# VPN

## Overview

- [\*\*OpenVPN - Easy to use, minimal hassle VPN server\*\*](#openvpn)

- [\*\*PiVPN - OpenVPN server installer and management tool\*\*](#pivpn)

- [\*\*WireGuard - An extremely simple yet fast and modern VPN\*\*](#wireguard)

??? info "How do I run \*\*DietPi-Software\*\* and install \*\*optimised software\*\* items?"

To install any of the \*\*DietPi optimised software items\*\* listed below run from the command line:

```sh

dietpi-software

```

Choose \*\*Browse Software\*\* and select one or more items. Finally select `Install`.

DietPi will do all the necessary steps to install and start these software items.

![DietPi-Software menu screenshot](../assets/images/dietpi-software.jpg){: width="643" height="365" loading="lazy"}

To see all the DietPi configurations options, review the [DietPi Tools](../../dietpi\_tools/) section.

[Return to the \*\*Optimised Software list\*\*](../../software/)

## OpenVPN

An easy to use VPN server and client system. The DietPi installation of OpenVPN uses a single client file to get you connected with minimal hassle.

![OpenVPN logo](../assets/images/dietpi-software-vpn-openvpn-logo.png){: width="200" height="58" loading="lazy"}

=== "Client connection file"

#### Generate client connection file for your VPN client system

As a prerequisite, a client connection file (`DietPi\_OpenVPN\_Client.ovpn`) has to be obtained and put on your target system where your VPN client is running.

DietPi will automatically generate unique 2048 bit server and client keys during installation and place them into a unified client config file. You will need this file to connect to your OpenVPN server from a client.

Client file location:

- DietPi will generate the client config file and place it here:

`/boot/DietPi\_OpenVPN\_Client.ovpn`.

Simply power off and plug the SD card into your target system to obtain the file from the FAT partition.

- DietPi will also create a copy of the file in

`/mnt/dietpi\_userdata/DietPi\_OpenVPN\_Client.ovpn`.

Use one of DietPi's file servers to access this file.

???+ warning "Security issue"

For security reasons, please remove those client connection files after they have been deployed on the client system!

#### Changing the target address for the client file

You will need to open the `DietPi\_OpenVPN\_Client.ovpn` file in a text editor to change the target domain/IP address. This can be anything from a website address, No-IP domain name, or IP address.

Examples for changing `mywebsite.com`. e.g.:

- `remote MySuperDooperWebsite.com 1194`

- `remote 81.252.0.1 1194`

=== "Router setup"

You have to set up your router to enable external access.

OpenVPN server uses the following ports:

- TCP 443

- TCP 943

- UDP 1194

These ports must all be enabled in port forwarding on your router and point to the IP address of your DietPi system.

=== "Windows client"

Installation of the Windows OpenVPN client program is done with the following steps:

1. Download the software under

URL = <https://openvpn.net/community-downloads/>

2. Download and install the installer that suites to your Windows version.

=== "Connecting to your OpenVPN server (Windows)"

Method 1 - Quick:

Simply right click the `DietPi\_OpenVPN\_Client.ovpn` file and choose "Start OpenVPN on this config file".

Method 2 - GUI:

If you want to use the OpenVPN GUI, you will need to copy `DietPi\_OpenVPN\_Client.ovpn` to the OpenVPN config location (e.g.: `C:\Program Files\OpenVPN\config`).

=== "OpenVPN + Pi-hole"

To allow VPN clients accessing your local Pi-hole instance, you need to allow DNS requests from all network interfaces:

`pihole -a -i local`

\*\*\*

Website: <https://openvpn.net>

Wikipedia: <https://wikipedia.org/wiki/OpenVPN>

Installation article (German language): [PiVPN: Raspberry Pi mit OpenVPN – Raspberry Pi Teil3](https://www.kuketz-blog.de/pivpn-raspberry-pi-mit-openvpn-raspberry-pi-teil3/){:class="nospellcheck"}

## PiVPN

PiVPN is an OpenVPN and WireGuard installer and management tool. It also has a command `pivpn` which allows for simple creation of additional user profiles and configurations.

![PiVPN logo](../assets/images/dietpi-software-vpn-pivpn-logo.png){: width="100" height="100" loading="lazy"}

=== "Using PiVPN"

Run the command `pivpn` to see a list of options.

=== "Create a new user profile"

Simply run the command `pivpn -a`.

\*\*\*

Website: <https://www.pivpn.io>

YouTube video tutorial: \*VPN configuration using Raspberry Pi and DietPi\*.

<iframe src="https://www.youtube-nocookie.com/embed/aYPaDeqtMG8?rel=0" frameborder="0" allow="fullscreen" width="560" height="315" loading="lazy"></iframe>

YouTube video tutorial: \*DietPi PiVPN Server Setup on Raspberry Pi 3 B Plus\*.

<iframe src="https://www.youtube-nocookie.com/embed/0t0bwskZJFw?rel=0" frameborder="0" allow="fullscreen" width="560" height="315" loading="lazy"></iframe>

## WireGuard

WireGuard is an extremely simple yet fast and modern VPN that utilizes state-of-the-art cryptography. It aims to be faster, simpler, leaner and more useful than IPsec, while avoiding the massive headache.

![WireGuard logo](../assets/images/dietpi-software-vpn-wireguard.svg){: width="300" height="53" loading="lazy"}

When installing using `dietpi-software`, you can choose whether to install WireGuard as VPN server or client.

=== "Installing as VPN server"

#### General

You are asked to enter your public IP/domain and the port on which the VPN server should be available. Remember to open/forward the port (UDP) through NAT on your router.

During installation, a client configuration file will be automatically created as well at:

`/etc/wireguard/wg0-client.conf`.

Configure the client configuration to your needs, it contains some informational comments. By default it will pass all clients network traffic through the VPN tunnel, including DNS requests which will be resolved by the servers DNS resolver.

If you e.g. want to use the servers Pi-hole instance on the client only, but keep all other traffic outside the VPN tunnel, you would edit the following values:

- `DNS = 192.168.0.100`

- `AllowedIPs = 192.168.0.100/32` (where the IP needs to match your DietPi servers local IP)

If your client is another Linux machine with iptables installed, you can uncomment the two kill switch lines to have all network traffic automatically disabled, when VPN connection is lost.

If your client is a mobile phone with WireGuard app installed, you can simply apply the config by printing a QR code onto the servers terminal via:

`grep -v '^#' /etc/wireguard/wg0-client.conf | qrencode -t ansiutf8`.

To allow VPN clients accessing your local Pi-hole instance, you need to allow DNS requests from all network interfaces: `pihole -a -i local`.

#### Adding multiple clients

Navigate to the servers WireGuard configuration directory: `cd /etc/wireguard`.

Create a second client key pair:

```sh

umask 0077

wg genkey > client2\_private.key

wg pubkey < client2\_private.key > client2\_public.key

umask 0022

```

Clone and configure the client config:

```sh

cp -a wg0-client.conf wg0-client2.conf

G\_CONFIG\_INJECT 'Address = ' 'Address = 10.9.0.3/24' wg0-client2.conf

G\_CONFIG\_INJECT 'PrivateKey = ' "PrivateKey = $(<client2\_private.key)" wg0-client2.conf

```

Configure `wg0.conf` (server config) so the last lines match:

```

[Peer]

PublicKey = <paste content of client2\_public.key here>

AllowedIPs = 10.9.0.3/32

```

Restart the VPN server (`systemctl restart wg-quick@wg0`) and apply `wg0-client2.conf` to your second VPN client as you did for the first before.

Repeat similar for third, fourth, ... VPN client.

=== "Installing as VPN client"

Usually the VPN provider will have install instructions and ship a configuration file.

If the you want to connect to another DietPi machine, use the generated `/etc/wireguard/wg0-client.conf` as mentioned above.

If no WireGuard (auto)start instructions are included, but you require it, please do the following:

- Check for the created configuration file/interface name: `ls -Al /etc/wireguard/`

- It has a `.conf` file ending, lets assume: `wg0-client.conf`

- To start the VPN interface, run: `systemctl start wg-quick@wg0-client`

- To autostart the VPN interface on boot, run: `systemctl enable wg-quick@wg0-client`

- To disable autostart again, run: `systemctl disable wg-quick@wg0-client`

Remark: If the client config sets the DNS server via `DNS = ...` directive, assure that resolvconf is installed: `apt install resolvconf`.

=== "View logs"

Logging can be viewed with:

```sh

journalctl -u wg-quick@wg0

```

respectively

```sh

journalctl -u wg-quick@<config\_name>

```

???+ info "Kernel update"

The WireGuard kernel module needs to rebuild whenever the kernel is updated. On most devices this will be done automatically, when the kernel (+headers) is updated via APT package, which then usually triggers the module rebuild.

If you update the kernel outside of APT, via `source build` or commands like `rpi-update`, assure that matching kernel headers are installed as well and rebuild the WireGuard module via: `dpkg-reconfigure wireguard-dkms`.

\*\*\*

Website: <https://www.wireguard.com>

Wikipedia: <https://wikipedia.org/wiki/WireGuard>

YouTube video tutorial (German language): `Raspberry Pi & PiVPN mit WireGuard: Installation unter DietPi mit NoIP und AVM Fritzbox`.

<iframe src="https://www.youtube-nocookie.com/embed/yRkdzGmnvA4?rel=0" frameborder="0" allow="fullscreen" width="560" height="315" loading="lazy"></iframe>

[Return to the \*\*Optimised Software list\*\*](../../software/)