Intro

Wireguard VPN: module Linux Kernel 5.x

Gordon Buchan
gordonbuchan.com
gordon.buchan@gordonbuchan.com

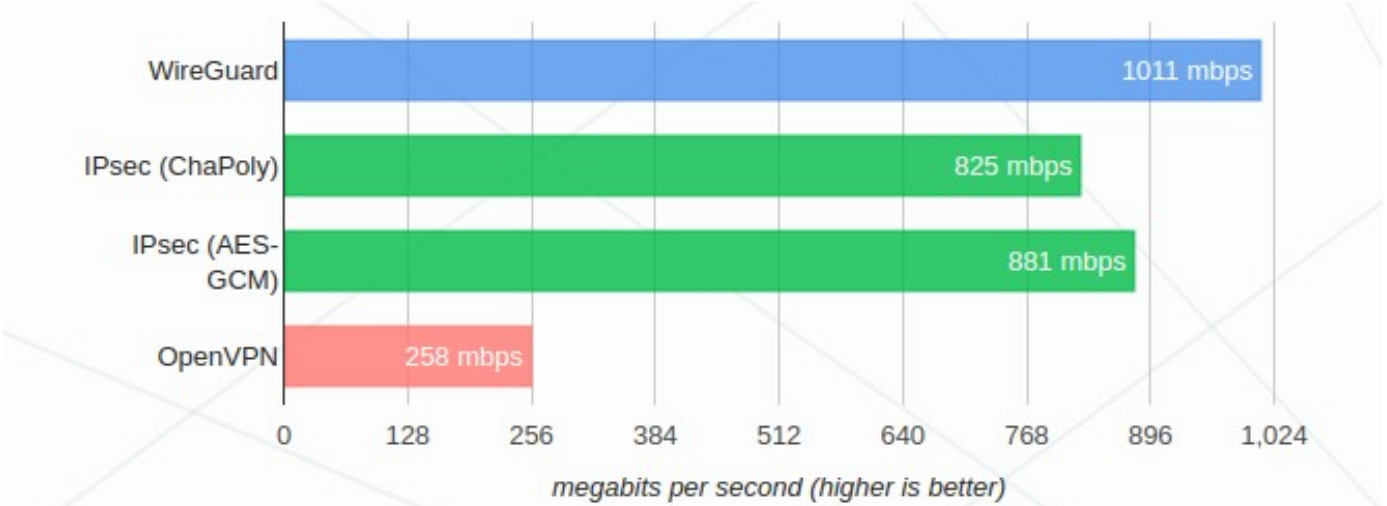
Linux Meetup Montréal
3 mars 2020

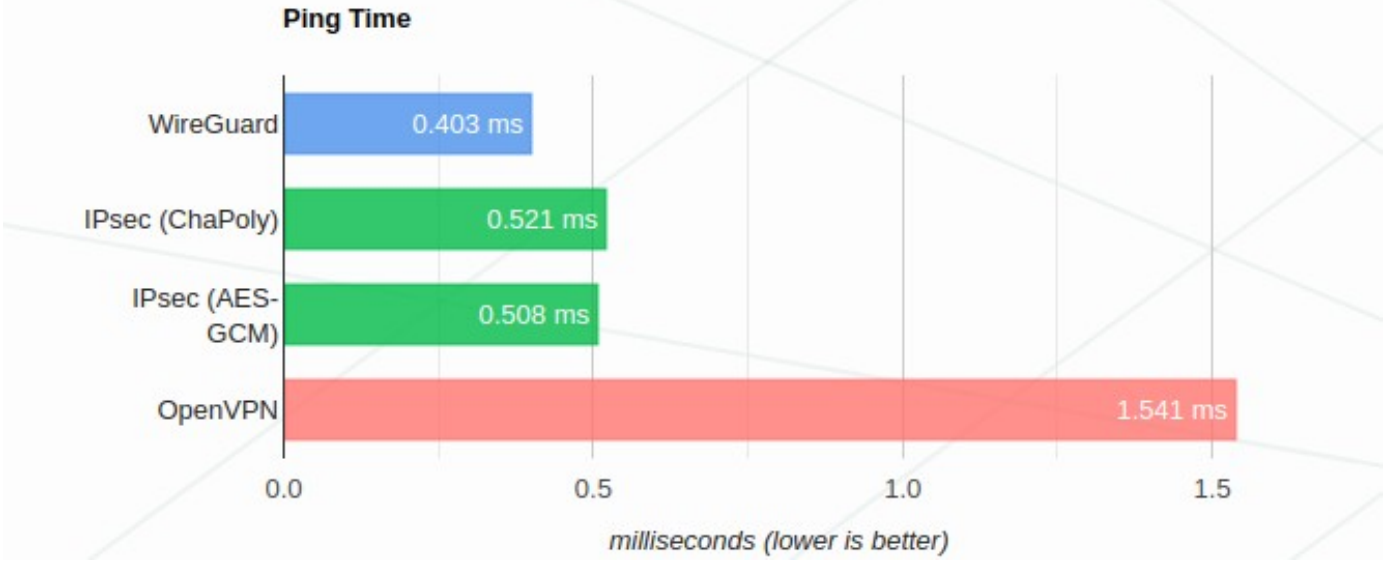
Vue d’ensemble

VPN logiciel libre
Ajouté au kernel 5.x
Plus vite que OpenVPN et Ipsec
Seulement UDP, pas TCP
Manque support pour Multi-Factor Authentication (MFA)
Client Windows fragile
Fichier de configuration: 10 lignes
Idéal pour sécuriser communications entre serveurs.
Plateformes: Linux, MacOS, Windows, Android, iOS, Docker
Licenses libres: modules GPLv2, outils BSD/Apache/GPLv2

Performance

(source: wireguard.com/performance)





Configuration

Ubuntu
timedatectl
apt install wireguard-dkms wireguard-tools

Fedora
timedatectl
dnf copr enable jdoss/wireguard
dnf install wireguard-dkms wireguard-tools

Windows
https://download.wireguard.com/windows-client/
wireguard-amd64-0.0.38.msi

Clés numériques

```
mkdir /etc/wireguard
cd /etc/wireguard
umask 077

wg genkey > /etc/wireguard/privkey
wg pubkey < /etc/wireguard/privkey > /etc/wireguard/publickey

wg genpsk > psk
```

Serveur et client Linux

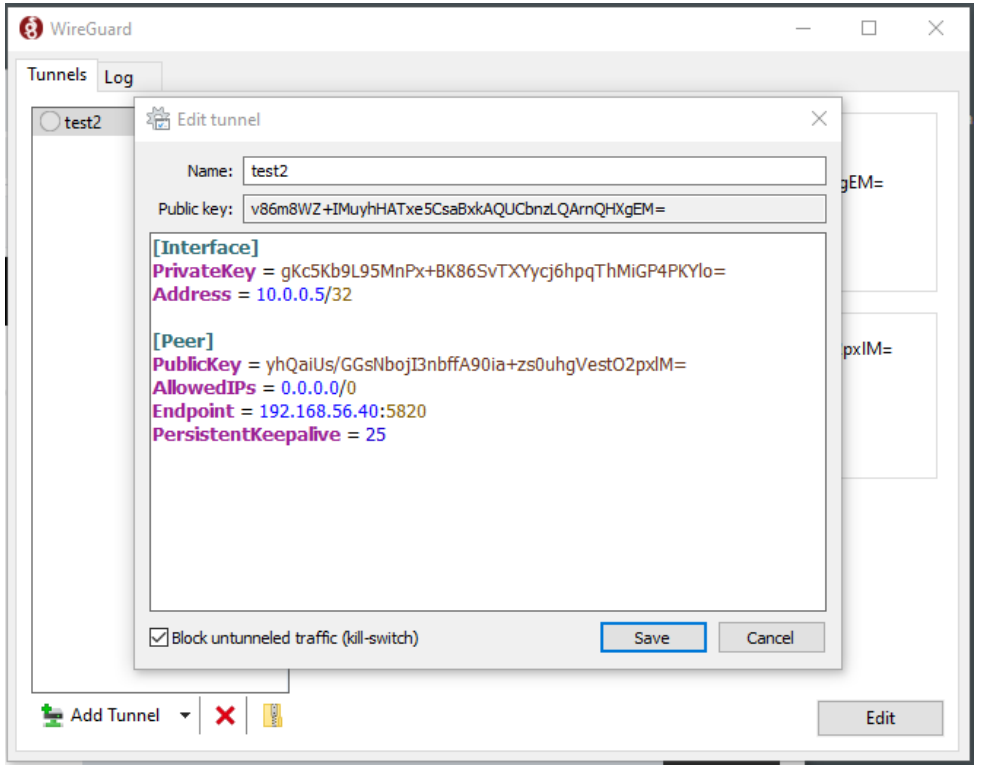
root@orlando: /etc/wireguard

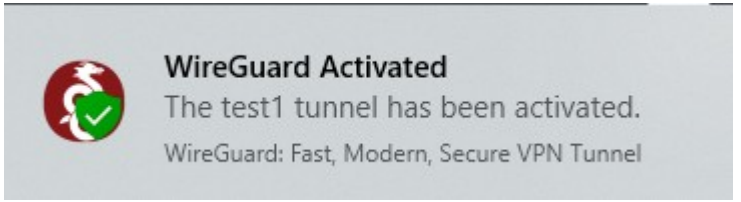
```
  GNU nano 4.3                          wg0.conf
[Interface]
ListenPort = 51871
PrivateKey = +BUYJjrCUOYrr+7dENpYoqeCJ9B3nrsadK2NI3+ltHE=
Address = 10.0.0.1/32

[Peer]
PublicKey = ADt0IhhRQz44luU/CAuktmwYACZobcd9YPWDmtNe80I=
AllowedIPs = 10.0.0.2/32
#Endpoint = 192.168.122.123:51902
PersistentKeepalive = 25

[Peer]
PublicKey = 9gAGBn/KhqnirMGFbUyPZFblSBpHzoHokWxFqB5sjCc=
AllowedIPs = 10.0.0.5/32
PersistentKeepalive = 25




                [ Read 17 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text^T To Spell  ^_ Go To Line
```

root@tampa: /etc/wireguard

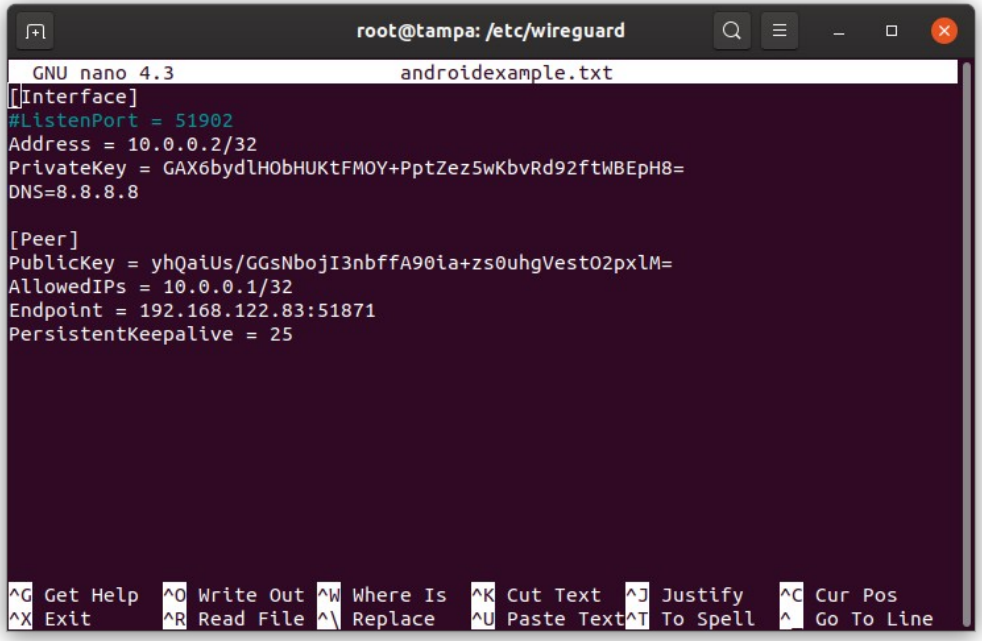
```
  GNU nano 4.3                          wg0.conf
[Interface]
#ListenPort = 51902
Address = 10.0.0.2/32
PrivateKey = GAX6bydlHObHUKtFMOY+PptZez5wKbvRd92ftWBEpH8=
DNS=8.8.8.8

[Peer]
PublicKey = yhQaiUs/GGsNbojI3nbffA90ia+zs0uhgVestO2pxlM=
AllowedIPs = 10.0.0.1/32
Endpoint = 192.168.122.83:51871
PersistentKeepalive = 25




                [ Read 11 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text^T To Spell  ^_ Go To Line
```

Client Windows



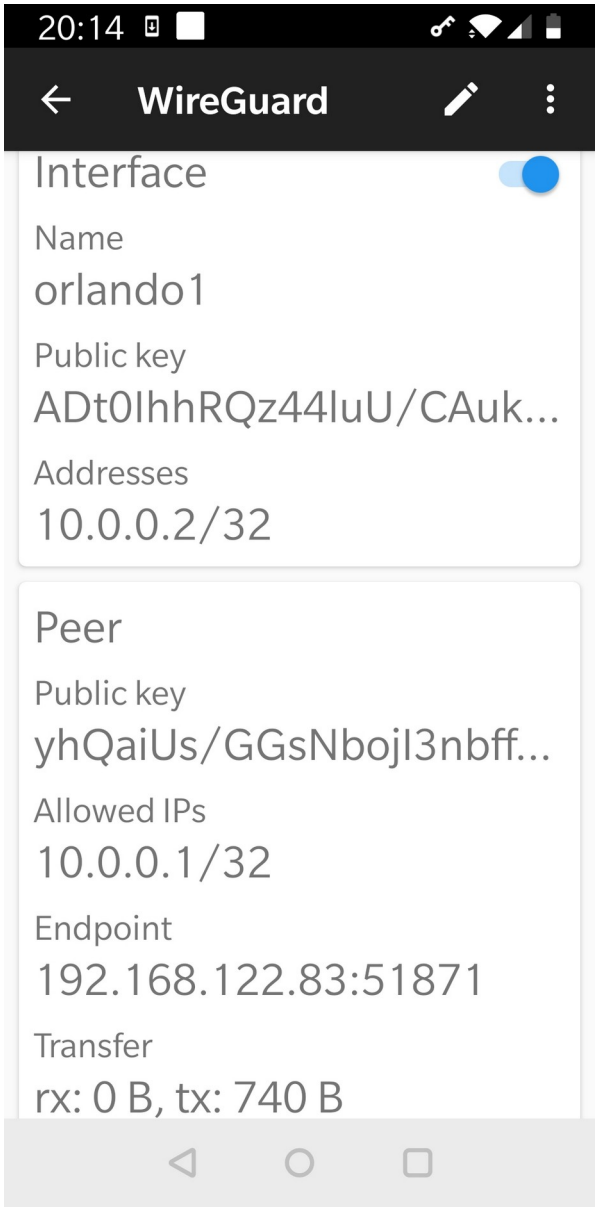


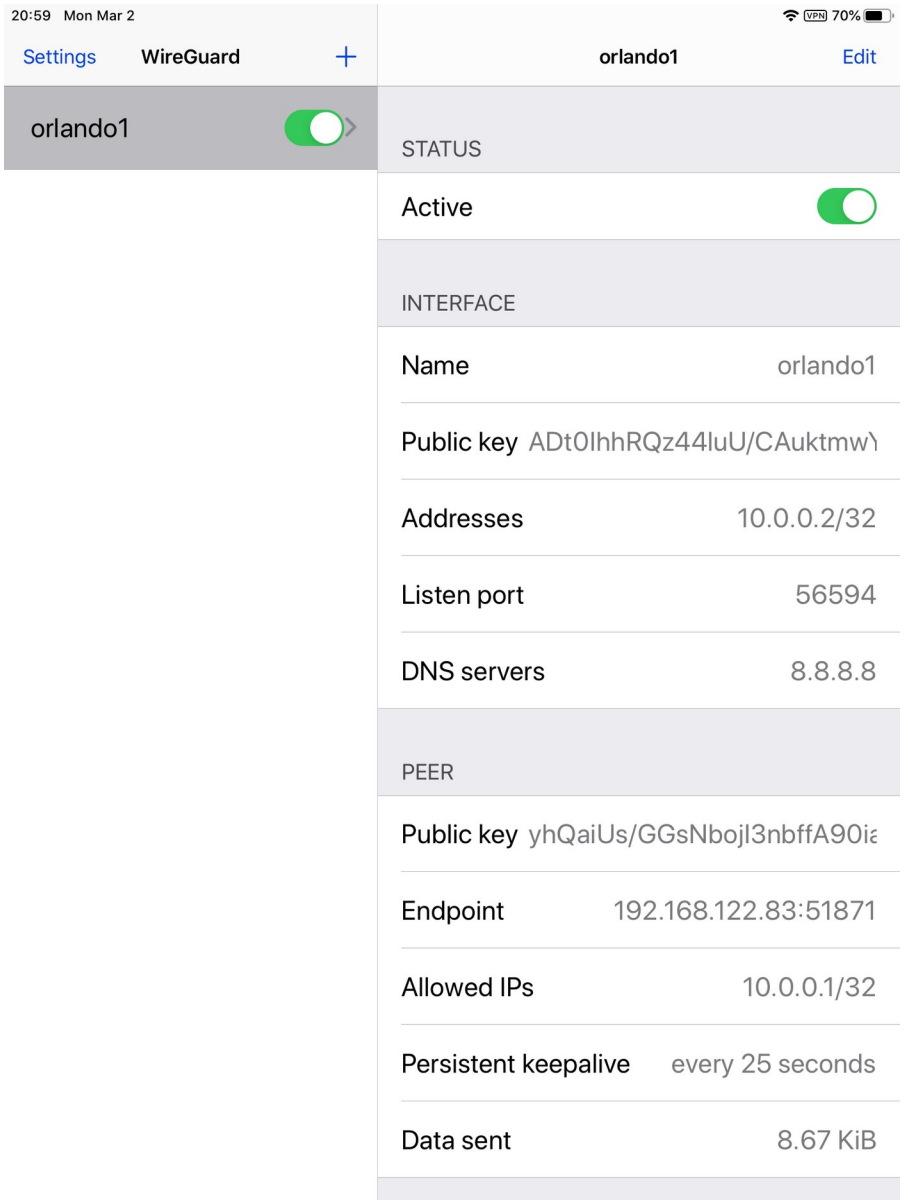
QR Code



qrencode -t ansiutf8 < androidexample.txt > qrcode.txt

Clients Android et iOS

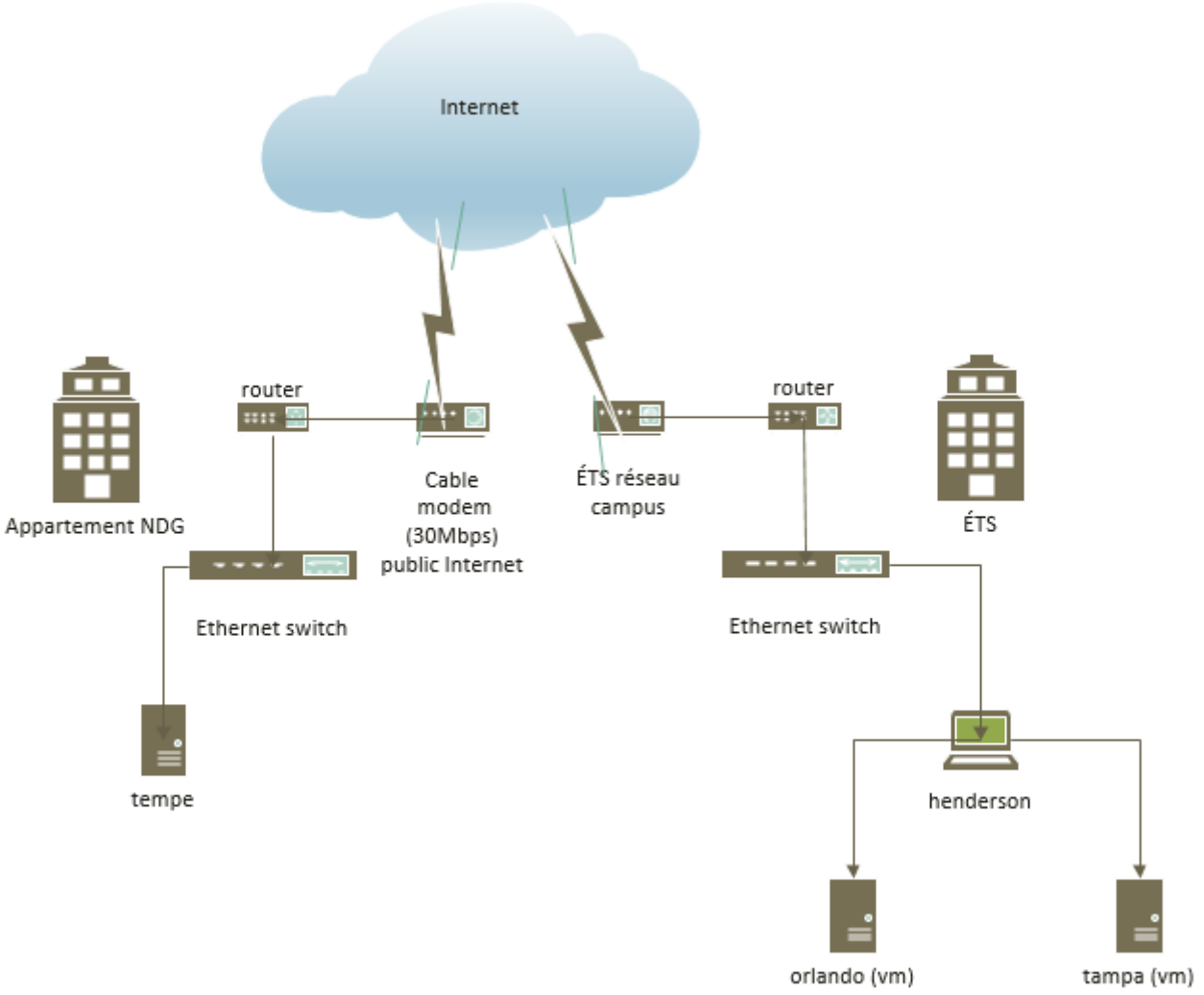




Docker (source : https://archive.ph/myI9I)

```
# Create a wireguard interface (device) named `wg1`. The kernel knows what a
# wireguard interface is as we've already installed the kernel module
ip link add dev wg1 type wireguard

# Point our new wireguard interface at the VPN server and allocate addresses
# for the interface
wg setconf wg1 /etc/wireguard/wg1.conf
ip address add 10.192.122.2/24 dev wg1

# Start the interface and add the VPN server as our DNS nameserver. This is so
# our VPN will resolve hostnames like httpbin.org or google.com.
ip link set up dev wg1
printf 'nameserver %s\n' '10.192.122.1' | resolvconf -a tun.wg1 -m 0 -x

# rp_filter is reverse path filtering. By default it will ensure that the
# source of the received packet belongs to the receiving interface. While a nice
# default, it will block data for our VPN client. By switching it to '2' we only
# drop the packet if it is not routable through any of the defined interfaced.
sysctl -w net.ipv4.conf.all.rp_filter=2

docker network create docker-vpn0 --subnet 10.193.0.0/16
ip rule add from 10.193.0.0/16 table 200
ip route add default via 10.192.122.2 table 200
docker run -ti --rm --net=docker-vpn0 appropriate/curl http://httpbin.org/ip
```

Topologie réseau



Accès au réseau local

```
echo "net.ipv4.ip_forward = 1
net.ipv6.conf.all.forwarding = 1" > /etc/sysctl.d/wg.conf

sysctl –system

iptables -t nat -A POSTROUTING -o ens3 -j MASQUERADE
ip6tables -t nat -A POSTROUTING -o ens3 -j MASQUERADE
```

Détails

Ajouté au kernel 5.x
Déjà inclu en Ubuntu 19.10
Façile à ajouter à Fedora avec DKMS
Plateformes: Linux, MacOS, Windows, Android, iOS, Docker
License: modules GPLv2, outils BSD/Apache/GPLv2

Ressources

https://www.wireguard.com/install/

https://wiki.archlinux.org/index.php/WireGuard#Specific_use-case:_VPN_server

https://utcc.utoronto.ca/~cks/space/blog/linux/MyKernelUpdateSteps

https://fedoramagazine.org/build-a-virtual-private-network-with-wireguard/

https://blog.linuxserver.io/2019/11/24/connect-an-ubuntu-client-to-opnsense-wireguard-tunnel-with-a-gui-toggle-in-gnome/

https://www.stavros.io/posts/how-to-configure-wireguard/

https://golb.hplar.ch/2019/07/wireguard-windows.html

https://blogs.gnome.org/thaller/2019/03/15/wireguard-in-networkmanager/

https://www.henrychang.ca/how-to-setup-wireguard-vpn-server-on-windows/

https://www.ckn.io/blog/2017/11/14/wireguard-vpn-typical-setup/

https://github.com/activeeos/wireguard-docker

https://nbsoftsolutions.com/blog/routing-select-docker-containers-through-wireguard-vpn

https://medium.com/@mdp/securing-docker-with-wireguard-82ad45004f4d

https://angristan.xyz/2019/01/how-to-setup-vpn-server-wireguard-nat-ipv6/