

# **M11 – Seguretat Informàtica – UF1**

**Pràctica 5 Realitzar una xarxa privada virtual (VPN) amb WireGuard**

**Nil Massó**



ASIX 2			
M11 – Seguretat Informàtica – UF1		Tipus	Individual
Cognoms, Nom:	Massó Cabaña, Nil	Curs	2022-23
Observacions:			

## Table of Contents

Enllaços:.....	1
Pràctica 5 – Realitzar una xarxa privada virtual (VPN) amb WireGuard.....	1
Exercici 1: Configuració d'un servidor WireGuard (2 punts).....	2
Exercici 2: Configuració d'un client Windows de WireGuard (2 punts).....	4
Exercici 3: Accés a un servidor WireGuard des de l'exterior (4 punts).....	5
Exercici 4: Configuració d'un client mòbil de WireGuard (2 punts).....	8

- A l'hora d'avaluar i qualificar el treball es tindran en compte els aspectes estètics, de correctesa lingüística (sintàctica i ortogràfica) a més del que s'hagi comentat al cicle formatiu sobre la redacció de documentació tècnica i manuals.

- El mòdul professional pertany a uns estudis orientats al món laboral, cosa que fa que un cop complerts els requisits mínims la nota resultant serà condicionada per la quantitat i qualitat del treball individual realitzat per cada alumne.

- **Recordeu crear una portada i un índex.**

## Enllaços:

### WireGuard

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

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

<https://enclaveinformatico.com/configurar-una-vpn-con-wireguard/>

<https://www.cyberciti.biz/faq/ubuntu-20-04-set-up-wireguard-vpn-server/>

[#Enable and start WireGuard](#)

<https://www.redeszone.net/tutoriales/vpn/wireguard-vpn-configuracion/>

<https://clouding.io/hc/es/articles/360013528839-C%C3%B3mo-instalar-WireGuard-VPN-en-Ubuntu-LucidChart>

## Pràctica 5 – Realitzar una xarxa privada virtual (VPN) amb WireGuard.

Fes els exercicis següents. Contesteu directament sota dels enunciats. Poseu-hi captures de pantalla si ho considereu necessari.

### **Recordeu a citar TOTES les fonts utilitzades**

### **Xarxes Privades Virtuals (VPN).**

Una xarxa privada virtual o VPN (Virtual Private Network) és una xarxa privada que s'estén a diferents punts remots mitjançant l'ús d'infraestructures públiques de transport (com per exemple, Internet).

Així, un usuari (una sucursal de l'organització, un teletreballador, un representant comercial...) connectat a través d'Internet a la xarxa corporativa de l'organització, establint un túnel VPN, pot funcionar com si estigués dins de l'organització a tots els efectes de connectivitat.

La característica que converteix la connexió "pública" en "privada" és el que s'anomena un túnel, terme referit a que únicament ambdós extrems són capaços de veure el que es transmet pel túnel,

convenientment xifrat i protegit de la resta d'Internet. La tecnologia de túnel xifra i encapsula els protocols de xarxa que s'utilitzen en els extrems sobre el protocol IP. D'aquesta manera podem operar com si es tractés d'un enllaç dedicat convencional, de forma transparent per a l'usuari.

**WireGuard** és una senzilla aplicació que us permetrà experimentar amb una VPN i establir túnels IP segurs. Hi ha versions per a Windows, macOS, Linux, Android, iOS,...

Teòricament proporciona un rendiment millor que fer una VPN utilitzant el protocol IPsec i OpenVPN. Treballa a nivell 3 de la capa de la OSI, per tant a nivell d'adreça IP. També facilita molt la configuració de la seguretat que es fa servir.

Per a realitzar aquesta pràctica podeu consultar aquests enllaços per a entendre la seva configuració i funcionament.

Per instal·lar-lo:

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

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

<https://enclaveinformatico.com/configurar-una-vpn-con-wireguard/>

[How to set up WireGuard VPN server on Ubuntu 20.04 - nixCraft \(cyberciti.biz\)](#)

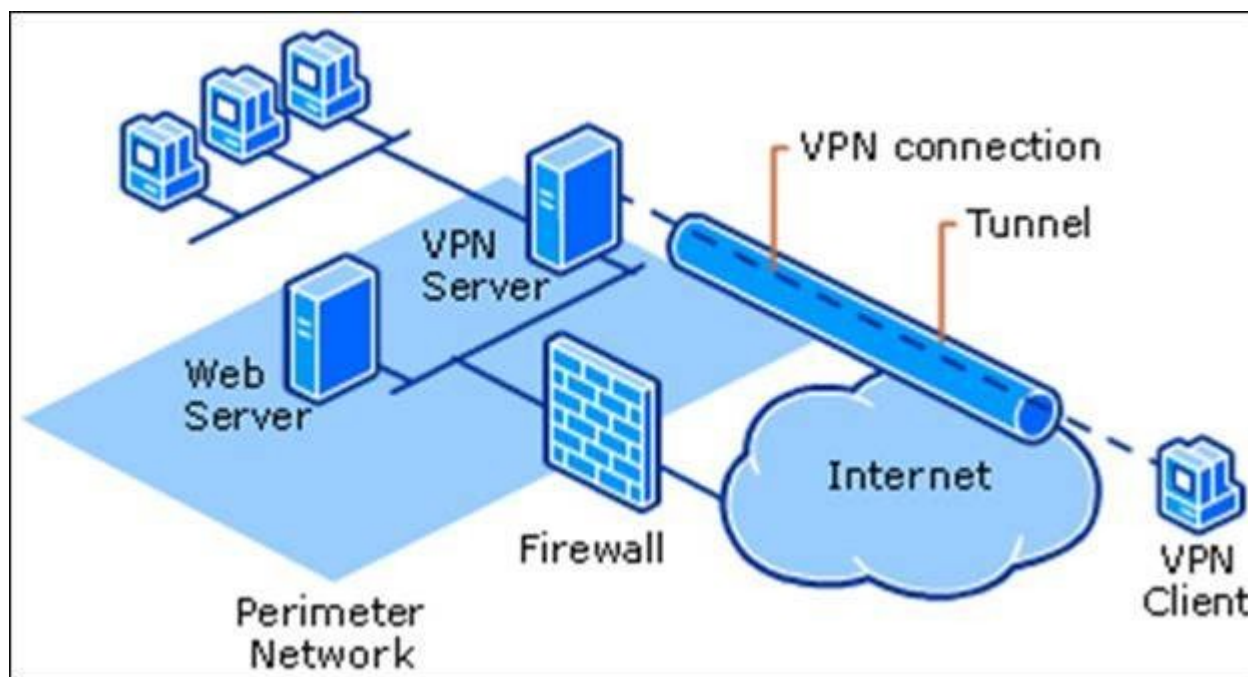
<https://www.redeszone.net/tutoriales/vpn/wireguard-vpn-configuracion/>

<https://clouding.io/hc/es/articles/360013528839-C%C3%B3mo-instalar-WireGuard-VPN-en-Ubuntu>

## **Exercici 1: Configuració d'un servidor WireGuard (2 punts)**

Realitzeu la **configuració d'un servidor WireGuard** utilitzant una màquina virtual de Linux.

Aquesta màquina seria la que es trobaria a la zona perimètrica i per tant estaria connectada tant a la xarxa pública com a la xarxa local que estem protegint. Per tant tindrà dues interfícies de xarxa: en una hi haurà una adreça "pública" i l'altra tindrà una adreça privada, d'una LAN.



La xarxa local podria ser per exemple del tipus 192.168.1.0/24

Aquesta màquina tindria tots els ports tancats excepte un port habilitat per a accedir al servei de WireGuard i establir les VPN amb les màquines externes permeses. Per defecte WireGuard utilitza el port 51820.

El que farà serà crear una interfície virtual que us donarà accés a la xarxa privada virtual (per exemple una adreça de la 192.168.2.0/24).

Per aquesta primera part de la pràctica, poseu una interfície de xarxa en mode NAT i una interfície de xarxa en bridge host-only i així es veurà amb el vostre host o amb una altra màquina virtual que també pugui estar en bridge host-only

A partir de la informació continguda als enllaços que us he proporcionat, realitzeu la configuració d'aquesta interfície virtual:

- Haureu de crear les claus privada i pública del servidor, establir la seva adreça a la VPN i el port d'escolta.
- Més tard també hi haureu d'afegir les claus públiques dels clients permesos i si cal una llista d'IPs externes permeses.

Documenteu el procés seguit:

- escriviu les comandes utilitzades.

`apt update && apt upgrade -y`

- `apt install wireguard -y`

- #Obrim el port 51820 al firewall
- apt install ufw -y
- ufw allow 51820/udp
- #Creem la carpeta per a les claus
- mkdir /etc/wireguard/keys
- #Creem les claus
- wg genkey | tee /etc/wireguard/keys/server\_privatekey | wg pubkey > /etc/wireguard/keys/server\_publickey
- #Mostrem les claus:
- cat /etc/wireguard/keys/server\_privatekey
- cat /etc/wireguard/keys/server\_publickey
- #Activem el servei
- systemctl enable wg-quick@wg0
- systemctl start wg-quick@wg0
- Escriviu la configuració utilitzada, els fitxers que heu fet servir i el contingut del fitxer de configuració del servidor (wg0.conf).

```
GNU nano 5.4 /etc/wireguard/wg0.conf *
```

- [Interface]
- Address = 192.168.2.0
- ListenPort = 51820
- PrivateKey = gJxoh8Sitfq38VGul58mI5pe5Wla+7Y0j2fpOtOQ8EY=

- Mostreu l'estat de la connexió i amb ifconfig la nova interfície.

No has demanat que configurem cap peer per tant entenc que no sha de veure cap transit de dades.

```
root@debian:~# ifconfig wg0
wg0: flags=209<UP,POINTOPOINT,RUNNING,NOARP> mtu 1420
    inet 192.168.2.0 netmask 255.255.255.255 destination 192.168.2.0
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 1000 (UNSPEC)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```


Recordeu a posar-me les captures de tot el necessari i les justificacions que creieu. Si heu d'afegir algun plànol de xarxa amb els hosts i les IPs, recordeu que podeu fer servir eines com el [LucidChart](#).

## Exercici 2: Configuració d'un client Windows de WireGuard (2 punts)

Realitzeu la configuració d'un **client WireGuard** utilitzant una màquina virtual o el vostre host de Windows. Aquesta màquina seria la que es trobaria a Internet i necessitaria entrar remotament a la xarxa protegida. En la nostra prova (per ex. mode bridge host-only) ha de tenir una adreça de la mateixa xarxa que la interfície “pública” del servidor WireGuard, ja que s'hi ha de poder connectar. El client de WireGuard li generarà les dues claus pel xifrat, la IP a la VPN i després de connectar-se al servidor (en aquest cas una IP de la xarxa que comparteixen en mode bridge host-only amb la màquina virtual del servidor), una interfície amb una IP de la xarxa privada virtual (per ex 192.168.2.0/24). A partir d'aquest moment ja es poden comunicar de manera segura els elements de la VPN utilitzant IPs d'aquesta xarxa virtual (en l'exemple seria la 192.168.2.0/24). La clau pública l'ha de conèixer el servidor VPN.

Documenteu el procés seguit i el contingut de la configuració del client.

- Mostreu el contingut de la configuració del client

 Edit tunnel

Name:	Server
Public key:	yPJGLczpSwyOIOSA81bG41ou5LnTKIL7PHy1XMiA8R0=
<b>[Interface]</b>	
PrivateKey	= gICyYYpv7uXKf83LuAvrEOIvKoPquHP0S2jUDRwLoX0=
Address	= 192.168.2.100/32
<b>[Peer]</b>	
PublicKey	= Fnj/yOR0xiDpqSZDIQistHNWSItv1yaqzc2WERxiE4=
AllowedIPs	= 0.0.0.0/0
Endpoint	= 10.2.145.43:51820

- Mostreu com heu modificat el fitxer de configuració del servidor

```
GNU nano 5.4 /etc/wireguard/wg0.conf
[Interface]
Address = 192.168.2.1
ListenPort = 51820
PrivateKey = gJxoh8Sitfq38VGuI58mI5pe5Wla+7Y0j2fp0t0Q8EY=

[Peer]
PublicKey = yPJGLczpSwy0IOsA81bG41ou5LnTKIL7PHy1XMiA8R0=
AllowedIPs = 192.168.2.100/32
```

- Mireu també què surt al Log quan us connecteu al servidor

```
2022-12-06 09:05:57.017 [TUN] [Server] Starting WireGuard/0.5.3 (Windows 10.0.19044; amd64)
2022-12-06 09:05:57.017 [TUN] [Server] Watching network interfaces
2022-12-06 09:05:57.020 [TUN] [Server] Resolving DNS names
2022-12-06 09:05:57.020 [TUN] [Server] Creating network adapter
2022-12-06 09:05:57.147 [TUN] [Server] Using existing driver 0.10
2022-12-06 09:05:57.149 [TUN] [Server] Creating adapter
2022-12-06 09:05:57.397 [TUN] [Server] Using WireGuardNT/0.10
2022-12-06 09:05:57.397 [TUN] [Server] Enabling firewall rules
2022-12-06 09:05:57.344 [TUN] [Server] Interface created
2022-12-06 09:05:57.401 [TUN] [Server] Dropping privileges
2022-12-06 09:05:57.402 [TUN] [Server] Setting interface configuration
2022-12-06 09:05:57.402 [TUN] [Server] Peer 1 created
2022-12-06 09:05:57.404 [TUN] [Server] Monitoring MTU of default v4 routes
2022-12-06 09:05:57.414 [TUN] [Server] Setting device v4 addresses
2022-12-06 09:05:57.404 [TUN] [Server] Interface up
2022-12-06 09:05:57.456 [TUN] [Server] Sending handshake initiation to peer 1 (10.2.145.43:51820)
2022-12-06 09:05:57.456 [TUN] [Server] Receiving handshake response from peer 1 (10.2.145.43:51820)
2022-12-06 09:05:57.456 [TUN] [Server] Keypair 1 created for peer 1
2022-12-06 09:05:57.473 [TUN] [Server] Monitoring MTU of default v6 routes
2022-12-06 09:05:57.473 [TUN] [Server] Setting device v6 addresses
2022-12-06 09:05:57.521 [TUN] [Server] Startup complete
2022-12-06 09:06:07.708 [TUN] [Server] Receiving keepalive packet from peer 1 (10.2.145.43:51820)
2022-12-06 09:06:23.048 [TUN] [Server] Receiving keepalive packet from peer 1 (10.2.145.43:51820)
2022-12-06 09:06:44.303 [TUN] [Server] Receiving keepalive packet from peer 1 (10.2.145.43:51820)
```

- Mostreu amb ipconfig la nova interfície al client

```
Unknown adapter Server:

Connection-specific DNS Suffix . : 
IPv4 Address. . . . . : 192.168.2.100
Subnet Mask . . . . . : 255.255.255.255
Default Gateway . . . . . : 0.0.0.0
```



- Feu un ping a la IP privada (VPN) del servidor i del servidor cap al client.

```
C:\Users\Administrator>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:
Reply from 192.168.2.1: bytes=32 time=2ms TTL=64
Reply from 192.168.2.1: bytes=32 time=2ms TTL=64
Reply from 192.168.2.1: bytes=32 time=1ms TTL=64
Reply from 192.168.2.1: bytes=32 time=1ms TTL=64

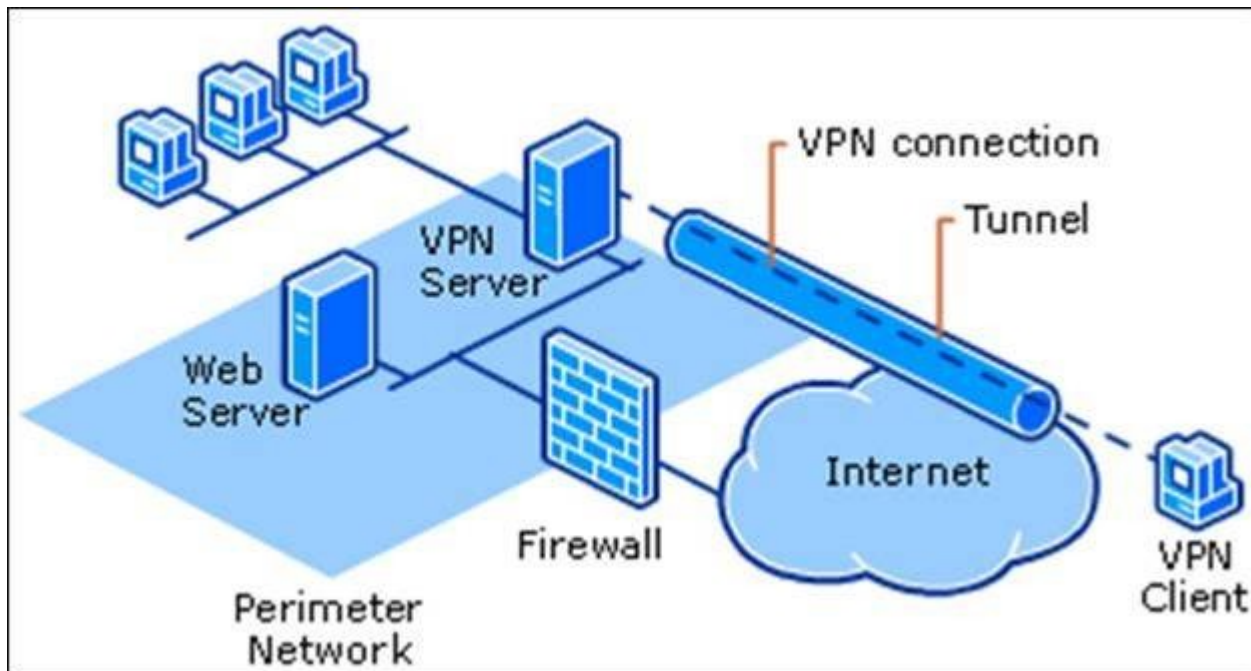
Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 2ms, Average = 1ms
```

```
root@debian:~# ping 192.168.2.100
PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.
```

No dona resposta ja que no ho he habilitat

Recordeu a posar-me les captures de tot lo necessari i les justificacions que creieu. Si heu d'afegir algun plànol de xarxa amb els hosts i les IPs, recordeu que podeu fer servir eines com el [LucidChart](#).

En aquesta part, connectareu la VPN amb la vostra xarxa interna, i configurareu els clients per a que encaminin per la VPN tot el trànsit que hagi d'anar a la xarxa interna, de manera que semblarà que el client estigui directament connectat a la xarxa interna.



### Exercici 3: Accés a un servidor WireGuard des de l'exterior (4 punts)

En una situació real, el servidor WireGuard estarà a una xarxa pública i no segura com internet (per exemple amb una adreça pública com 147.161.163.142) . Però també tindrà una altra targeta de xarxa

(NIC) amb una adreça de xarxa local, per exemple de la xarxa 192.168.1.0/24 . I a part d'això tindrà la IP de la VPN, per exemple 192.168.2.1 de la xarxa VPN 192.168.2.0/24. La vostra configuració només admetrà connexions exteriors per accedir a la IP pública pel port que hàgiu destinat a escoltar les peticions de Wireguard. La màquina no acceptarà altres peticions públiques, per tant, no es podrà accedir a la resta de la xarxa interna.

A casa vostra, les màquines no tenen una IP pública sinó que estareu darrere un router domèstic i aneu amb NAT. Haureu de configurar el vostre router de manera que faci un Port Forwarding (PAT estàtic). Les peticions que es rebin pel port que trieu, es desviaran a una adreça IP no pública de la vostra xarxa local, o sigui a la vostra màquina virtual amb Wireguard i pel port que trieu, en l'exemple que hem fet, el destí seria el 51820.

Això es configura mitjançant el router.

Firewall

Rules

Port Forwarding

DMZ

Parental Control

## Port Forwarding

Port forwarding allows remote computers to connect to a specific device within your private network.

Name	Direction	Dst. IP	Protocol	Public Port(s)	Private Port(s)	Enabled		
ElMeuWeb	wan to lan	192.168.1.223	tcpudp	54321	443	<input type="checkbox"/>		
AccesAVPN	wan to lan	192.168.1.166	tcpudp		51820	<input checked="" type="checkbox"/>		

**Enable** ☒

**Rule Name**

**Source Zone**

**Destination Zone**

**Source IP Address** ☐

**Dst. Device**

**Dst. IP Address**

**Protocol**

**Source Port(s)**   
A port number or range of the form startport:endport

**Destination Port(s)**   
A port number or range of the form startport:endport

**NAT Loopback** ☒

El meu servidor està a la 192.168.1.166 de la meua LAN i escolta pel port 51820. Aquesta màquina virtual tindria tots els ports tancats excepte un port habilitat per a accedir al servei de WireGuard i establir les VPN amb les màquines externes permeses. Per defecte WireGuard utilitza el port 51820.

Les peticions per un port XXX a la meua IP pública, el meu router les envia a al port 192.168.1.166:51820 Això vol dir que puc connectar-me a la meua VPN des de qualsevol lloc del món si se la meua IP pública i el port que tinc obert.

Aquí no tenim una connexió externa. Simularem aquesta topologia de xarxa afegint una xarxa interna a la màquina virtual del servidor Wireguard i allà posant-hi una altra màquina virtual, així simularem una xarxa interna no accessible des del vostre host, per exemple la 172.16.0.0/16 . Des del vostre host heu de poder accedir al servidor Wireguard (comparteixen xarxa bridge only-host), però no podeu arribar a l'altra màquina de la xarxa interna. Quan el configureu com a client de la VPN, llavors hi tindreu accés a través de la IP de la VPN, i el wireguard us enrutarà cap a la xarxa interna

Servidor VPN amb Wireguard :

- interfície de xarxa bridge only host
- interfície de xarxa interna

Client de xarxa local :

- interfície de xarxa interna

El vostre host:

- xarxa bridge only host i client VPN Wireguard

En primer lloc cal habilitar l'enrutament al servidor de Wireguard. Necessiteu unes línies com aquestes al fitxer de configuració del servidor.

*PostUp = iptables -A FORWARD -i %i -j ACCEPT; iptables -A FORWARD -o %i -j ACCEPT; iptables -t nat -A POSTROUTING -o **eth0** -j MASQUERADE*

*PostDown = iptables -D FORWARD -i %i -j ACCEPT; iptables -D FORWARD -o %i -j ACCEPT; iptables -t nat -D POSTROUTING -o **eth0** -j MASQUERADE*

On diu **eth0** hi heu de posar la interfície que us connecta amb la xarxa interna, en el meu exemple la interfície que té una adreça IP de la xarxa 172.16.0.0/16. A la pròxima UF veurem què volen dir aquestes línies.

També cal activar l'enrutament al servidor **echo 1 > /proc/sys/net/ipv4/ip\_forward**

Si voleu que l'enrutament sigui permanent, cal editar */etc/sysctl.conf* i afegir-hi **net.ipv4.ip\_forward = 1**

Amb **sysctl -p** els canvis són efectius al moment.

Heu de configurar el client VPN de la màquina de la xarxa interna per a que capturi tot el tràfic que volem enviar a la VPN i a la xarxa interna (recordeu, teòricament no accessible: 172.16.0.0/16 des del vostre host). En el nostre cas hauria de capturar el tràfic cap a 192.168.2.0/24 i 172.16.0.0/16

Documenteu el procés seguit i el contingut de la configuració del servidor i el client de la xarxa interna. Aprofito que personalment tinc muntat aquest sistema, per tant, no el tornaré a fer, et mostraré les configuracions i comprovacions que demanes però tapant els meus valors, ja que no vull que entrin a la meva xarxa.

Explicació: Màquina virtual Windows amb wireguard connectada amb nat al portàtil, aquest connectat per USB Thetering al mobil, aquest connectat a dades mobils (ja que els \*\*\*\*\* de la meua companyia de internet no em volen activar el NAT loopback), el tunel passa per internet fins arribar a casa meua on un altre ordinador rep els paquets de wireguard desde el port habilitat al router.

- Mostreu el contingut de la configuració del servidor

```

GNU nano 5.4 /etc/wireguard/wg0.conf
# define the WireGuard service
[Interface]
Address = 192.168.111.1
# contents of file wg-private.key that was recently created
PrivateKey = [REDACTED]

# UDP service port; 51820 is a common choice for WireGuard
ListenPort = [REDACTED]

PostUp = iptables -A FORWARD -i wg0 -j ACCEPT; iptables -t nat -A POSTROUTING -o enp0s14 -j MASQUERADE
PostDown = iptables -D FORWARD -i wg0 -j ACCEPT; iptables -t nat -D POSTROUTING -o enp0s14 -j MASQUERADE

```

```

# define the remote WireGuard interface (client)
[Peer]

# contents of file wg-public.key on the WireGuard client
PublicKey = LI [REDACTED]

# the IP address of the client on the WireGuard network
AllowedIPs = 192.168.111.101/32

```

- Mostreu el contingut de la configuració del client. Que quedi clar el tràfic que captureu.

Name:

Public key:

```

[Interface]
PrivateKey = [REDACTED]
Address = 192.168.111.101/32

[Peer]
PublicKey = [REDACTED]
AllowedIPs = 0.0.0.0/0
Endpoint = [REDACTED]:51820
PersistentKeepalive = 20

```

- Mireu també què surt al Log quan us connecteu al servidor

2022-12-06 09:38:16.873	[TUN] [MyServer] Starting WireGuard/0.5.3 (Windows 10.0.19044; amd64)
2022-12-06 09:38:16.873	[TUN] [MyServer] Watching network interfaces
2022-12-06 09:38:16.878	[TUN] [MyServer] Resolving DNS names
2022-12-06 09:38:16.878	[TUN] [MyServer] Creating network adapter
2022-12-06 09:38:16.977	[TUN] [MyServer] Using existing driver 0.10
2022-12-06 09:38:16.980	[TUN] [MyServer] Creating adapter
2022-12-06 09:38:17.180	[TUN] [MyServer] Using WireGuardNT/0.10
2022-12-06 09:38:17.180	[TUN] [MyServer] Enabling firewall rules
2022-12-06 09:38:17.108	[TUN] [MyServer] Interface created
2022-12-06 09:38:17.193	[TUN] [MyServer] Dropping privileges
2022-12-06 09:38:17.194	[TUN] [MyServer] Setting interface configuration
2022-12-06 09:38:17.194	[TUN] [MyServer] Peer 1 created
2022-12-06 09:38:17.195	[TUN] [MyServer] Monitoring MTU of default v4 routes
2022-12-06 09:38:17.195	[TUN] [MyServer] Setting device v4 addresses
2022-12-06 09:38:17.195	[TUN] [MyServer] Sending keepalive packet to peer 1 [REDACTED]
2022-12-06 09:38:17.195	[TUN] [MyServer] Sending handshake initiation to peer 1 [REDACTED]
2022-12-06 09:38:17.195	[TUN] [MyServer] Interface up
2022-12-06 09:38:17.237	[TUN] [MyServer] Monitoring MTU of default v6 routes
2022-12-06 09:38:17.238	[TUN] [MyServer] Setting device v6 addresses
2022-12-06 09:38:17.255	[TUN] [MyServer] Startup complete
2022-12-06 09:38:17.298	[TUN] [MyServer] Receiving handshake response from peer 1 ([REDACTED])
2022-12-06 09:38:17.299	[TUN] [MyServer] Keypair 1 created for peer 1
2022-12-06 09:38:47.154	[TUN] [MyServer] Sending keepalive packet to peer 1 [REDACTED]

- Mostreu amb ipconfig la nova interfície al client

```
Unknown adapter MyServer:

Connection-specific DNS Suffix . : 
IPv4 Address. . . . . : 192.168.111.101
Subnet Mask . . . . . : 255.255.255.255
Default Gateway . . . . . : 0.0.0.0
```

- Feu un ping a la IP privada (VPN) del servidor i del servidor cap al client.

```
C:\Users\Administrator>ping 192.168.111.1

Pinging 192.168.111.1 with 32 bytes of data:
Reply from 192.168.111.1: bytes=32 time=61ms TTL=64
Reply from 192.168.111.1: bytes=32 time=83ms TTL=64
Reply from 192.168.111.1: bytes=32 time=70ms TTL=64
```

Igual que abans tinc configurat el client perquè no respongui peticions de ping

```
root@croqueta:~# ping 192.168.111.101
PING 192.168.111.101 (192.168.111.101) 56(84) bytes of data.
```

- Feu un ping des del vostre host a la IP del client a la xarxa interna

```
C:\Users\Administrator>ping 192.168.1. [redacted]

Pinging 192.168.1. [redacted] with 32 bytes of data:
Reply from 192.168.1. [redacted]: bytes=32 time=77ms TTL=64
Reply from 192.168.1. [redacted]: bytes=32 time=63ms TTL=64
Reply from 192.168.1. [redacted]: bytes=32 time=102ms TTL=64
Reply from 192.168.1. [redacted]: bytes=32 time=110ms TTL=64

Ping statistics for 192.168.1. [redacted]:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 63ms, Maximum = 110ms, Average = 88ms
```

## Exercici 4: Configuració d'un client mòbil de WireGuard (2 punts)

Realitzeu la configuració d'un client [WireGuard](#) utilitzant un mòbil o una tablet que estigui connectada a la mateixa xarxa que el servidor VPN. Poseu el servidor en mode bridge, per exemple. Quan es crea el client, es creen les seves claus privada i pública. La clau pública l'ha de conèixer el servidor VPN i l'ha de tenir al fitxer de configuració.

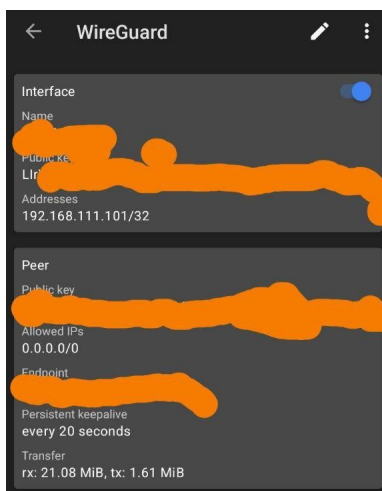
El client de [WireGuard](#) li generarà, després de connectar-se a la IP del servidor, una interfície amb una IP de la xarxa privada virtual (per ex 192.168.2.0/24). A partir d'aquest moment ja es poden comunicar de manera segura els elements de la VPN. Creeu una connexió a una IP de la VPN per a comprovar que funciona bé.

També volem accedir a la màquina que està a la xarxa interna, per tant heu de configurar el client per a que capturi tot el tràfic que volem enviar a la VPN i a la xarxa interna (teòricament no accessible: 172.16.0.0/16). En el meu cas hauria de capturar el tràfic cap a 192.168.2.0/24 i 172.16.0.0/16

Documenteu el procés seguit i el contingut de la configuració del client.

Al igual que a l'exercici anterior també tinc configurat el mòbil, per anar més ràpid, el que he fet a l'exercici anterior es exportar el zip del meu tunel del mòbil i posarlo a la màquina virtual de tal manera que ara al ferho desde mòbil la configuració no canvia, simplement les captures desde el mòbil. Al igual que abans el mòbil estarà amb dades per a poder accedir.

- Mostreu el contingut de la configuració del client



- Mostreu com heu modificat el fitxer de configuració del servidor

```
GNU nano 5.4 /etc/wireguard/wg0.conf
# define the WireGuard service
[Interface]
Address = 192.168.111.1
# contents of file wg-private.key that was recently created
PrivateKey = 
# UDP service port; 51820 is a common choice for WireGuard
ListenPort = 
PostUp = iptables -A FORWARD -i wg0 -j ACCEPT; iptables -t nat -A POSTROUTING -o enp0s14 -j MASQUERADE
PostDown = iptables -D FORWARD -i wg0 -j ACCEPT; iptables -t nat -D POSTROUTING -o enp0s14 -j MASQUERADE
# define the remote WireGuard interface (client)
[Peer]
# contents of file wg-public.key on the WireGuard client
PublicKey = LI
# the IP address of the client on the WireGuard network
AllowedIPs = 192.168.111.101/32
```

- Mireu també què surt al Log quan us connecteu al servidor

```
12-06 20:40:22.668 28829 28876 I WireGuard/GoBackend: Bringing tunnel Servidor UP

12-06 20:40:22.691 28829 28876 D WireGuard/GoBackend: Go backend ef5c587

12-06 20:40:22.691 28829 28876 D WireGuard/GoBackend/Servidor: Attaching to interface tun0

12-06 20:40:22.691 28829 28876 D WireGuard/GoBackend/Servidor: UAPI: Updating private key

12-06 20:40:22.691 28829 28879 D WireGuard/GoBackend/Servidor: Routine: handshake worker 1 - started

12-06 20:40:22.691 28829 31111 D WireGuard/GoBackend/Servidor: Routine: encryption worker 3 - started

12-06 20:40:22.692 28829 31111 D WireGuard/GoBackend/Servidor: Routine: encryption worker 2 - started

12-06 20:40:22.692 28829 31111 D WireGuard/GoBackend/Servidor: Routine: decryption worker 2 - started

12-06 20:40:22.692 28829 28879 D WireGuard/GoBackend/Servidor: Routine: encryption worker 1 - started

12-06 20:40:22.692 28829 31111 D WireGuard/GoBackend/Servidor: Routine: handshake worker 2 - started

12-06 20:40:22.692 28829 28879 D WireGuard/GoBackend/Servidor: Routine: decryption worker 1 - started

12-06 20:40:22.692 28829 31111 D WireGuard/GoBackend/Servidor: Routine: handshake worker 3 - started

12-06 20:40:22.692 28829 28879 D WireGuard/GoBackend/Servidor: Routine: decryption worker 3 - started

12-06 20:40:22.692 28829 31111 D WireGuard/GoBackend/Servidor: Routine: TUN reader - started
```



```

12-06 20:40:22.692 28829 28879 D WireGuard/GoBackend/Servidor: Routine: event worker - started

12-06 20:40:22.692 28829 28876 D WireGuard/GoBackend/Servidor: UAPI: Removing all peers

12-06 20:40:22.692 28829 28876 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - UAPI: Created

12-06 20:40:22.692 28829 28876 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - UAPI: Adding allowedip

12-06 20:40:22.692 28829 28876 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - UAPI: Updating endpoint

12-06 20:40:22.692 28829 28876 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - UAPI: Updating persistent keepalive interval

12-06 20:40:22.693 28829 28876 D WireGuard/GoBackend/Servidor: UDP bind has been updated

12-06 20:40:22.693 28829 28876 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - Starting

12-06 20:40:22.693 28829 31111 D WireGuard/GoBackend/Servidor: Routine: receive incoming v4 - started

12-06 20:40:22.693 28829 28876 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - Sending keepalive packet

12-06 20:40:22.693 28829 28876 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - Sending handshake initiation

12-06 20:40:22.693 28829 31111 D WireGuard/GoBackend/Servidor: Routine: receive incoming v6 - started

12-06 20:40:22.693 28829 31111 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - Routine: sequential sender - started

12-06 20:40:22.693 28829 31111 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - Routine: sequential receiver - started

12-06 20:40:22.694 28829 28876 D WireGuard/GoBackend/Servidor: Interface state was Down, requested Up, now Up

12-06 20:40:22.694 28829 28876 D WireGuard/GoBackend/Servidor: Device started

12-06 20:40:22.814 28829 31111 D WireGuard/GoBackend/Servidor: peer(sBf3...qSUM) - Received handshake response

```

- Feu un ping a la IP privada (VPN) del servidor i del servidor cap al client.

```

~ $ ping 192.168.111.1
PING 192.168.111.1 (192.168.111.1) 56(84) bytes of data.
64 bytes from 192.168.111.1: icmp_seq=1 ttl=64 time=313
ms
64 bytes from 192.168.111.1: icmp_seq=2 ttl=64 time=335

```

Igual que abans tinc configurat el client perquè no respongui peticions de ping

```

root@croqueta:~# ping 192.168.111.101
PING 192.168.111.101 (192.168.111.101) 56(84) bytes of data.

```

- Feu un ping des del client mòbil a la IP del client a la xarxa interna

```
~ $ ping 192.168.1.████  
PING 192.168.1.████ (192.168.1.████) 56(84) bytes of data  
64 bytes from 192.168.1.████: icmp_seq=1 ttl=64 time=269  
ms
```

Recordeu a posar-me les captures de tot lo necessari i les justificacions que creieu. Si heu d'afegir algun plànol de xarxa amb els hosts i les IPs, recordeu que podeu fer servir eines com el [LucidChart](#).