org or http://icmpcheckv6.popcount.org/
Packet loss test: https://packetlosstest.com/

Android, Windows and iOS can use their own defaults so when having problems connecting from Windows, Android or iOS try to delete the MTU entry in the conf file and let the OS itself decide what to use.

WireGuard Watchdog

https://github.com/egc112/OpenWRT-egc-add-on/tree/main/wireguard-watchdog
Purpose: WireGuard watchdog with fail-over, by pinging every x seconds through the WireGuard interface, the WireGuard tunnel is monitored.
In case of failure of the WireGuard tunnel the next tunnel is automatically started.
When the last tunnel has failed, the script will start again with the first tunnel.
So in case you have only one tunnel this is just a watchdog which restarts the one tunnel you have.

with prioritizing :
https://forum.openwrt.org/t/bash-script-for-automatic-change-between-2-wireguard-tunnels/228696/11

WireGuard Companion

https://github.com/egc112/OpenWRT-egc-add-on/tree/main/wireguard-companion

Purpose: Toggle WireGuard tunnels on/off, show status and log.

Custom user files for Azure, AWS etc

https://forum.openwrt.org/t/policy-based-routing-pbr-package-discussion/140639/2051?u=egc

Centralized Management and Deployment

https://help.netmaker.io/en/articles/9773942-making-openwrt-successfully-integrate-with-the-netmaker-network