

## **MEMORIA FINAL DE PROYECTO**

***IMPLANTAR SERVIDOR VPN CON 'WIREGUARD' EN UN ENTORNO EMPRESARIAL.***



**CICLO FORMATIVO DE GRADO SUPERIOR:**

***ADMINISTRACIÓN DE SISTEMAS INFORMÁTICOS EN RED***

**AUTOR:**

***MIGUEL ÁNGEL RIVERA ROLDÁN***

**TUTORA:**

***TERESA GONZÁLEZ GONZÁLEZ***

**COORDINADORA:**

***MARÍA JOSÉ VILLAR RUIBAL***

**CURSO:**

***2019 – 2020***

***I.E.S CLARA DEL REY***

# ÍNDICE

1. INTRODUCCIÓN.....	3
2. ALCANCE DEL PROYECTO.....	4
2.1 OBJETIVO.....	5
3. ANÁLISIS DE SOLUCIÓN ESCOGIDA.....	5
3.1 ¿QUÉ ES WIREGUARD?.....	5
3.2 PLANIFICACIÓN DE TAREAS.....	7
4. TOPOLOGÍA DE RED UTILIZADA.....	8
5. IMPLEMENTACIÓN DEL DISEÑO.....	9
5.1 CREACIÓN DE MÁQUINAS VIRTUALES.....	10
5.1.1 CONFIGURACIÓN DE MÁQUINAS VIRTUALES.....	11
1- Router Empresa.....	11
2- Cortafuegos iptables.....	13
3 - Server_DHCP_DNS.....	16
4 – Server_LDAP_DNS maestro.....	19
5 - Server_Correo_SFTP.....	24
6 - Server_LAMP.....	33
7 - Router Casa.....	38
8 - Server_VPN_WireGuard.....	38
9 - Laptop_Casa.....	43
5.1.2 CONEXIÓN DEL CLIENTE CON EL SERVIDOR VPN.....	47
6. CONCLUSIONES.....	53
6.1 LIMITACIONES ENCONTRADAS.....	53
7. ÍNDICE DE FIGURAS.....	54
8. GLOSARIO.....	57
9. BIBLIOGRAFÍA.....	59
9. ANEXO.....	60

## 1. INTRODUCCIÓN

### **Descripción:**

**Instalación y configuración de servidores en Linux para una empresa, con la implementación tanto en el servidor como en el cliente del protocolo VPN “WireGuard” en máquinas virtuales.**

- La idea de este proyecto surge por las noticias de ciberseguridad que se han ido conociendo desde que comenzó la pandemia del Covid-19. Todo este tiempo y debido al estado en el que se encuentra el mundo, millones de personas se han visto obligadas a realizar telestudios o a teletrabajar con sus consecuentes riesgos, cuando antes no tenían necesidad de hacerlo .



Fuente: ViktorHanacek.cz

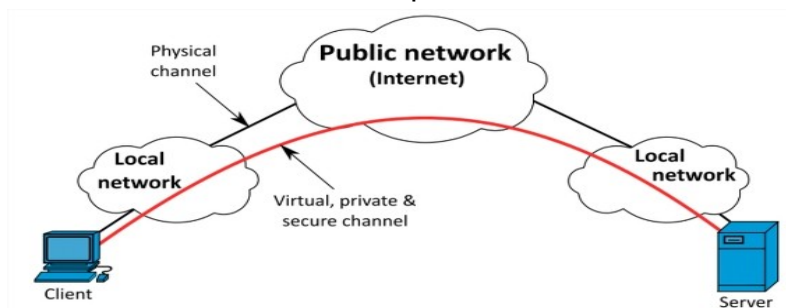
- Aunque mayoritariamente los ciberataques están automatizados y no siempre van dirigidos a una persona en concreto, existiendo miles de IPs maliciosas que buscan algún resquicio en la seguridad de nuestros sistemas y con ello obtener algún tipo de beneficio. Probablemente las empresas es donde mayor daño pueden hacer y mayor beneficio económico pueden sacar estos atacantes.
- Cada día que pasa es más importante la ciberseguridad en un entorno empresarial, evitando en la mayor medida la pérdida o el robo de información relevante de la empresa y de los propios clientes.

## 2. ALCANCE DEL PROYECTO



Fuente : [INCIBE](#)

- En el proyecto se simulará el entorno de una empresa (PYME) con los diferentes servicios básicos necesarios con servidores implantados en Linux, a los cuales se añadirá un servidor con WireGuard y un cliente se conectará desde casa.
- Una de las formas más efectivas de evitar la pérdida o el robo de información aunque no la única es utilizando una **VPN<sup>1</sup> (Virtual Private Network)**, que es una conexión encriptada a través de Internet desde un dispositivo a una red. La conexión cifrada ayuda a garantizar que los datos confidenciales se transmitan de forma segura. Evita que personas no autorizadas interpreten nuestra información y permite al usuario realizar el trabajo de forma remota. La tecnología VPN es ampliamente utilizada en entornos corporativos.



Fuente : [INCIBE](#)

1 <https://www.cisco.com/c/en/us/products/security/vpn-endpoint-security-clients/what-is-vpn.html>

## 2.1 OBJETIVO

- Con este proyecto se pretende demostrar la funcionalidad, versatilidad y posibilidades de WireGuard en un entorno empresarial. Así como afianzar algunos de los conocimientos obtenidos durante el ciclo de ASIR.

## 3. ANÁLISIS DE SOLUCIÓN ESCOGIDA



Fuente : [WireGuard](https://www.wireguard.com/)

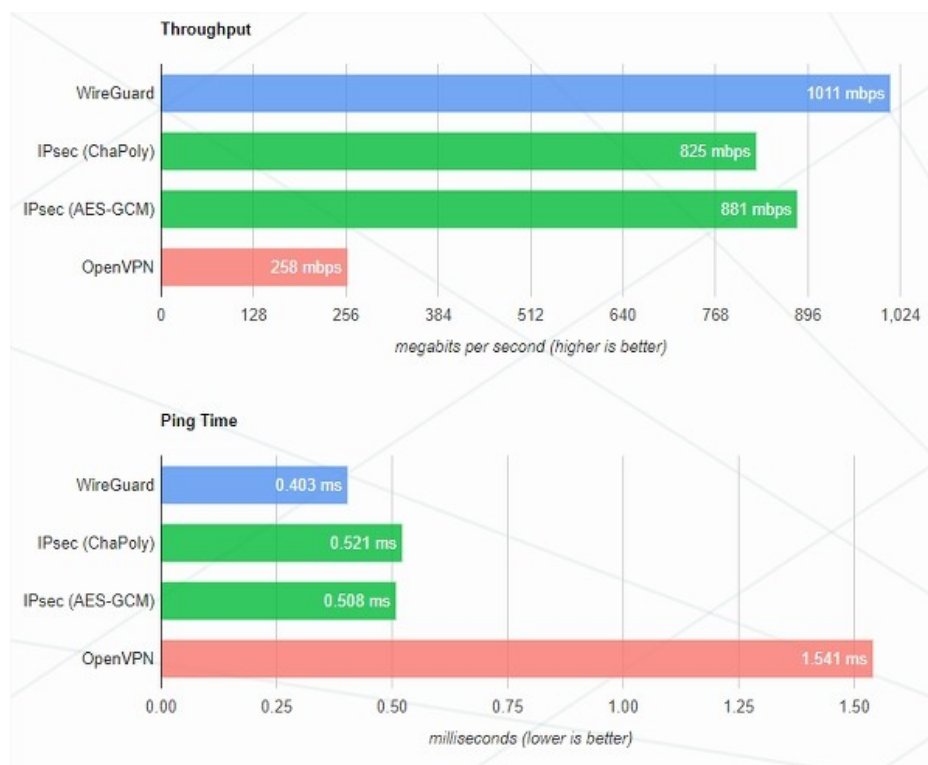
### 3.1 ¿QUÉ ES WIREGUARD?

- De todos los protocolos VPN que existen se utilizará “ **WireGuard<sup>2</sup>** “ por ser de código abierto, por su utilización en todos los sistemas operativos basados en Unix (Android, iOS, macOS, Linux) de forma nativa, además de otras características que hacen a este protocolo muy interesante. Se trata de una aplicación de software libre y un protocolo de comunicación que implementa técnicas de red privada virtual (VPN) para crear conexiones seguras punto a punto en configuraciones enrutadas o puenteadas, con técnicas de encriptación modernas como son : **Noise\_protocol\_framework, Curve25519, ChaCha20, Poly1305, BLAKE2, SipHas24, HKDF**
- *Jason A. Donenfeld* el fundador de [Edge Security](https://www.edge-security.com/), creó este código. Se dio cuenta de que los mismos métodos utilizados para infiltrarse en una red eran los que la protegían. WireGuard contiene muchas características ocultas ya integradas y proporciona un túnel VPN fiable que supera las tecnologías obsoletas que usamos. Admite la capa 3 para IPv4 en IPv6 y puede encapsular IPv4 en IPv6 y viceversa. **WireGuard parece ser el futuro de los protocolos VPN.**

---

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

- Este protocolo que lleva desde el año 2016 tratándose de forma experimental por los desarrolladores, ha llegado para quedarse. Prueba de ello es que el responsable de la pila de red de Linux, David Miller aceptó la inclusión de este protocolo VPN en el nuevo kernel 5.6 en Diciembre de 2019 y en Enero de 2020 Linux Torvalds lo añadió al árbol de dicho kernel.
- Se dice que es el protocolo más fácil de usar, seguro y efectivo disponible en la actualidad. Este protocolo ofrece un salto de rendimiento superior al que ofrece otros protocolos como [OpenVPN](#), debido a que se ejecuta como un módulo dentro del kernel y no como ocurre con otros protocolos que lo hace como una aplicación. Esto hace que los servicios criptográficos y sus procesos de cifrado y descifrado se ejecuten en WireGuard mucho más rápido.



*Figura 1: Comparativa protocolos*

Fuente : <https://www.wireguard.com/performance/>



### 3.2 PLANIFICACIÓN DE TAREAS

Proyecto de Implementación de servidor VPN con Wireguard			
Lista de tareas	Fecha Inicio	Duración	Fecha Final
<i>Analisis e Investigación</i>	27/04/2020	12	09/05/2020
<i>Topología de Red</i>	07/05/2020	12	19/05/2020
<i>Creación de Máquinas Virtuales</i>	20/05/2020	8	28/05/2020
<i>Configuración de Máquinas Virtuales</i>	24/05/2020	15	08/06/2020
<i>Pruebas de conexión</i>	31/05/2020	9	09/06/2020
<i>Realización de Memoria</i>	04/05/2020	37	10/06/2020

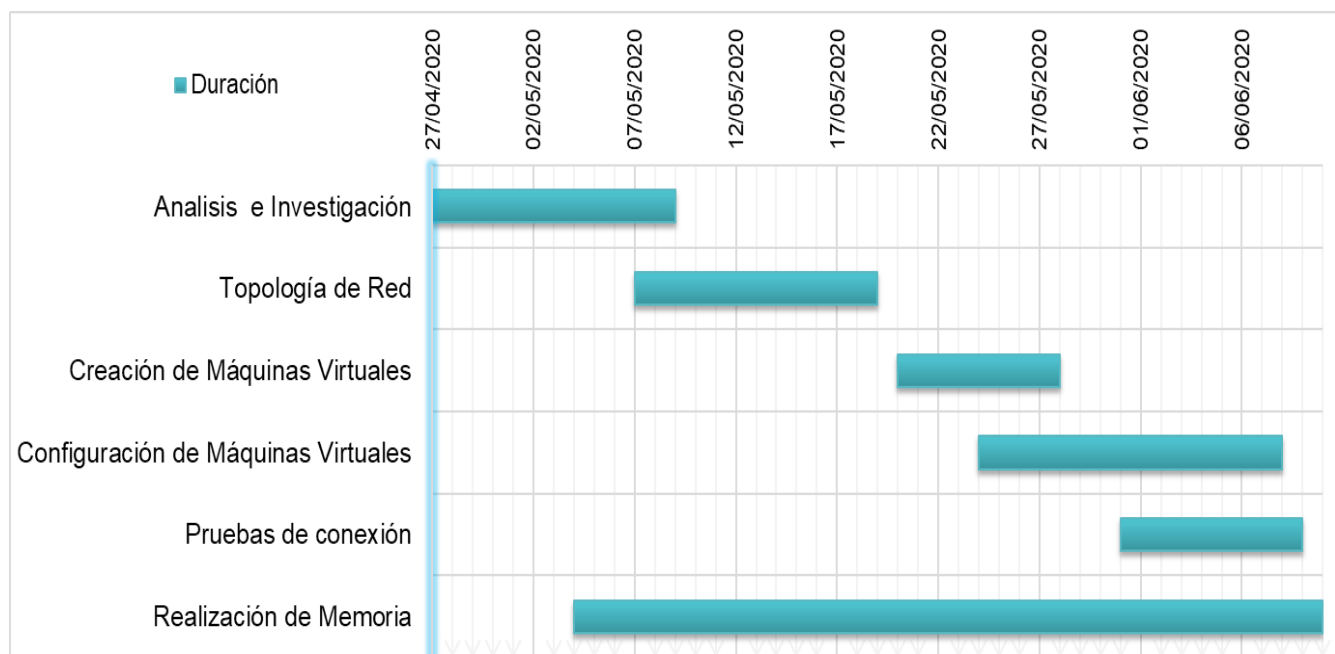


Figura 2: Diagrama de Gannt

Fuente : Propia

#### 4. TOPOLOGÍA DE RED UTILIZADA

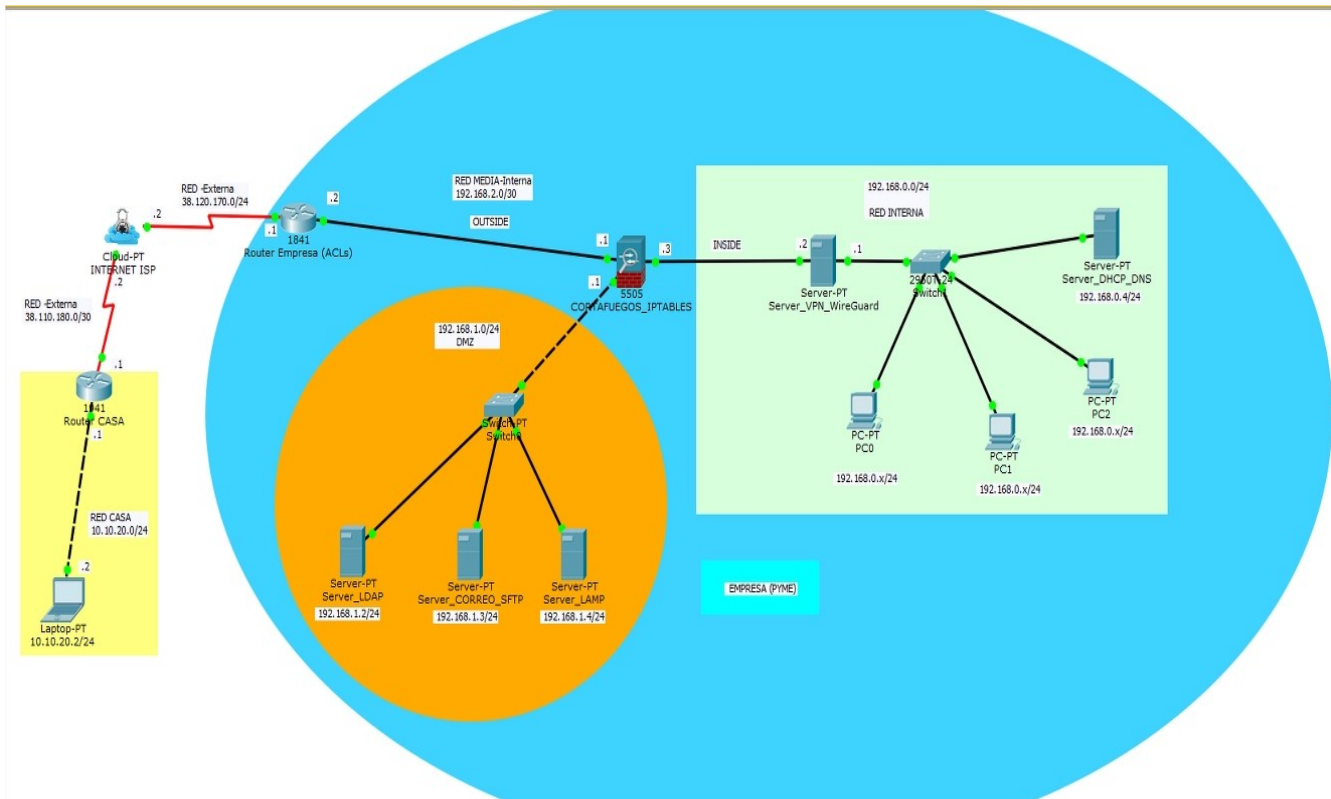


Figura 3: Topología de red (PYME)

Fuente : Propia

- Para realizar la topología de red se ha utilizado el software de Cisco “ Packet Tracer “.
- El diseño cuenta con cuatro (4) zonas :
  - (a) “ Red Interna “ donde se encontraran los empleados, un servidor DHCP-DNS y el servidor VPN con WireGuard.
  - (b) “ Zona DMZ “ donde se encontraran los diferentes servidores utilizados por la empresa ( OpenLDAP, Correo, SFTP y LAMP ).
  - (c) “ Red Media interna “ donde el router de la empresa se conectará con el cortafuegos.
  - (d) “ Red Casa “ desde donde se conectará el empleado.



## 5. IMPLEMENTACIÓN DEL DISEÑO

- Para realizar la creación y configuración del diseño se usarán recursos de diversas fuentes y los conocimientos adquiridos en los diferentes módulos del ciclo de ASIR:
  - Implantación y Administración de sistemas operativos.
  - Administración y Servicios de redes.
  - Seguridad y alta disponibilidad.
  - Administración de sistemas gestores de bases de datos.
  - Implantación de aplicaciones web.
- Se creará un laboratorio virtual para la implantación y configuración de los servidores en máquinas virtuales con prueba del servicio VPN WireGuard con el software de Oracle “ VM VirtualBox “.

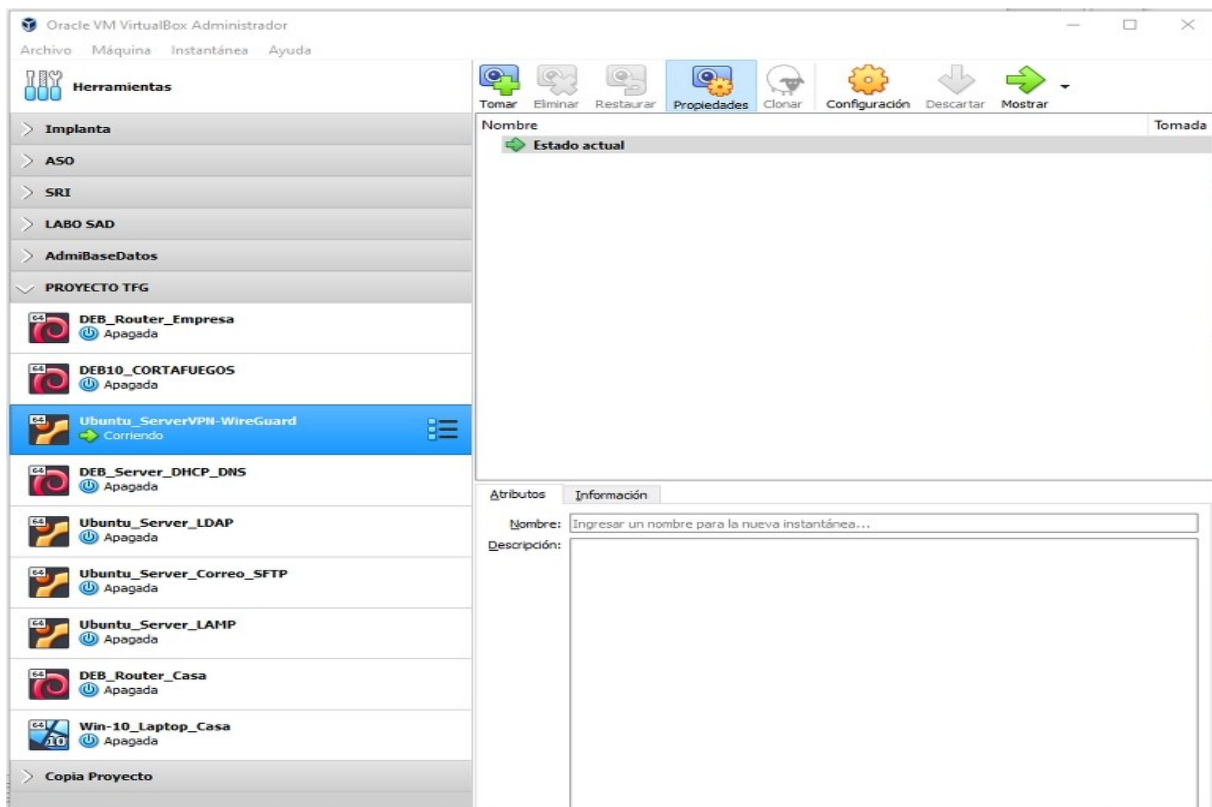


Figura 4: Máquinas creadas en VirtualBox

Fuente : Propia

## **5.1 CREACIÓN DE MÁQUINAS VIRTUALES**

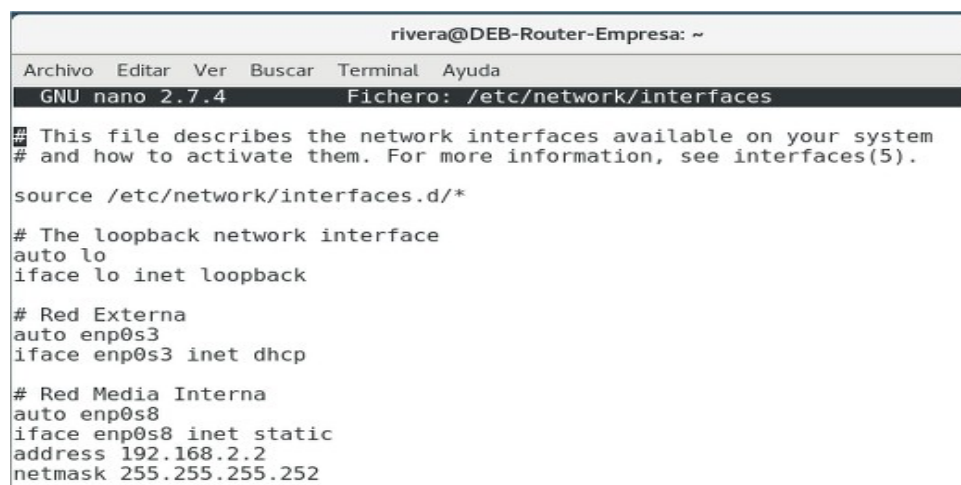
1. Router Empresa con 2 interfaces de red
  - a) Adaptador puente (Red Externa)
  - b) Red Media-Interna (192.168.2.2/30)
2. Cortafuegos iptables con 3 interfaces de red
  - a) Red Media-Interna (192.168.2.1/30)
  - b) Red DMZ (192.168.1.1/24)
  - c) Red Interna (192.168.0.3/24)
3. Server\_DHCP\_DNS con 1 interfaz de red
  - a) Red Interna (192.168.0.4/24)
4. Server\_LDAP\_DNS maestro con 1 interfaz de red
  - a) Red DMZ (192.168.1.2/24)
5. Server\_Correo\_SFTP con 1 interfaz de red
  - a) Red DMZ (192.168.1.3/24)
6. Server\_LAMP con 1 interfaz de red
  - a) Red DMZ (192.168.1.4/24)
7. Router Casa con 2 interfaces de red
  - a) Adaptador puente (Red Externa)
  - b) Red Casa (10.10.20.1/24)
8. Server\_VPN\_WireGuard con 2 interfaces de red
  - a) Red Interna (192.168.0.1/24)
  - b) Red Interna (192.168.0.2/24)
9. Laptop\_Casa con 1 interfaz de red
  - a) Red Casa (10.10.20.2/24)

### 5.1.1 CONFIGURACIÓN DE MÁQUINAS VIRTUALES

# Todos los comandos se ejecutan en un terminal como " root "

#### 1- Router Empresa

- ◆ Las configuraciones de red se realizan con un editor de texto en el fichero:  
*/etc/network/interfaces*
- ◆ Este fichero quedará configurado con dos interfaces de red de la siguiente forma:



```
rivera@DEB-Router-Empresa: ~
GNU nano 2.7.4 Fichero: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# Red Externa
auto enp0s3
iface enp0s3 inet dhcp

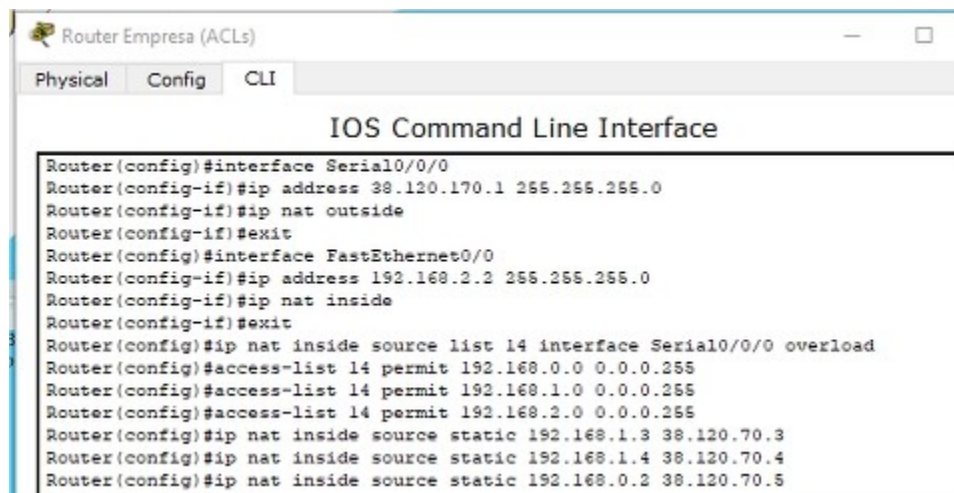
# Red Media Interna
auto enp0s8
iface enp0s8 inet static
address 192.168.2.2
netmask 255.255.255.252
```

Figura 5: Fichero */etc/network/interfaces*

Fuente : Propia

- ◆ Se activará de forma permanente el reenvío de datagramas en el fichero */etc/sysctl.conf* eliminando (#) de la línea *net.ipv4.ip\_forward = 1* y se ejecuta el comando *# sysctl -p* para hacer efectivos los cambios.
- ◆ Se utilizará el siguiente comando para hacer NAT :  
*# iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE*
- ◆ Para hacerlo permanente primero se crea el fichero *nat* con el comando :  
*# iptables-save > /etc/nat*
- ◆ Después se añadirá al final del fichero */etc/network/interfaces* la orden :  
*pre-up iptables-restore < /etc/nat*

- ◆ Los *routers* ofrecen un medio de filtrado del tráfico de la red mediante las ACLs. Para realizar este filtrado en un router Cisco o en el Packet tracer se pueden aplicar una serie de ACEs y de esta forma conseguir que el router haga NAT-P, también se les podrá asignar IPs publicas a los host de la red interna.



```
Router Empresa (ACLs)
Physical Config CLI
IOS Command Line Interface
Router(config)#interface Serial0/0/0
Router(config-if)#ip address 38.120.170.1 255.255.255.0
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#interface FastEthernet0/0
Router(config-if)#ip address 192.168.2.2 255.255.255.0
Router(config-if)#ip nat inside
Router(config-if)#exit
Router(config)#ip nat inside source list 14 interface Serial0/0/0 overload
Router(config)#access-list 14 permit 192.168.0.0 0.0.0.255
Router(config)#access-list 14 permit 192.168.1.0 0.0.0.255
Router(config)#access-list 14 permit 192.168.2.0 0.0.0.255
Router(config)#ip nat inside source static 192.168.1.3 38.120.70.3
Router(config)#ip nat inside source static 192.168.1.4 38.120.70.4
Router(config)#ip nat inside source static 192.168.0.2 38.120.70.5
```

*Figura 6: ACLs Packet tracer*

Fuente : Propia

- ◆ Ahora se pasará a configurar las DNS en el fichero `/etc/resolv.conf` donde se pondrá el dominio, las dns de Google, las de Opendns<sup>3</sup> y nuestro servidor DNS maestro.



```
riviera@DEB-Router-Empresa: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.7.4 Fichero: /etc/resolv.conf Modificado
domain Rivera.org
nameserver 8.8.8.8
nameserver 208.67.222.222
nameserver 192.168.1.2
```

*Figura 7: Fichero resolv.conf*

Fuente : Propia

<sup>3</sup> <https://www.opendns.com/>

- ◆ Por último se reiniciará el servicio con el comando :

```
# /etc/init.d/networking restart
```

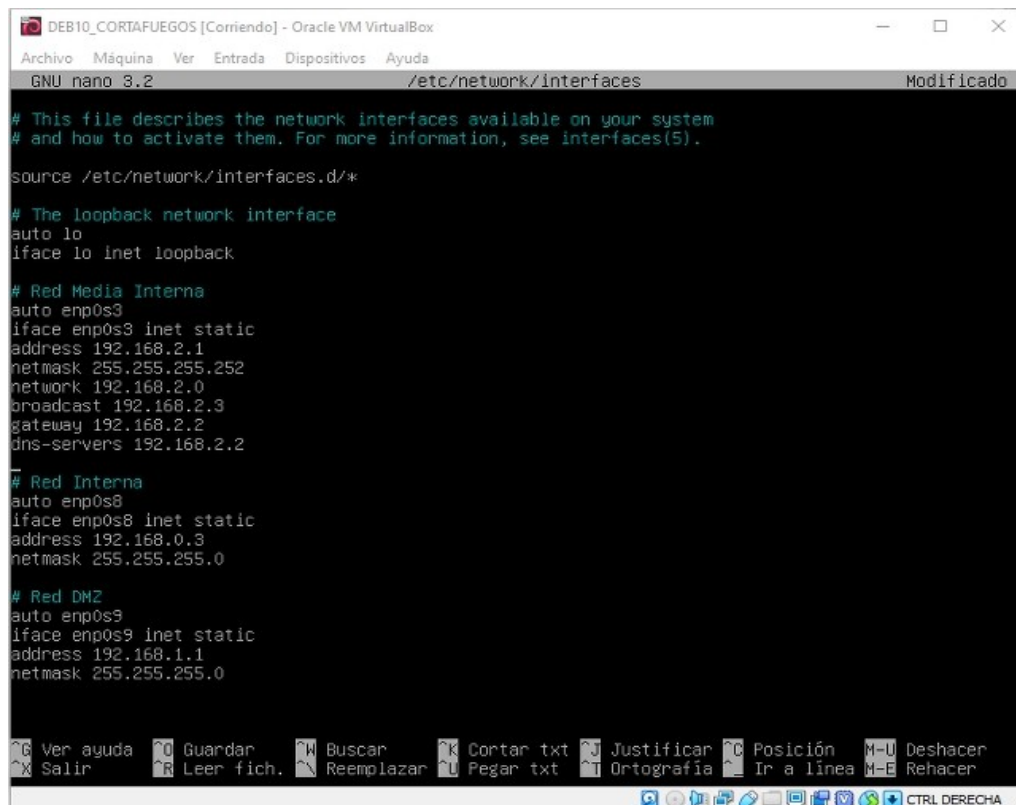
```
root@DEB-Router-Empresa:/home/rivera# /etc/init.d/networking restart
[ ok ] Restarting networking (via systemctl): networking.service.
root@DEB-Router-Empresa:/home/rivera#
```

Figura 8: Reinicio de servicio

Fuente : Propia

## 2- Cortafuegos iptables

- ◆ En esta máquina se configuran 3 interfaces de red de forma estática y activamos el reenvío de datagramas como en la máquina anterior.



```
DEB10_CORTAFUEGOS [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 3.2 /etc/network/interfaces Modificado

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# Red Media Interna
auto enp0s3
iface enp0s3 inet static
address 192.168.2.1
netmask 255.255.255.252
network 192.168.2.0
broadcast 192.168.2.3
gateway 192.168.2.2
dns-servers 192.168.2.2

# Red Interna
auto enp0s8
iface enp0s8 inet static
address 192.168.0.3
netmask 255.255.255.0

# Red DMZ
auto enp0s9
iface enp0s9 inet static
address 192.168.1.1
netmask 255.255.255.0

Ver ayuda  Guardar  Buscar  Cortar txt  Justificar  Posición  M-U  Deshacer
Salir  Leer fich.  Reemplazar  Pegar txt  Ortografía  Ir a línea  M-E  Rehacer
CTRL DERECHA
```

Figura 9: Fichero /etc/network/interfaces

Fuente : Propia

- ◆ Para realizar el cortafuegos con iptables, se podrá implementar Systemd<sup>4</sup> como servicio del sistema por su facilidad de uso. Se realizarán unos scripts para su correcta configuración y su posterior funcionamiento.
- ◆ Se crea el script “cortafuegos-on “

```
root@DEB10CORTAFUEGOS:~# nano /usr/lib/systemd/scripts/cortafuegos-on_
```

```
#!/bin/sh

# Borrar Reglas
iptables -F
iptables -X
iptables -Z
iptables -t nat -F

# Políticas por defecto
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
iptables -t nat -P PREROUTING ACCEPT
iptables -t nat -P POSTROUTING ACCEPT

# Enmascaramos la Red Interna y la DMZ
iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE

# Aceptamos al localhost
/sbin/iptables -A INPUT -i lo -j ACCEPT

# Aceptamos a la Red Interna acceder al Cortafuegos por ssh
iptables -A INPUT -s 192.168.0.0/24 -p tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -d 192.168.0.0/24 -p tcp --sport 22 -j ACCEPT

## Permitimos las conexiones Establecidas y Relacionadas hacia la Red Interna desde la Red Externa
## y desde la DMZ
iptables -A FORWARD -d 192.168.0.0/24 -p tcp -m state --state ESTABLISHED,RELATED -j ACCEPT
## Aceptamos el tráfico desde la Red Interna hacia la Red Externa y la DMZ
iptables -A FORWARD -s 192.168.0.0/24 -p tcp -j ACCEPT
iptables -A FORWARD -s 192.168.0.0/24 -p udp -j ACCEPT
iptables -A FORWARD -s 192.168.0.0/24 -p icmp -j ACCEPT
# Permitimos las respuestas PING hacia la Red Interna
iptables -A FORWARD -d 192.168.0.0/24 -p icmp --icmp-type echo-reply -j ACCEPT
## Aceptamos el tráfico hacia el Servidor WEB
iptables -A FORWARD -d 192.168.1.4/24 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -d 192.168.1.4/24 -p tcp --dport 443 -j ACCEPT
## Permitimos las conexiones Establecidas desde el servidor WEB
iptables -A FORWARD -d 192.168.1.4/24 -p tcp --sport 80 -m state --state ESTABLISHED -j ACCEPT
iptables -A FORWARD -d 192.168.1.4/24 -p tcp --sport 443 -m state --state ESTABLISHED -j ACCEPT
## Aceptamos el tráfico hacia el Servidor (Correo,ftp-sftp)
iptables -A FORWARD -d 192.168.1.3/24 -p tcp --dport 20,21,22,25 -j ACCEPT
## Permitimos las conexiones Establecidas y Relacionadas para el Servidor (correo,ftp-sftp)
iptables -A FORWARD -d 192.168.1.3/24 -p tcp --sport 20,21,22,25 -m state --state ESTABLISHED,RELATED -j ACCEPT
## Aceptamos las conexiones Establecidas desde la Red Externa hacia la Red Interna por udp
iptables -A FORWARD -d 192.168.0.0/24 -i enp0s3 -p udp -m state --state ESTABLISHED -j ACCEPT
## Ahora hacemos DNAT para las peticiones recibidas en los Servidores
iptables -t nat -A PREROUTING -i enp0s9 -p tcp --dport 80 -j DNAT --to-destination 192.168.1.4/24:80
iptables -t nat -A PREROUTING -i enp0s9 -p tcp --dport 443 -j DNAT --to-destination 192.168.1.4/24:443
iptables -t nat -A PREROUTING -i enp0s9 -p tcp --dport 22,25 -j DNAT --to-destination 192.168.1.3/24:22,25
## Ver RESULTADO iptables
iptables -L -n
iptables -t nat -L -n -v

exit 0
```

Figura 10: Fichero script “cortafuegos-on “

Fuente : Propia

4 [https://wiki.archlinux.org/index.php/Systemd\\_\(Espa%C3%B1ol\)](https://wiki.archlinux.org/index.php/Systemd_(Espa%C3%B1ol))



- ◆ Se crea el script “ cortafuegos-off “

```
root@DEB10CORTAFUEGOS:~# nano /usr/lib/systemd/scripts/cortafuegos-off_

GNU nano 3.2 cortafuegos-off
#!/bin/sh

iptables -F
iptables -X
iptables -t nat -F

# Ver Resultado
iptables -L -n -v
iptables -t nat -L -n -v

exit 0
```

Figura 11: Fichero script “ cortafuegos-off “  
Fuente : Propia

- ◆ Ahora se les dará permisos de ejecución a los dos scripts.

```
root@DEB10CORTAFUEGOS:~# chmod +x /usr/lib/systemd/scripts/cortafuegos-off
root@DEB10CORTAFUEGOS:~# chmod +x /usr/lib/systemd/scripts/cortafuegos-on
```

- ◆ El siguiente paso es crear el fichero del servicio “ cortafuegos-on.service “ con el comando : `# usr/lib/systemd/system/cortafuegos-on.service`

```
GNU nano 3.2 /usr/lib/systemd/system/cortafuegos-on.service

[Unit]
Description=Packet Filtering Framework

[Service]
Type=oneshot
ExecStart=/usr/lib/systemd/scripts/cortafuegos-on
ExecReload=/usr/lib/systemd/scripts/cortafuegos-on
ExecStop=/usr/lib/systemd/scripts/cortafuegos-off
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
```

Figura 12: Fichero del servicio  
Fuente : Propia

- ◆ Después estando en el directorio `usr/lib/systemd/system/`, solo faltaría habilitar el servicio e iniciar el cortafuegos, para ello se utilizarán los siguientes comandos :

```
# systemctl enable cortafuegos-on.service ( Habilitar )
```

```
# systemctl "start/stop" cortafuegos-on.service ( Iniciar/Parar )
```

```
# systemctl status cortafuegos-on.service ( Ver estado )
```

```
root@DEB10CORTAFUEGOS:/usr/lib/systemd/system# systemctl enable cortafuegos-on.service
Created symlink /etc/systemd/system/multi-user.target.wants/cortafuegos-on.service → /lib/systemd/system/cortafuegos-on.service.
root@DEB10CORTAFUEGOS:/usr/lib/systemd/system# systemctl start cortafuegos-on.service
root@DEB10CORTAFUEGOS:/usr/lib/systemd/system# systemctl status cortafuegos-on.service
• cortafuegos-on.service - Packet Filtering Framework
   Loaded: loaded (/lib/systemd/system/cortafuegos-on.service; enabled; vendor preset: enabled)
   Active: active (exited) since Wed 2020-06-03 02:32:53 CEST; 14s ago
   Process: 613 ExecStart=/usr/lib/systemd/scripts/cortafuegos-on (code=exited, status=0/SUCCESS)
   Main PID: 613 (code=exited, status=0/SUCCESS)
```

Figura 13: Habilitar servicio

Fuente : Propia

### 3 - Server\_DHCP\_DNS

- ◆ En esta máquina virtual situada en la red interna se implantará un servidor DHCP y DNS caché sencillo con DNSMASQ<sup>5</sup>. La única interfaz de red se configura con una IP estática desde el fichero `/etc/network/interfaces`.

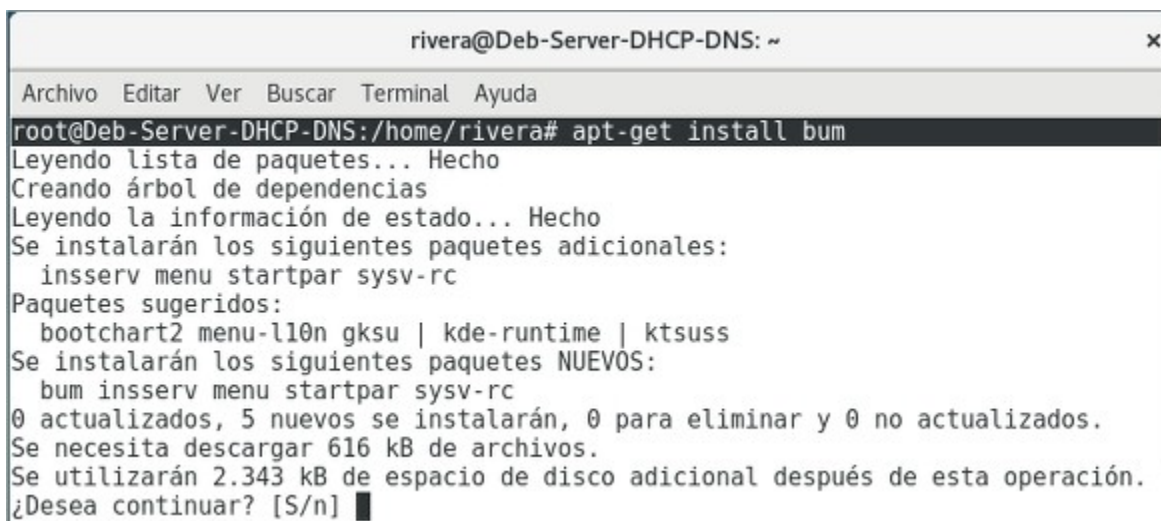
```
# Red Interna
auto enp0s3
iface enp0s3 inet static
address 192.168.0.4
netmask 255.255.255.0
gateway 192.168.0.1
dns-nameservers 192.168.0.4 192.168.1.2
dns-search Rivera.org
```

Figura 14: Fichero `/etc/network/interfaces`

Fuente : Propia

5 [https://www.gamificafp.com/pluginfile.php/8111/mod\\_resource/content/0/2-Servidor-DHCP-y-DNS.pdf](https://www.gamificafp.com/pluginfile.php/8111/mod_resource/content/0/2-Servidor-DHCP-y-DNS.pdf)

- ◆ De esta forma se tendrá un servidor caché en la red interna que resolverá las IPs, los nombres y adicionalmente asignará las IPs de manera dinámica dentro del rango establecido.
- ◆ Se ejecutaran en un terminal para su instalación y arranque los comandos :  
`# apt-get install dnsmasq`  
`# /etc/init.d/dnsmasq restart`
- ◆ Si pretendemos el arranque automático del servicio al iniciar la máquina, podemos utilizar la herramienta gráfica “ boot-up manager<sup>6</sup> ” .
- ◆ Para su instalación se utilizará el comando que aparece en la siguiente imagen :



```
riviera@Deb-Server-DHCP-DNS: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@Deb-Server-DHCP-DNS:/home/riviera# apt-get install bum  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Se instalarán los siguientes paquetes adicionales:  
  insserv menu startpar sysv-rc  
Paquetes sugeridos:  
  bootchart2 menu-ll0n gksu | kde-runtime | ktsuss  
Se instalarán los siguientes paquetes NUEVOS:  
  bum insserv menu startpar sysv-rc  
0 actualizados, 5 nuevos se instalarán, 0 para eliminar y 0 no actualizados.  
Se necesita descargar 616 kB de archivos.  
Se utilizarán 2.343 kB de espacio de disco adicional después de esta operación.  
¿Desea continuar? [S/n]
```

*Figura 15: Comando instalar bum*

Fuente : Propia

6 <http://somebooks.es/administrar-servicios-demonios-de-ubuntu-con-boot-up-manager/>

- ◆ Con esta herramienta la gestión de los servicios se vería de la siguiente forma :

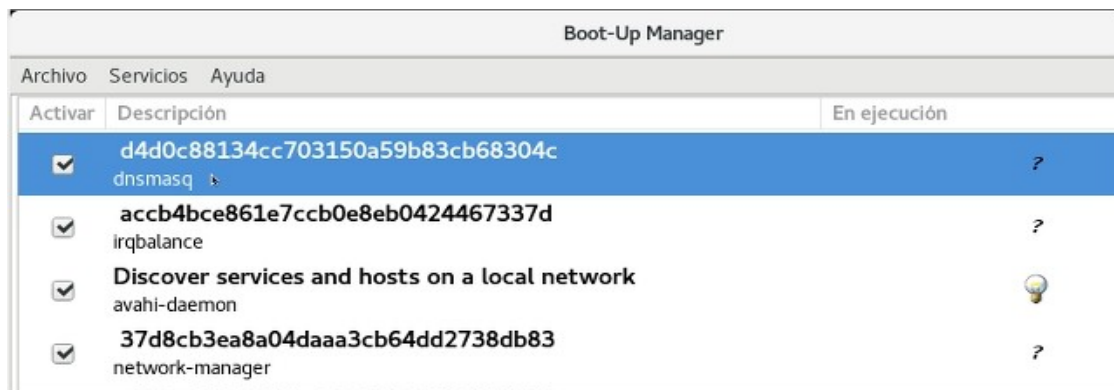


Figura 16: Herramienta gráfica "boot-up manager"

Fuente : Propia

- ◆ Ahora en el fichero de configuración `/etc/dnsmasq.conf` se añadirá el rango y el tiempo de cesión de IPs :

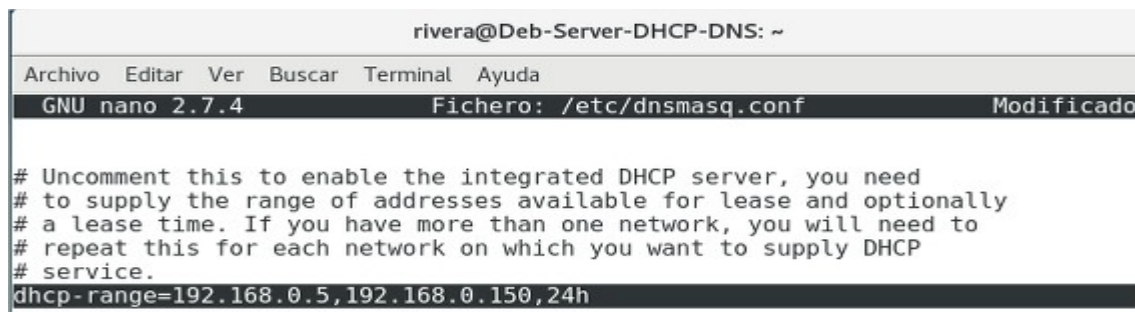


Figura 17: Fichero `/etc/dnsmasq.conf`

Fuente : Propia

- ◆ Como la puerta de enlace de los clientes será otra máquina distinta a este servidor se le indicará de la siguiente forma, terminando así la configuración de este servidor :

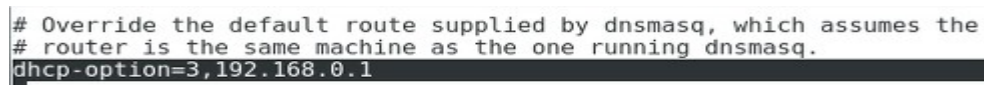


Figura 18: Fichero `/etc/dnsmasq.conf`

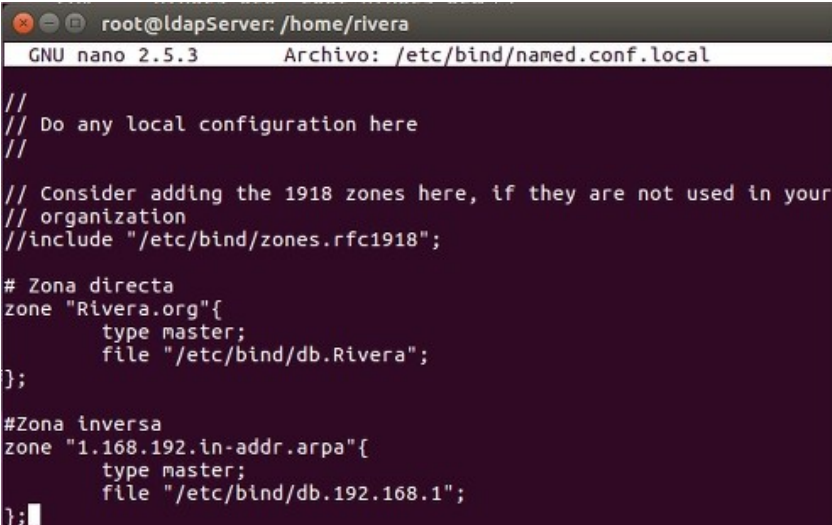
Fuente : Propia

#### 4 – Server\_LDAP\_DNS maestro

- ◆ LDAP, es un protocolo que ofrece el acceso a un servicio de directorio implementado sobre un entorno de red, con el objeto de acceder a una determinada información.
- ◆ En este servidor también instalaremos el servidor DNS Bind como servidor maestro, que será capaz de resolver peticiones de nombres internos para nuestra red de empresa.

##### A) DNS

- ◆ El paquete bind9 será instalado para este fin, para ello se hace uso de los siguientes comandos: `# sudo apt-get update` y `# sudo apt-get install bind9`
- ◆ Después de la instalación, se modificará el fichero `/etc/bind/named.conf.local` y en este fichero se declaran las zonas asociadas al servidor de dominio en cuestión. Para ello, se utiliza el comando : `# nano /etc/bind/named.conf.local`



```
root@ldapServer: /home/rivera
GNU nano 2.5.3 Archivo: /etc/bind/named.conf.local

//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

# Zona directa
zone "Rivera.org"{
    type master;
    file "/etc/bind/db.Rivera";
};

#Zona inversa
zone "1.168.192.in-addr.arpa"{
    type master;
    file "/etc/bind/db.192.168.1";
};
```

Figura 19: Fichero `named.conf.local`

Fuente : Propia

◆ A continuación se hará una copia de los ficheros creados con la instalación /etc/bind/db.local nombrándolo como db.Rivera y /etc/bind/db.127 como db.192.168.1. En estos ficheros se cambiará “ localhost “ por el nombre de dominio y 127.0.0.1 por la dirección IP del servidor, además se añadirán los servidores de correo, web y SFTP. La configuración de estos ficheros quedará como en las siguientes imágenes:

```

root@ldapServer: /etc/bind
GNU nano 2.5.3 Archivo: db.Rivera
; BIND data file for local loopback interface
;
$TTL 604800
@ IN SOA Rivera.org. root.Rivera.org. (
    2          ; Serial
    604800     ; Refresh
    86400      ; Retry
    2419200    ; Expire
    604800 )    ; Negative Cache TTL
;
; IN NS Rivera.org.
; IN MX 10 mail
@ IN A 192.168.1.2
@ IN AAAA ::1
ldapServer IN A 192.168.1.2
ServerCorreoSFTP IN A 192.168.1.3
ServerLAMP IN A 192.168.1.4
mail IN A 192.168.1.2
ns1 IN CNAME ldapServer
ftp IN CNAME ServerCorreoSFTP
www IN CNAME ServerLAMP
    
```

Figura 20: Fichero zona directa

Fuente : Propia

```

root@ldapServer: /etc/bind
GNU nano 2.5.3 Archivo: db.192.168.1
; BIND reverse data file for local loopback interface
;
$TTL 604800
@ IN SOA Rivera.org. root.Rivera.org. (
    1          ; Serial
    604800     ; Refresh
    86400      ; Retry
    2419200    ; Expire
    604800 )    ; Negative Cache TTL
;
; IN NS ns.
; IN NS ns1.
2 IN PTR Rivera.org.
2 IN PTR Rivera.org
2 IN PTR ns1.Rivera.org
3 IN PTR ServerCorreoSFTP.Rivera.org.
4 IN PTR ServerLAMP.Rivera.org.
    
```

Figura 21: Fichero zona inversa

Fuente : Propia



- ◆ Se puede comprobar la correcta sintaxis de las zonas con el comando :

```
# named-checkzone Rivera.org /etc/bind/db.Rivera
# named-checkzone Rivera.org /etc/bind/db.192.168.1
```

```
root@ldapServer:/etc/bind# named-checkzone Rivera.org /etc/bind/db.Rivera
zone Rivera.org/IN: loaded serial 2
OK
root@ldapServer:/etc/bind# named-checkzone Rivera.org /etc/bind/db.192.168.1
zone Rivera.org/IN: loaded serial 1
OK
root@ldapServer:/etc/bind#
```

Figura 22: Comprobar zonas

Fuente : Propia

- ◆ El fichero de configuración también se puede comprobar con :

```
# named-checkconf -z /etc/bind/named.conf
```

```
root@ldapServer:/etc/bind# named-checkconf -z /etc/bind/named.conf
zone Rivera.org/IN: loaded serial 2
zone 1.168.192.in-addr.arpa/IN: loaded serial 1
zone localhost/IN: loaded serial 2
zone 127.in-addr.arpa/IN: loaded serial 1
zone 0.in-addr.arpa/IN: loaded serial 1
zone 255.in-addr.arpa/IN: loaded serial 1
root@ldapServer:/etc/bind#
```

Figura 23: Comprobar named.conf

Fuente : Propia

- ◆ **Importante :**

- A todas las maquinas de la empresa se les indicará el servidor DNS y el dominio.

```
dns-nameservers 192.168.1.2
```

```
name-search Rivera.org
```

## B) LDAP

- ◆ La finalidad de nuestro servidor LDAP es que sirva de almacén de usuarios y grupos para autenticar sistemas linux y servicios como ftp y web y se deberá crear una estructura que parta de la base de nuestro directorio, para almacenar dicha información.
- ◆ Se instalará el paquete SLAPD y un paquete adicional que contiene las utilidades de administración de LDAP, ldap-utils. Se pedirá la contraseña para la administración de LDAP. `# sudo apt-get install slapd ldap-utils`
- ◆ A continuación, se instalará la librería NSS para LDAP. Esta librería ofrece una interfaz para acceder y configurar distintas bases de datos utilizadas para almacenar cuentas de usuario. `# sudo apt install libnss-ldap`
- ◆ Nos aparecerá el menú de configuración de ldap-auth-config, primero nos solicita la dirección URI del servidor LDAP. Es importante dejarla como en la siguiente imagen y poner la IP de nuestro servidor:



Figura 24: URI servidor

Fuente : Propia

- ◆ Lo siguiente será indicarle el dominio y la cuenta de administrador.

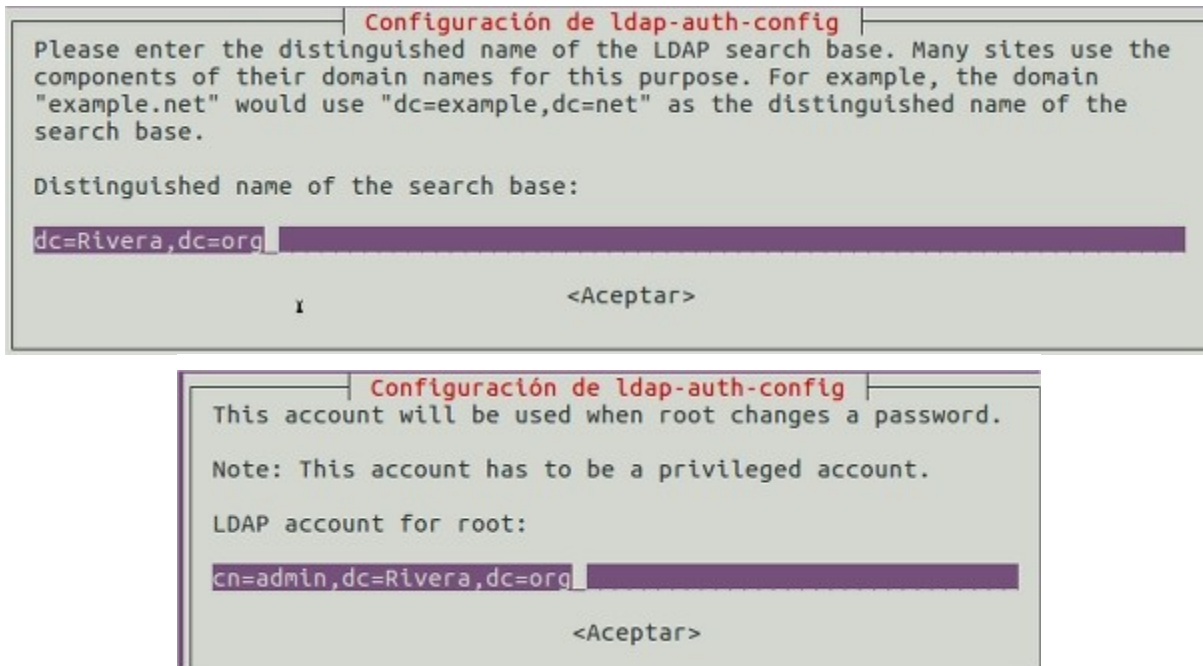


Figura 25: Dominio y cuenta admin  
Fuente : Propia

- ◆ Indicaremos la versión de protocolo “3”, utilidades PAM “si” y después “no” para la necesidad de identificarse al realizar consultas.
- ◆ Se usará un script que nos ayuda a modificar los archivos de configuración de PAM y NSS. Para usar este script, ejecutamos el siguiente comando:  

```
# sudo auth-client-config -t nss -p lac_ldap
```
- ◆ Seguidamente, se deberá actualizar la configuración de las políticas de autenticación predeterminadas de PAM, esto permitirá autenticar usuarios. Para ello, se usará el siguiente comando:  

```
# sudo pam-auth-update
```

- ◆ En el asistente de configuración se seleccionarán los módulos para autenticar usuarios y una vez haya terminado se confirmará en el fichero `/etc/ldap.conf` que los siguientes datos son los correctos.

```
Host 192.168.1.2
base dc=Rivera,dc=org
uri ldap://192.168.1.2/
rootbinddn cn=admin,dc=Rivera,dc=org
ldap_version 3
bind_policy soft
```

- ◆ Con esto se habrá terminado la configuración del servidor LDAP. Ahora está listo para autenticar usuarios.

## 5 - Server\_Correo\_SFTP

### A) SFTP

- ◆ Al igual que con los otros servidores, se tiene que configurar una IP estática desde el fichero `/etc/network/interfaces`.

```
# Red DMZ
auto enp0s3
iface enp0s3 inet static
address 192.168.1.3
netmask 255.255.255.0
gateway 192.168.1.1
```

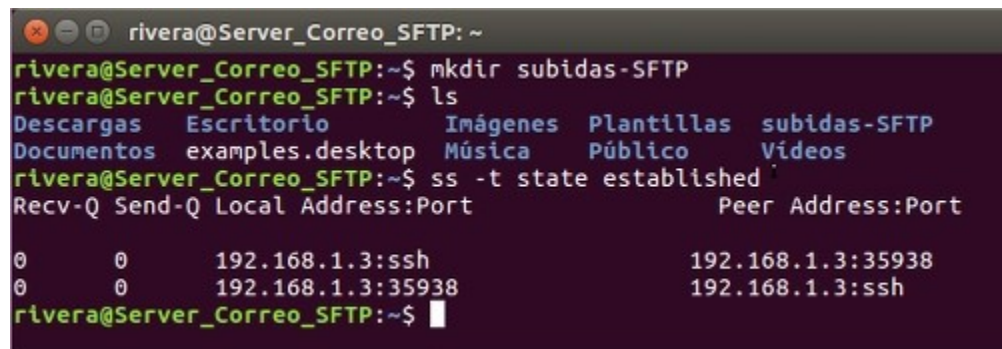
*Figura 26: Fichero  
`/etc/network/interfaces`*

Fuente : Propia

- ◆ Para instalar el servicio SFTP<sup>7</sup> se utilizará el comando :

```
# apt-get install openssh-server
```

- ◆ Una vez instalado se creará una carpeta en el servidor y se comprobará el estado de la conexión SSH como se puede apreciar en la siguiente imagen.



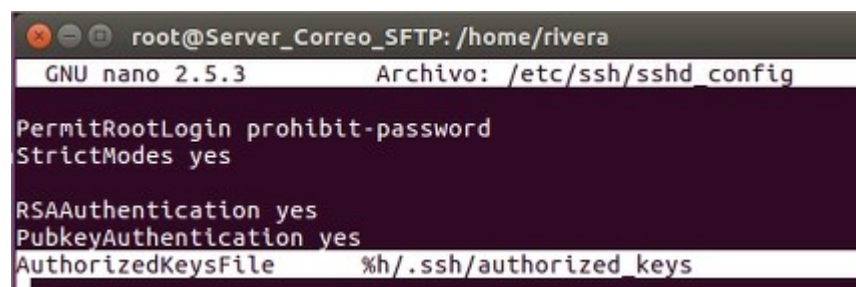
```
rivera@Server_Correo_SFTP: ~  
rivera@Server_Correo_SFTP:~$ mkdir subidas-SFTP  
rivera@Server_Correo_SFTP:~$ ls  
Descargas  Escritorio  Imágenes  Plantillas  subidas-SFTP  
Documentos examples.desktop Música  Público  Vídeos  
rivera@Server_Correo_SFTP:~$ ss -t state established  
Recv-Q Send-Q Local Address:Port Peer Address:Port  
0      0      192.168.1.3:ssh      192.168.1.3:35938  
0      0      192.168.1.3:35938    192.168.1.3:ssh  
rivera@Server_Correo_SFTP:~$
```

Figura 27: Creación de carpeta y estado de la conexión

Fuente : Propia

- ◆ Comprobamos en el archivo de la configuración del servidor (/etc/ssh/sshd\_config) que la siguiente cláusula está descomentada:

```
AuthorizedKeysFile .ssh/authorized_keys
```



```
root@Server_Correo_SFTP: /home/rivera  
GNU nano 2.5.3 Archivo: /etc/ssh/sshd_config  
  
PermitRootLogin prohibit-password  
StrictModes yes  
  
RSAAuthentication yes  
PubkeyAuthentication yes  
AuthorizedKeysFile %h/.ssh/authorized_keys
```

Figura 28: Autorizar claves

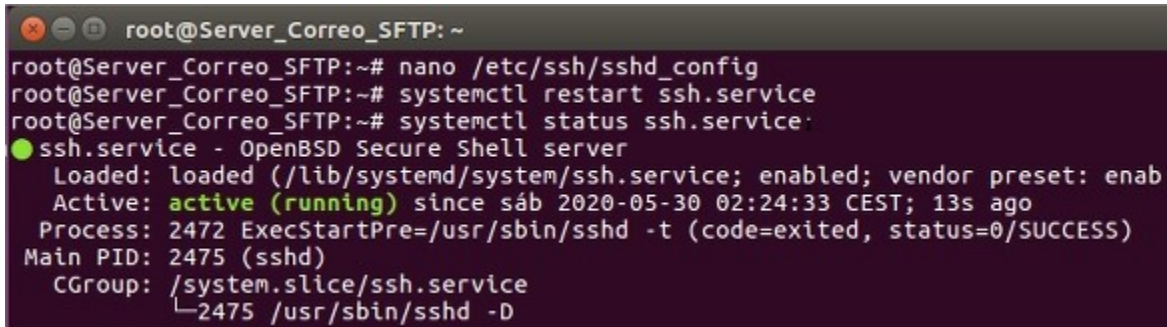
Fuente : Propia

<sup>7</sup> <https://www.ssh.com/ssh/sftp/>

- ◆ Reiniciamos el servicio y se comprueba su estado con los siguientes comandos :

```
# systemctl restart ssh.service
```

```
# systemctl status ssh.service
```



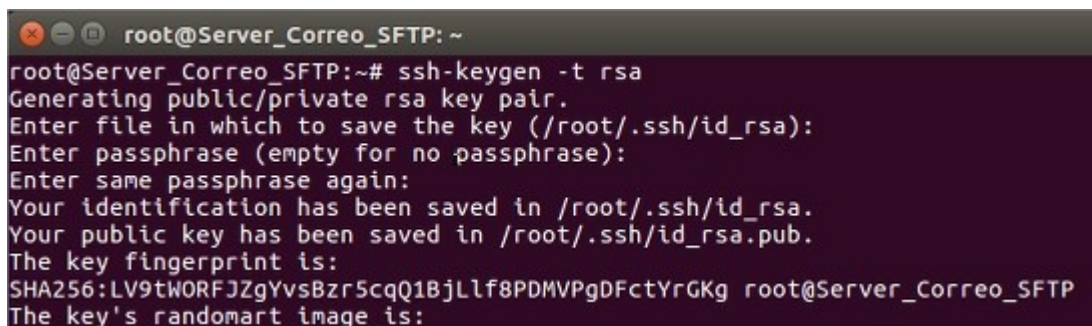
```
root@Server_Correo_SFTP: ~
root@Server_Correo_SFTP:~# nano /etc/ssh/sshd_config
root@Server_Correo_SFTP:~# systemctl restart ssh.service
root@Server_Correo_SFTP:~# systemctl status ssh.service:
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enab
   Active: active (running) since sáb 2020-05-30 02:24:33 CEST; 13s ago
   Process: 2472 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 2475 (sshd)
   CGroup: /system.slice/ssh.service
           └─2475 /usr/sbin/sshd -D
```

Figura 29: Estado del servicio ssh

Fuente : Propia

- ◆ Como se puede comprobar en la imagen anterior el servicio está activo y es el momento de generar las claves RSA<sup>8</sup> y copiar la clave pública en el directorio del usuario.

- Comando para generar las claves : `# ssh-keygen -t rsa`



```
root@Server_Correo_SFTP: ~
root@Server_Correo_SFTP:~# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:LV9tWORFJZgYvsBzr5cqQ1BjLlf8PDMVPgDFctYrGKg root@Server_Correo_SFTP
The key's randomart image is:
```

Figura 30: Generar claves ssh

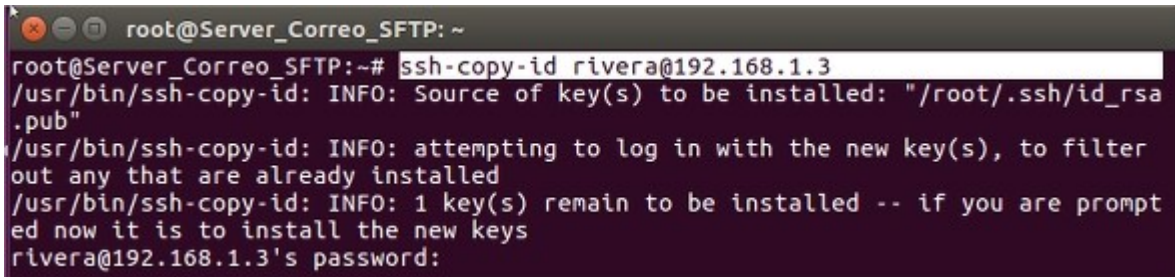
Fuente : Propia

8 <https://www.ecured.cu/RSA>



- Comando para copiar clave publica en el directorio del usuario :

```
# ssh-copy-id "usuario" @ " IP servidor "
```

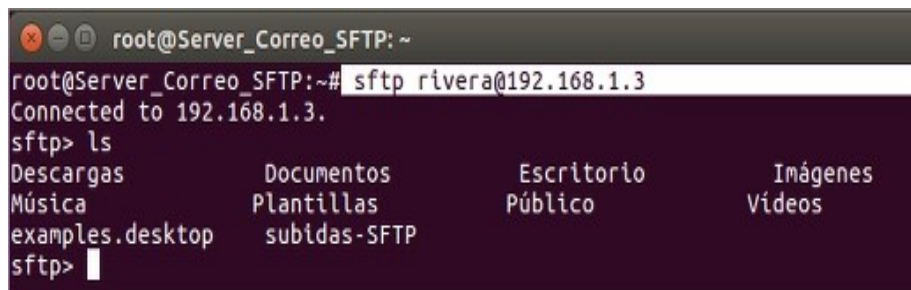


```
root@Server_Correo_SFTP: ~  
root@Server_Correo_SFTP:~# ssh-copy-id rivera@192.168.1.3  
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa  
.pub"  
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter  
out any that are already installed  
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt  
ed now it is to install the new keys  
rivera@192.168.1.3's password:
```

Figura 31: Copiar clave publica

Fuente : Propia

- ◆ Para terminar nos conectamos con el usuario al servidor SFTP



```
root@Server_Correo_SFTP: ~  
root@Server_Correo_SFTP:~# sftp rivera@192.168.1.3  
Connected to 192.168.1.3.  
sftp> ls  
Descargas          Documentos          Escritorio          Imágenes  
Música             Plantillas          Público            Vídeos  
examples.desktop  subidas-SFTP  
sftp>
```

Figura 32: Conexión al servidor SFTP

Fuente : Propia

### B) MailServer

- ◆ En esta misma máquina se implementará el servidor de correo, para ello se instalarán un agente de transferencia de correos ( MTA ) que en este caso será “Postfix<sup>9</sup>”, un agente de entrega de correo ( MDA ) como “Dovecot<sup>10</sup>” y un cliente de correo web ( MUA ) como es el caso de “SquirrelMail<sup>11</sup>”.
- ◆ Se empezará instalando y configurando Postfix, para ello se usará el comando :  
`# apt-get install postfix`
- ◆ Durante la instalación se solicitará la configuración básica, se debe elegir la opción “Sitio de Internet” como lo indica la imagen siguiente :

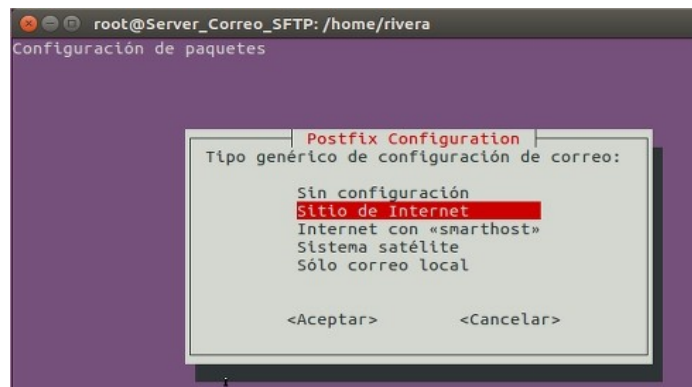


Figura 33: Conexión al servidor SFTP

Fuente : Propia

9 [https://wiki.archlinux.org/index.php/Postfix\\_\(Espa%C3%B1ol\)](https://wiki.archlinux.org/index.php/Postfix_(Espa%C3%B1ol))

10 <https://wiki.archlinux.org/index.php/Dovecot>

11 <https://wiki.archlinux.jp/index.php/Squirrelmail>

- ◆ Ahora se le asigna el nombre del sistema de correos o dominio.

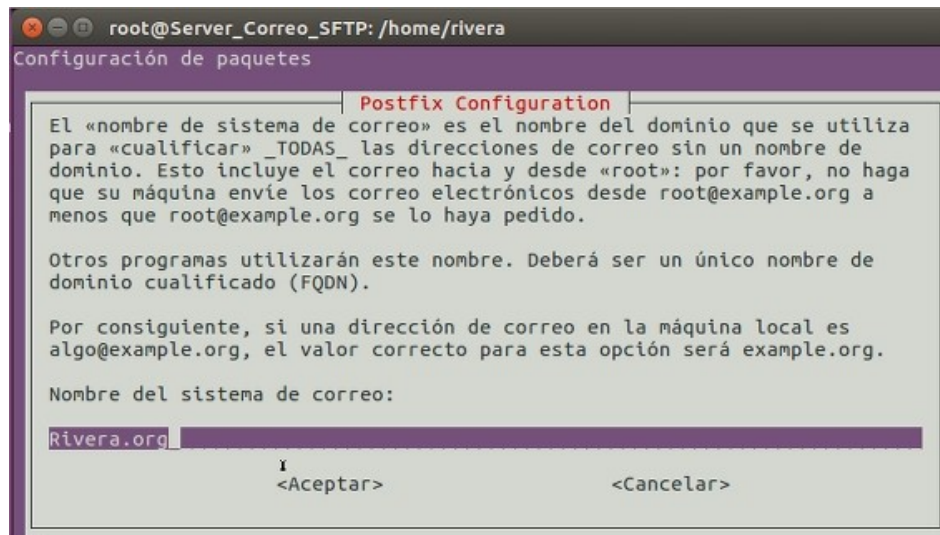


Figura 34: Conexión al servidor SFTP

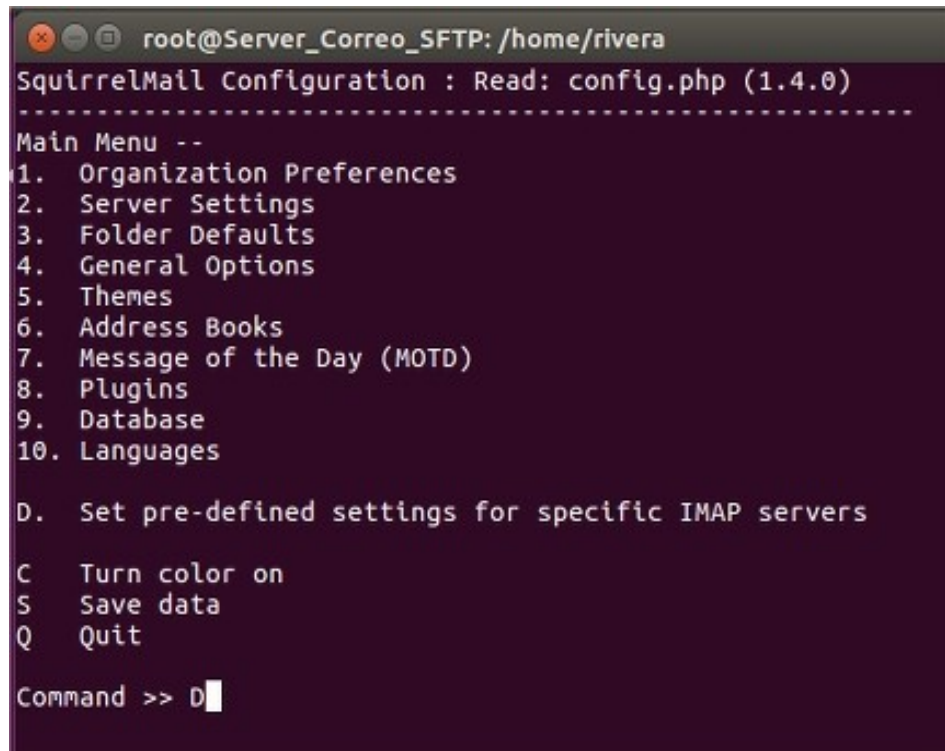
Fuente : Propia

- ◆ El siguiente paso para poner el servidor de correo en marcha, es instalar Dovecot, que será el servidor de IMAP/POP3. Se ejecuta en terminal el comando y se deja la configuración por defecto :  
`# apt-get install dovecot-core dovecot-imapd dovecot-pop3d`
- ◆ A continuación se instalará el cliente de correo web SquirrelMail y para su correcto funcionamiento se deberá instalar también Apache2<sup>12</sup> para que funcione en el servidor . Los comandos serán :  
`# apt-get install apache2`  
`# apt-get install squirrelmail`

12 <https://httpd.apache.org/>

- ◆ La configuración de SquirrelMail es bastante sencilla y se abre ejecutando en un terminal lo siguiente:

```
# squirrelmail-configure
```



```
root@Server_Correo_SFTP: /home/rivera
SquirrelMail Configuration : Read: config.php (1.4.0)
-----
Main Menu --
1. Organization Preferences
2. Server Settings
3. Folder Defaults
4. General Options
5. Themes
6. Address Books
7. Message of the Day (MOTD)
8. Plugins
9. Database
10. Languages

D. Set pre-defined settings for specific IMAP servers

C Turn color on
S Save data
Q Quit

Command >> D
```

Figura 35: Configuración SquirrelMail

Fuente : Propia

- ◆ Se ingresa **D** para establecer una configuración predeterminada para el servidor IMAP donde se elegirá **dovecot**, a continuación, se ingresa **2** para acceder a la configuración del servidor, luego **1** para configurar el dominio y **S** para guardar los cambios. La configuración se muestra en las siguientes imágenes :

```

root@Server_Correo_SFTP: /home/rivera
-----
While we have been building SquirrelMail, we have discovered some
preferences that work better with some servers that don't work so
well with others. If you select your IMAP server, this option will
set some pre-defined settings for that server.

Please note that you will still need to go through and make sure
everything is correct. This does not change everything. There are
only a few settings that this will change.

Please select your IMAP server:
  bincimap      = Binc IMAP server
  courier       = Courier IMAP server
  cyrus         = Cyrus IMAP server
  dovecot       = Dovecot Secure IMAP server
  exchange     = Microsoft Exchange IMAP server
  hmailserver   = hMailServer
  macosx       = Mac OS X Mailserver
  mercury32     = Mercury/32
  uw           = University of Washington's IMAP server
  gmail        = IMAP access to Google mail (Gmail) accounts
  quit         = Do not change anything
Command >> dovecot

```

Figura 36: Configuración Imap server

Fuente : Propia

```

root@Server_Correo_SFTP: /home/rivera
-----
General
-----
1. Domain          : trim(implode('', file('/etc/'.(file_exists('/etc/mailname')?'mail':'host').'name')))
2. Invert Time     : false
3. Sendmail or SMTP : SMTP

A. Update IMAP Settings : localhost:143 (dovecot)
B. Update SMTP Settings : localhost:25

R Return to Main Menu
C Turn color on
S Save data
Q Quit

Command >> 1

The domain name is the suffix at the end of all email addresses. If
for example, your email address is jdoe@example.com, then your domain
would be example.com.

[trim(implode('', file('/etc/'.(file_exists('/etc/mailname')?'mail':'host').'name'))): Rivera.org

```

Figura 37: Configuración del dominio

Fuente : Propia

- ◆ Después de terminar la configuración, se creará un enlace simbólico en el directorio `/var/www/html` y a continuación se reinicia el servicio de apache para que la carpeta pueda ser vista desde un navegador.

```
root@Server_Correo_SFTP: /var/www/html
root@Server_Correo_SFTP:/home/rivera# cd /var/www/html
root@Server_Correo_SFTP:/var/www/html# ln -s /usr/share/squirrelmail webmail
root@Server_Correo_SFTP:/var/www/html#
```

*Figura 38: Enlace simbólico*

Fuente : Propia

```
root@Server_Correo_SFTP: ~
root@Server_Correo_SFTP:~# /etc/init.d/apache2 restart
[ ok ] Restarting apache2 (via systemctl): apache2.service.
root@Server_Correo_SFTP:~#
```

*Figura 39: Reinicio Apache*

Fuente : Propia

- ◆ Por último se comprueba en el navegador ingresando el dominio seguido de `/webmail`.



*Figura 40: Cliente de correo web*

Fuente : Propia



## 6 - Server\_LAMP

- ◆ LAMP es una plataforma de código abierto para el desarrollo web que usa Linux como sistema operativo, Apache como servidor web, MySQL como base de datos relacional y PHP como lenguaje de scripts orientado a objetos.
- ◆ Contiene el software necesario para configurar sitios web o servidores dinámicos con esfuerzo reducido.
- ◆ Como se viene haciendo con los anteriores servidores, se configura la única interfaz de red con una IP estática desde el fichero : `/etc/network/interfaces`

```
# Red DMZ
auto enp0s3
iface enp0s3 inet static
address 192.168.1.4
netmask 255.255.255.0
gateway 192.168.1.1
```

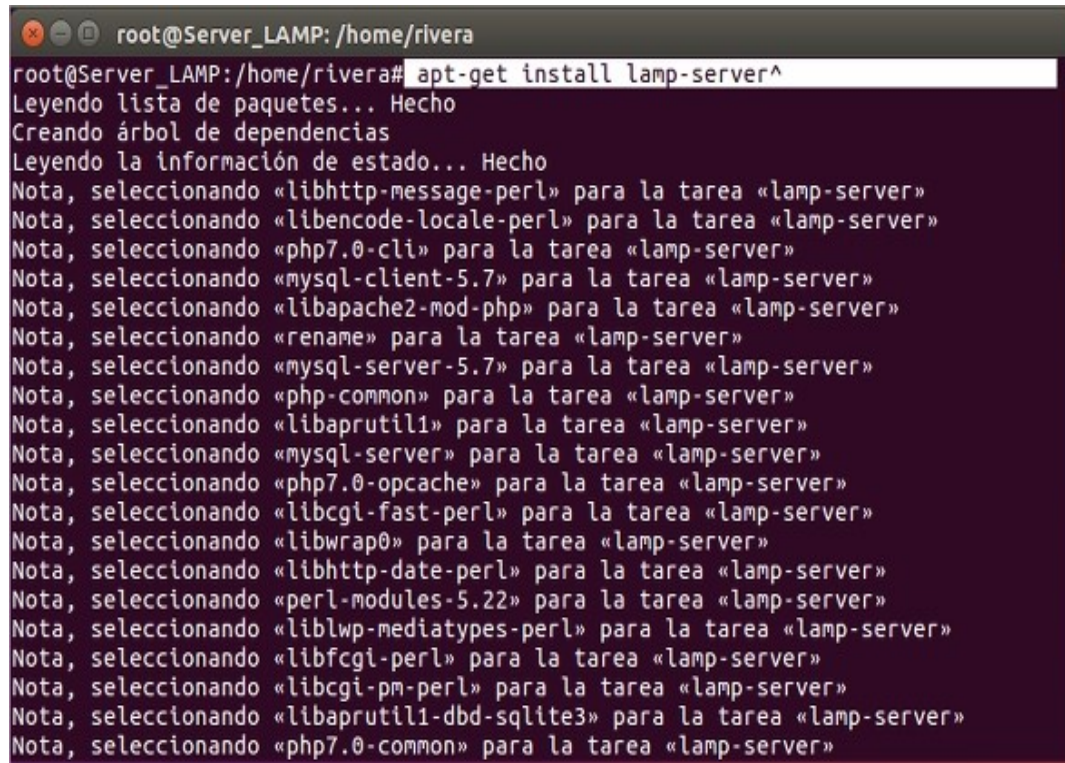
Figura 41: Fichero

`/etc/network/interfaces`

Fuente : Propia

- ◆ Su instalación será bastante sencilla y con muy pocos comandos se tendrá el servidor operativo.
- ◆ Existen diferentes métodos para instalar LAMP en Ubuntu, pero el más eficiente es utilizando el comando : `# apt-get install lamp-server^`
- ◆ El uso del carácter ^ significa que lo que precede es un metapaquete<sup>13</sup>. Al instalar metapaquetes, también se instalarán otros paquetes. Se agregarán paquetes como `apache2-utils`, `libaprutil1`, `libhttp-date-perl`, `php-mysql`, `php7.3-readline`, `ssl-cert` y más, junto con Apache, MySQL y PHP.

13 <https://www.linuxadictos.com/que-son-los-meta-paquetes-de-linux.html>



```
root@Server_LAMP: /home/rivera
root@Server_LAMP:/home/rivera# apt-get install lamp-server^
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Nota, seleccionando «libhttp-message-perl» para la tarea «lamp-server»
Nota, seleccionando «libencode-locale-perl» para la tarea «lamp-server»
Nota, seleccionando «php7.0-cli» para la tarea «lamp-server»
Nota, seleccionando «mysql-client-5.7» para la tarea «lamp-server»
Nota, seleccionando «libapache2-mod-php» para la tarea «lamp-server»
Nota, seleccionando «rename» para la tarea «lamp-server»
Nota, seleccionando «mysql-server-5.7» para la tarea «lamp-server»
Nota, seleccionando «php-common» para la tarea «lamp-server»
Nota, seleccionando «libaprutil1» para la tarea «lamp-server»
Nota, seleccionando «mysql-server» para la tarea «lamp-server»
Nota, seleccionando «php7.0-opcache» para la tarea «lamp-server»
Nota, seleccionando «libcgi-fast-perl» para la tarea «lamp-server»
Nota, seleccionando «libwrap0» para la tarea «lamp-server»
Nota, seleccionando «libhttp-date-perl» para la tarea «lamp-server»
Nota, seleccionando «perl-modules-5.22» para la tarea «lamp-server»
Nota, seleccionando «liblwp-mediatypes-perl» para la tarea «lamp-server»
Nota, seleccionando «libfcgi-perl» para la tarea «lamp-server»
Nota, seleccionando «libcgi-pm-perl» para la tarea «lamp-server»
Nota, seleccionando «libaprutil1-dbd-sqlite3» para la tarea «lamp-server»
Nota, seleccionando «php7.0-common» para la tarea «lamp-server»
```

Figura 42: Instalación LAMP metapaquete

Fuente : Propia

- ◆ Lo único que hay que hacer después de que se instalen todos estos paquetes será securizar MySQL, para ello se utilizará el comando :  
`# mysql_secure_installation`

```
root@Server_LAMP: /home/rivera
root@Server_LAMP:/home/rivera# sudo mysql_secure_installation

Securing the MySQL server deployment.

Enter password for user root:

VALIDATE PASSWORD PLUGIN can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD plugin?

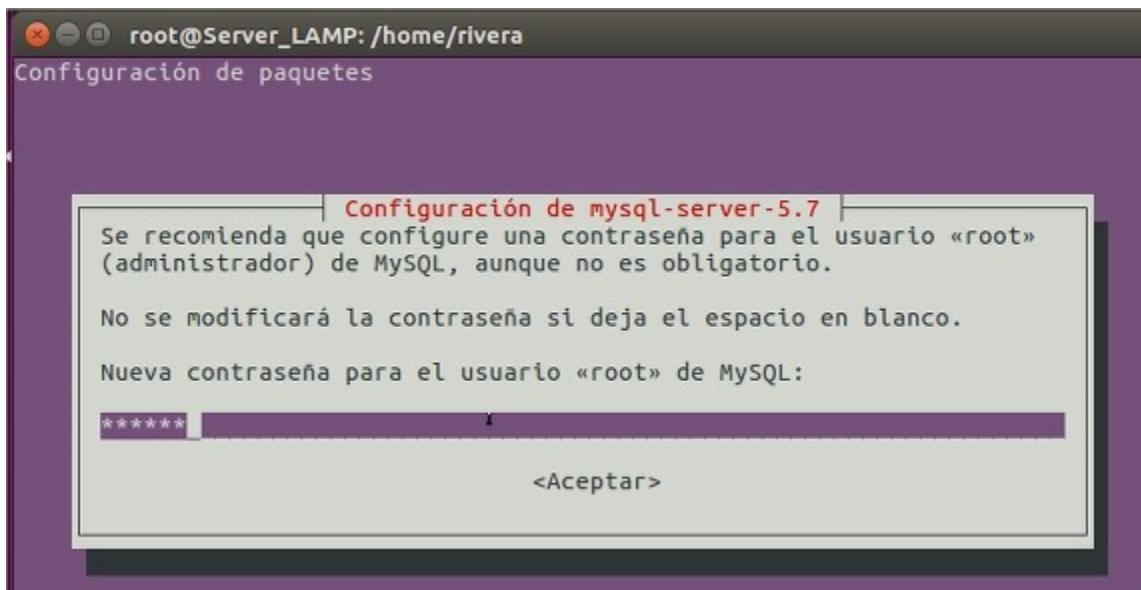
Press y|Y for Yes, any other key for No: y

There are three levels of password validation policy:

LOW      Length >= 8
MEDIUM  Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary
        file
```

*Figura 43: Securizar MySQL*

Fuente : Propia

*Figura 44: Contraseña MySQL*

Fuente : Propia

- ◆ En el directorio `/var/www/` es donde se guardará el proyecto web por defecto, para poder copiar los ficheros se creará un enlace simbólico a una carpeta en “home”, sin necesidad de editar permisos.



```
root@Server_LAMP: /home/rivera
root@Server_LAMP:/home/rivera# ln -s /var/www /home/rivera/www
root@Server_LAMP:/home/rivera#
```

Figura 45: Enlace simbólico

Fuente : Propia

- ◆ El siguiente paso será instalar phpmyadmin para crear, editar, borrar o realizar consultas en nuestra base de datos. Se utilizará el comando :

```
# apt-get install phpmyadmin
```

- ◆ Se configura el servidor web

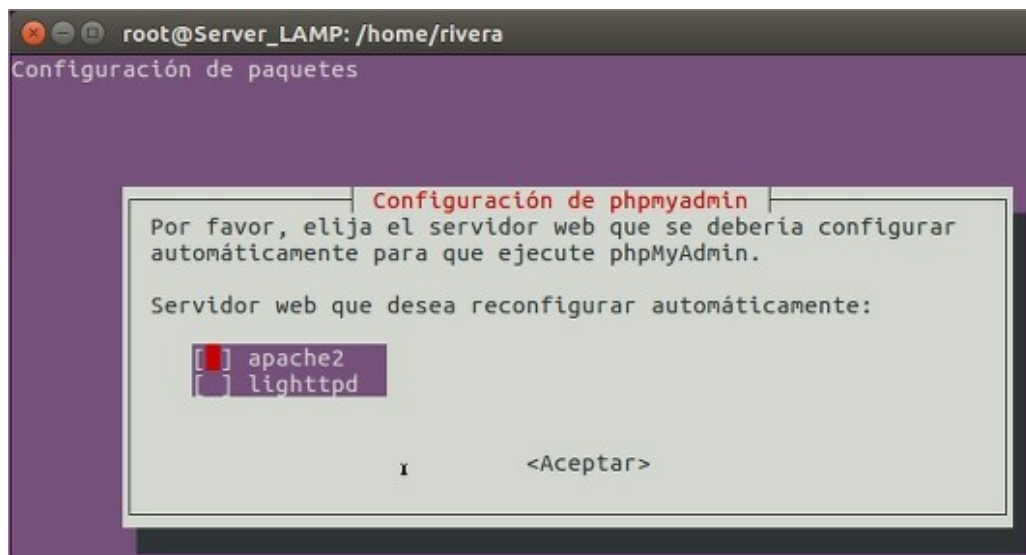


Figura 46: Servidor web

Fuente : Propia

- ◆ Después desde el navegador se escribe <http://localhost/phpmyadmin/> para tener acceso a phpmyadmin y se inicia sesión.



Figura 47: Inicio phpmyadmin

Fuente : Propia

- ◆ Una vez dentro ya se podrá gestionar la base de datos relacional.



Figura 48: Creación base de datos y tablas

Fuente : Propia

## 7 - Router Casa

- ◆ Al igual que con el router de empresa se crearan dos interfaces de red y se seguirán los mismos pasos, con una interfaz por DHCP que tendrá conexión con Internet haciendo NAT-P y otra estática que hará de puerta de enlace para el cliente.

```
# Red Externa
auto enp0s3
iface enp0s3 inet dhcp

# Red Casa
auto enp0s8
iface enp0s8 inet static
address 10.10.20.1
netmask 255.255.255.0
```

Figura 49: Interfaces router-casa

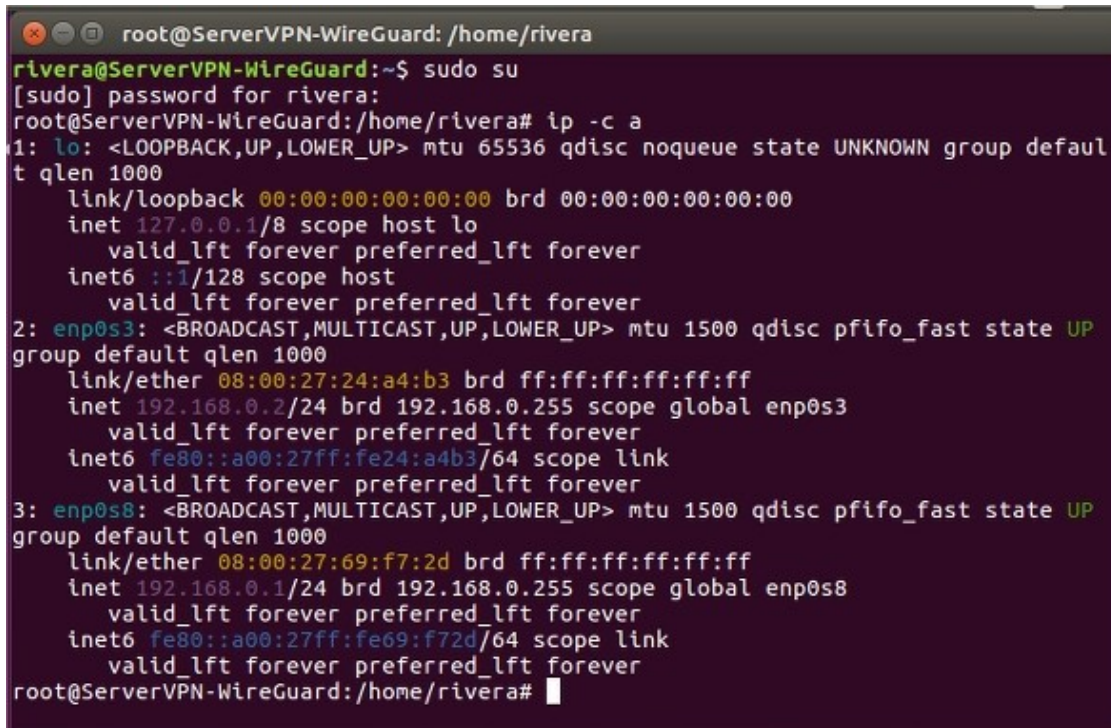
Fuente : Propia

## 8 - Server\_VPN\_WireGuard

- ◆ WireGuard está diseñado para poder realizar roaming de manera fácil y rápida. Si nuestro dispositivo cambia de redes, y lógicamente cambia de IP pública, como por ejemplo cuando pasamos de la red Wi-Fi y la red 4G/LTE de nuestro operador, la conexión VPN seguirá levantada porque se volverán a autenticar rápidamente con este servidor VPN, de tal forma que siempre estaremos conectados a la VPN.
- ◆ También se podrá habilitar el Kill-Switch en el dispositivo, de esta forma, si la conexión VPN se interrumpe, el propio software también se encargará de interrumpir todo el tráfico de red hasta que se vuelva a restablecer la conexión VPN, con el objetivo de que no se navegue sin la protección que brinda esta VPN.



- ◆ Para el servidor VPN se configuran dos interfaces estáticas.



```
root@ServerVPN-WireGuard: /home/rivera
rivera@ServerVPN-WireGuard:~$ sudo su
[sudo] password for rivera:
root@ServerVPN-WireGuard:/home/rivera# ip -c a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
    group default qlen 1000
    link/ether 08:00:27:24:a4:b3 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.2/24 brd 192.168.0.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe24:a4b3/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
    group default qlen 1000
    link/ether 08:00:27:69:f7:2d brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.1/24 brd 192.168.0.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe69:f72d/64 scope link
        valid_lft forever preferred_lft forever
root@ServerVPN-WireGuard:/home/rivera#
```

Figura 50: Interfaces Server\_VPN

Fuente : Propia

- ◆ Se añadirán los repositorios específicos de WireGuard, ya que actualmente por defecto no se encuentra en los paquetes de Ubuntu.

- Comandos necesarios para ello :

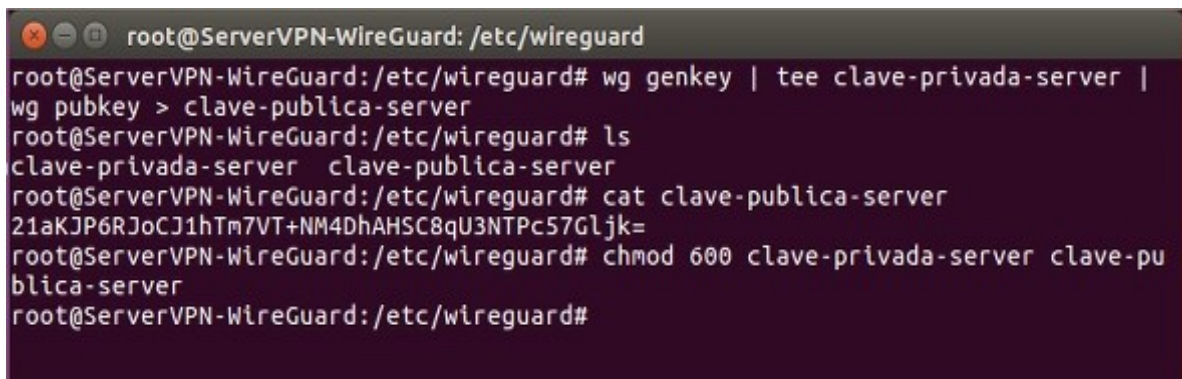
```
# sudo add-apt-repository ppa:wireguard/wireguard
```

```
# sudo apt-get update
```

```
# sudo apt-get install wireguard
```

- ◆ Cuando la instalación haya terminado se activará de forma permanente el reenvío de datagramas en el fichero `/etc/sysctl.conf` eliminando (#) de la línea `net.ipv4.ip_forward = 1` y se ejecuta el comando `# sysctl -p` para hacer efectivos los cambios, nos posicionaremos en el directorio creado por defecto : `# cd /etc / wireguard`

- ◆ En el directorio anterior habrá que generar las claves pública y privada como root.
- ◆ El comando para generar las claves será :  
`# wg genkey | tee clave-privada-server | wg pubkey > clave-publica-server`
- ◆ Con los comandos “ls “ y “cat “ se podrá ver si se han generado las claves.
- ◆ Después se cambiarán los permisos del usuario y para ello se utilizará el comando :  
`# chmod 600 clave-privada-server clave-publica-server`



```
root@ServerVPN-WireGuard: /etc/wireguard
root@ServerVPN-WireGuard:/etc/wireguard# wg genkey | tee clave-privada-server |
wg pubkey > clave-publica-server
root@ServerVPN-WireGuard:/etc/wireguard# ls
clave-privada-server  clave-publica-server
root@ServerVPN-WireGuard:/etc/wireguard# cat clave-publica-server
21aKJP6RJoCJ1hTm7VT+NM4DhAHSC8qU3NTPc57Gljk=
root@ServerVPN-WireGuard:/etc/wireguard# chmod 600 clave-privada-server clave-pu
blica-server
root@ServerVPN-WireGuard:/etc/wireguard#
```

Figura 51: Claves servidor VPN

Fuente : Propia

- ◆ Una vez generadas las claves correctamente se pasa a crear el archivo de configuración para el servidor (wg0.conf). En este archivo editaremos los parámetros necesarios para el correcto funcionamiento del servidor. Aunque todavía no se añadirán clientes.
  - Comando de creación:  
`# nano /etc/wireguard/wg0.conf`

**[Interfaces]**

**Address:** Define las direcciones IPv4 e IPv6 privadas para el servidor WireGuard.

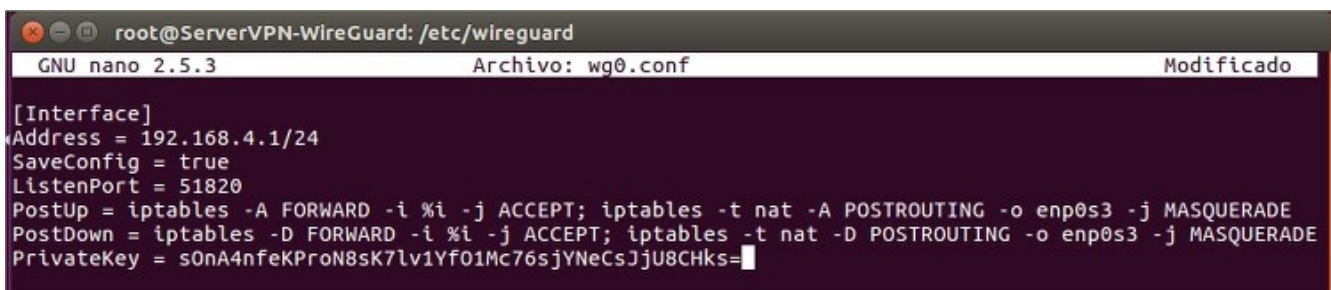
**ListenPort:** Especifica qué puerto utilizará WireGuard para las conexiones entrantes.

**PostUp y PostDown:** Define los pasos que se ejecutarán después de encender o apagar la interfaz con respecto a las iptables.

**SaveConfig:** Ordena al archivo de configuración que se actualice automáticamente cada vez que se agrega un nuevo par.

**PrivateKey:** Clave privada del servidor.

**\*\* Se creará la red virtual 192.168.4.0/24 y el puerto de escucha será el 51820 \*\***



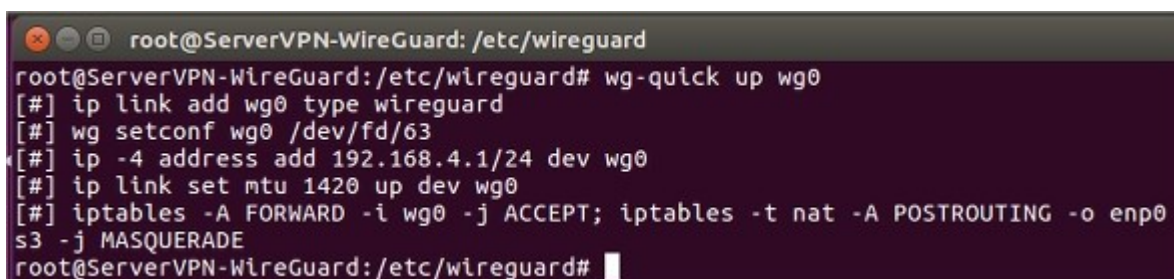
```
root@ServerVPN-WireGuard: /etc/wireguard
GNU nano 2.5.3 Archivo: wg0.conf Modificado
[Interface]
Address = 192.168.4.1/24
SaveConfig = true
ListenPort = 51820
PostUp = iptables -A FORWARD -i %i -j ACCEPT; iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
PostDown = iptables -D FORWARD -i %i -j ACCEPT; iptables -t nat -D POSTROUTING -o enp0s3 -j MASQUERADE
PrivateKey = s0nA4nfeKPr0n8sK7lv1Yf01Mc76sjYNeCsJjU8CHks=
```

Figura 52: Configuración inicial Wg0.conf

Fuente : Propia

- ◆ Para gestionar la interfaz se utilizará el comando : # wg-quick

# wg-quick up wg0 (habilitar)



```
root@ServerVPN-WireGuard: /etc/wireguard
root@ServerVPN-WireGuard:/etc/wireguard# wg-quick up wg0
[#] ip link add wg0 type wireguard
[#] wg setconf wg0 /dev/fd/63
[#] ip -4 address add 192.168.4.1/24 dev wg0
[#] ip link set mtu 1420 up dev wg0
[#] iptables -A FORWARD -i wg0 -j ACCEPT; iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
root@ServerVPN-WireGuard:/etc/wireguard#
```

Figura 53: Habilitar interfaz Wg0

Fuente : Propia

- ◆ Se podrá ver si está activa introduciendo : `# wg show`

```
root@ServerVPN-WireGuard: /etc/wireguard
root@ServerVPN-WireGuard:/etc/wireguard# wg show
interface: wg0
  public key: 21aKJP6RJ0CJ1hTm7VT+NM4DhAHSC8qU3NTPc57G1jk=
  private key: (hidden)
  listening port: 51820
root@ServerVPN-WireGuard:/etc/wireguard#
```

Figura 54: Interfaz Wg0 activa

Fuente : Propia

- ◆ Cuando se quiera hacer algún tipo de modificación en el archivo de configuración, como sería introducir un nuevo cliente, se tendrá que deshabilitar la interfaz.

```
root@ServerVPN-WireGuard:/etc/wireguard# wg-quick down wg0
[#] wg showconf wg0
[#] ip link delete dev wg0
[#] iptables -D FORWARD -i wg0 -j ACCEPT; iptables -t nat -D POSTROUTING -o enp0s3 -j MASQUERADE
```

Figura 55: interfaz Wg0 deshabilitada

Fuente : Propia

- ◆ Otra forma de gestionar la interfaz es como servicio en Systemd, con ello se logrará que se reinicie automáticamente en el arranque y los comandos necesarios serán :

```
# systemctl enable wg-quick@wg0.service
# systemctl start/stop wg-quick@wg0.service
# systemctl status wg-quick@wg0.service
```

```
root@ServerVPN-WireGuard: /etc/wireguard
Created symlink from /etc/systemd/system/multi-user.target.wants/wg-quick@wg0.service to /lib/systemd/system/wg-quick@wg0.service.
root@ServerVPN-WireGuard:/etc/wireguard# systemctl start wg-quick@wg0.service
root@ServerVPN-WireGuard:/etc/wireguard# systemctl status wg-quick@wg0.service
● wg-quick@wg0.service - WireGuard via wg-quick(8) for wg0
   Loaded: loaded (/lib/systemd/system/wg-quick@wg0.service; enabled; vendor preset: Active: active (exited) since dom 2020-06-07 05:10:26 CEST; 17s ago
   Docs: man:wg-quick(8)
          man:wg(8)
          https://www.wireguard.com/
          https://www.wireguard.com/quickstart/
          https://git.zx2c4.com/wireguard-tools/about/src/man/wg-quick.8
          https://git.zx2c4.com/wireguard-tools/about/src/man/wg.8
   Process: 2499 ExecStart=/usr/bin/wg-quick up %i (code=exited, status=0/SUCCESS)
   Main PID: 2499 (code=exited, status=0/SUCCESS)
```

Figura 56: Wg0 en Systemd

Fuente : Propia

- ◆ Llegados a este punto, solo quedaría añadir los parámetros de configuración de los clientes.
- ◆ Para poder conectarse, se añadirán estos parámetros relacionados y necesarios en el fichero “ wg0.conf “ añadiendo la sección [ Peer ] tanto en el servidor como en el cliente .

## 9 - Laptop\_Casa

- ◆ En este cliente se ha decidido implementar WireGuard en Windows 10 por su facilidad de instalación y por su sencilla configuración. Este se puede conseguir desde la web oficial de WireGuard <https://www.wireguard.com/install/>

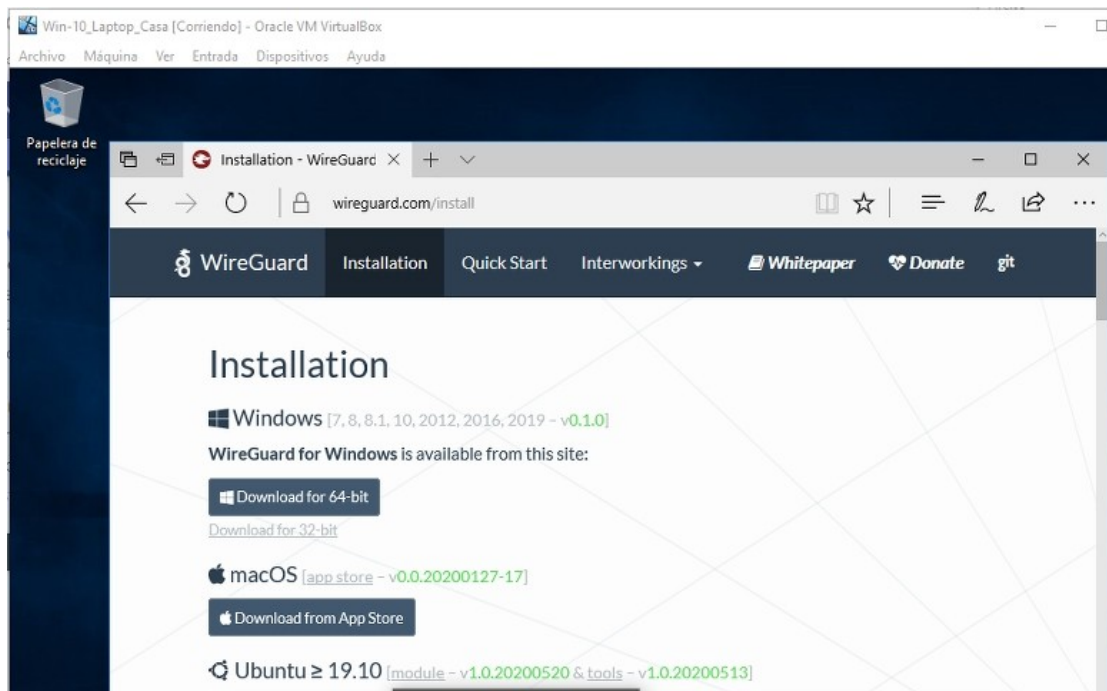
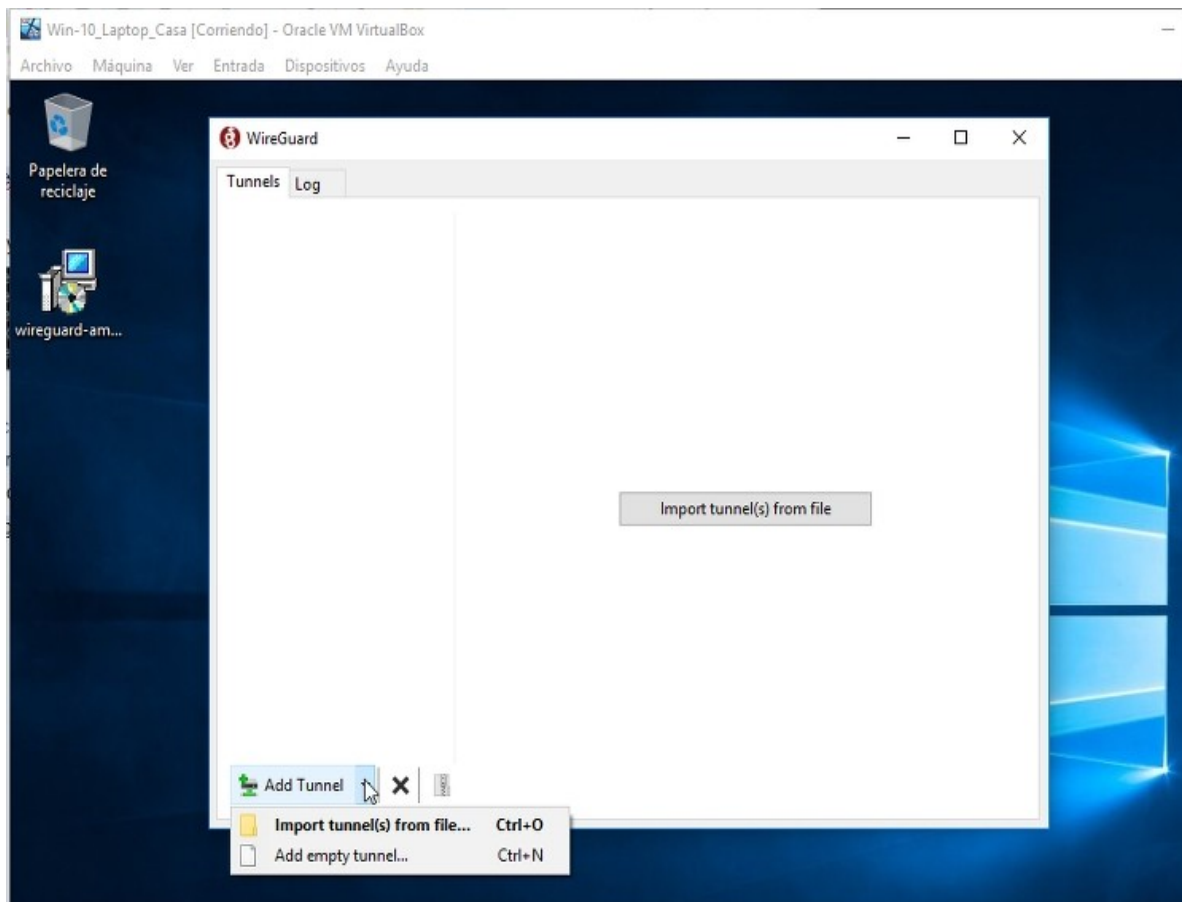


Figura 57: Instalación Windows 10

Fuente : Propia



- ◆ Una vez terminada la instalación se ejecutará el software de WireGuard y aparecerá un sencillo e intuitivo entorno gráfico, donde podremos importar un archivo de configuración o crearlo casi desde cero. En la siguiente imagen se puede apreciar su sencillez.



*Figura 58: Software WireGuard en Windows 10*

Fuente : Propia

- ◆ Para crear una configuración completa tanto en el cliente como en el servidor se necesitará intercambiar las claves públicas de ambos. Una forma correcta de hacerlo es utilizando SFTP como se muestra a continuación :



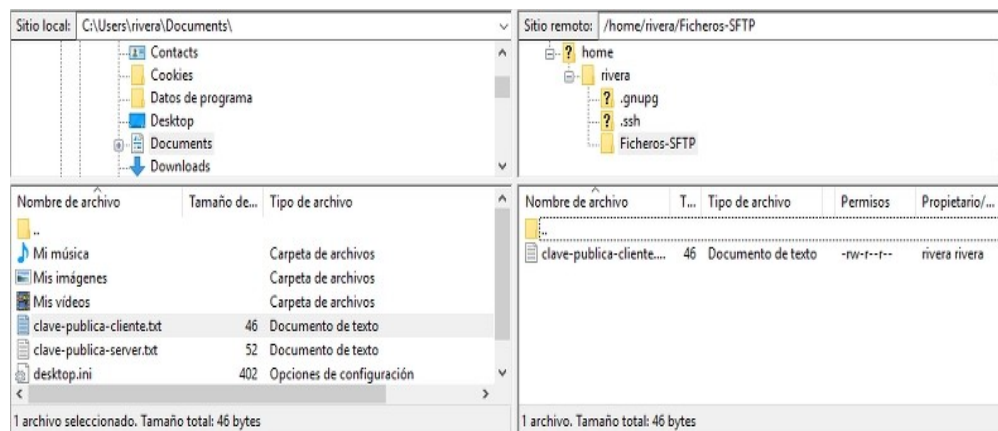


Figura 59: Intercambio de claves en Filezilla

Fuente : Propia

- ◆ Es el momento de crear el archivo de configuración en el cliente. Para ello en el menú de selección de “ Add Tunnel “ en el software de WireGuard se hace clic en :  
“ Add empty tunnel ”
- ◆ Lo primero que se puede ver es que el software ya ha generado la clave publica y la privada y se le tendrá que asignar un nombre al túnel e introducir los parámetros necesarios y su valor. Los parámetros podrán ser los siguientes, aunque algunos no serán obligatorios :

[Interface]

Address = < IP del cliente >

PrivateKey = < Clave privada del cliente >

DNS = < IP del DNS para el servidor VPN, así se evitan fugas DNS >

[Peer]

PublicKey = < Clave publica del servidor >

AllowedIPs = < Filtrará y enrutará el tráfico de una determinada IP o de todas utilizando 0.0.0.0/0 >

Endpoint = < IP publica del servidor VPN ( :) Puerto de escucha >

PersistentKeepalive = < Tiempo en segundos para mantener el túnel activo detrás del NAT o Firewall >

- ◆ La configuración del cliente con todos los parámetros que se utilizaran quedaría como en la siguiente imagen.

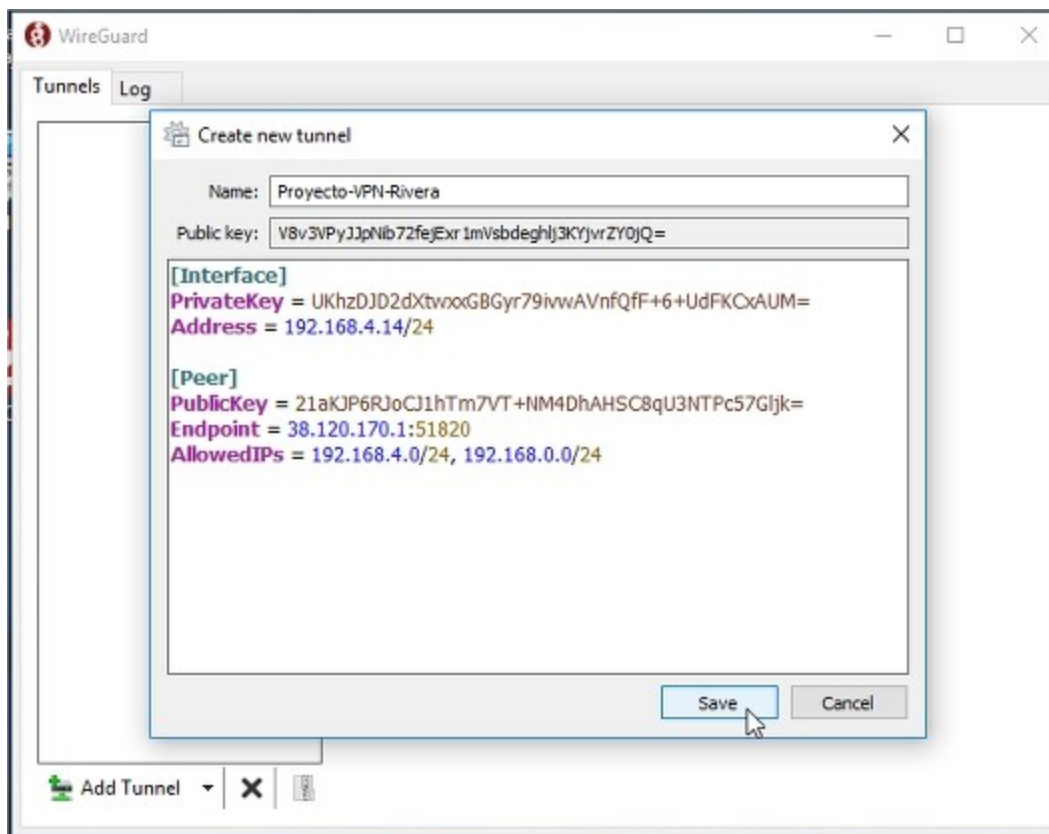


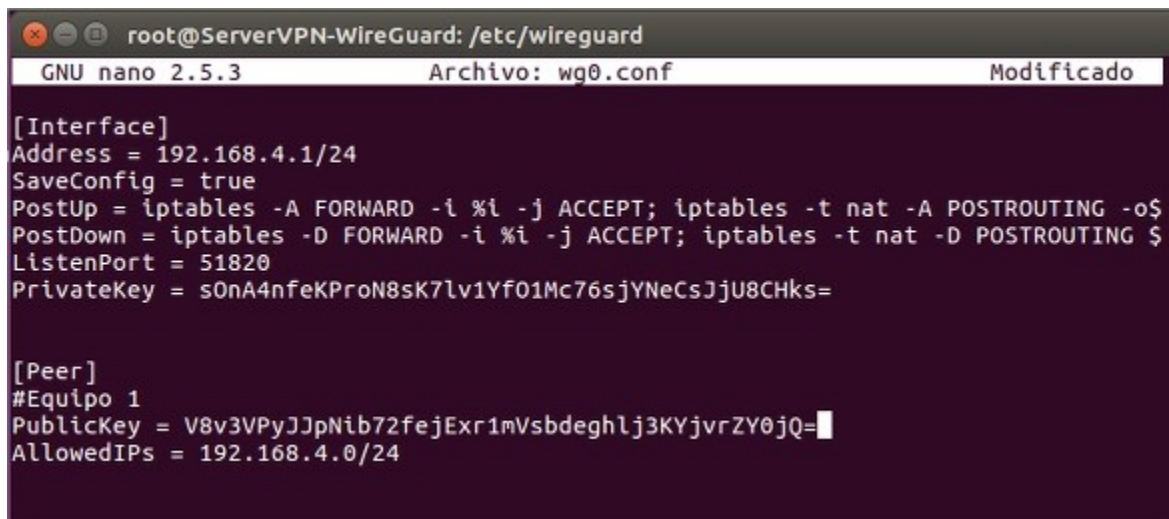
Figura 60: Configuración túnel cliente

Fuente : Propia

- ◆ Existe una opción “ Block untunneled traffic “. Esta opción agrega reglas de firewall para eliminar todo el tráfico que no viaja a través de la VPN en Windows.
- ◆ Se guarda la configuración y solo faltaría relacionar al cliente con su clave pública en el servidor.

### 5.1.2 CONEXIÓN DEL CLIENTE CON EL SERVIDOR VPN

- ◆ Al conectar un cliente con el servidor, el cliente ya habrá configurado su interfaz con los parámetros del servidor y generado sus claves. Es el momento de incluir al cliente en el archivo de configuración del servidor, para ello se editará el fichero “ Wg0.conf ”.
- ◆ En este fichero se añadirá la sección [Peer], una por cada cliente que se quiera conectar, donde se introducirán dos parámetros esenciales como son la clave publica del cliente ( PublicKey ) y ( AllowedIPs) la IP o rangos de IPs relacionados con dicha clave . Se utilizará el comando : `# nano /etc/wireguard/wg0`
- ◆ El fichero quedaría como la imagen siguiente :



```
root@ServerVPN-WireGuard: /etc/wireguard
GNU nano 2.5.3      Archivo: wg0.conf      Modificado

[Interface]
Address = 192.168.4.1/24
SaveConfig = true
PostUp = iptables -A FORWARD -i %i -j ACCEPT; iptables -t nat -A POSTROUTING -o $
PostDown = iptables -D FORWARD -i %i -j ACCEPT; iptables -t nat -D POSTROUTING $
ListenPort = 51820
PrivateKey = s0nA4nfeKPr0n8sK7lv1Yf01Mc76sjYNcSjJjU8CHks=

[Peer]
#Equipo 1
PublicKey = V8v3VPyJJpNib72fejExr1mVsbdeghlj3KYjvrZY0jQ=
AllowedIPs = 192.168.4.0/24
```

Figura 61: Añadir cliente a Wg0.conf

Fuente : Propia

- ◆ Levantaremos la interfaz virtual con el comando : `# wg-quick up wg0`
- ◆ Para ver el estado de la configuración realizada se puede utilizar el comando : `# wg showconf wg0`

- ◆ Si se necesita más información se puede consultar la ayuda :

# wg --help

```
root@ServerVPN-WireGuard:/etc/wireguard# wg --help
Usage: wg <cmd> [<args>]

Available subcommands:
show: Shows the current configuration and device information
showconf: Shows the current configuration of a given WireGuard interface, for use with 'setconf'
set: Change the current configuration, add peers, remove peers, or change peers
setconf: Applies a configuration file to a WireGuard interface
addconf: Appends a configuration file to a WireGuard interface
synconf: Synchronizes a configuration file to a WireGuard interface
genkey: Generates a new private key and writes it to stdout
genpsk: Generates a new preshared key and writes it to stdout
pubkey: Reads a private key from stdin and writes a public key to stdout
You may pass '--help' to any of these subcommands to view usage.
root@ServerVPN-WireGuard:/etc/wireguard#
```

Figura 62: Ayuda de Wg

Fuente : Propia

- ◆ Como se puede ver en la siguiente imagen, en el cliente se tiene que activar la interfaz.

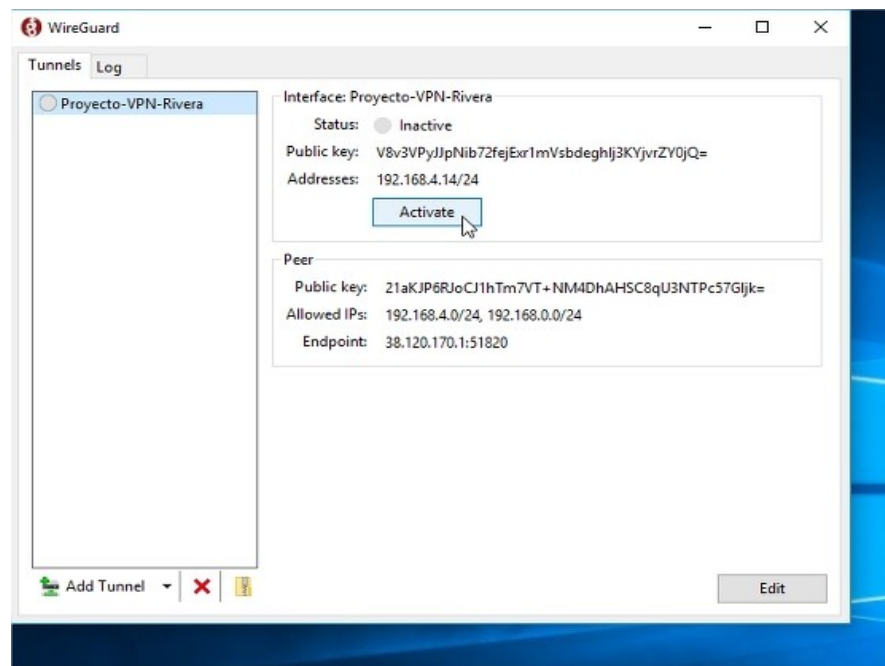
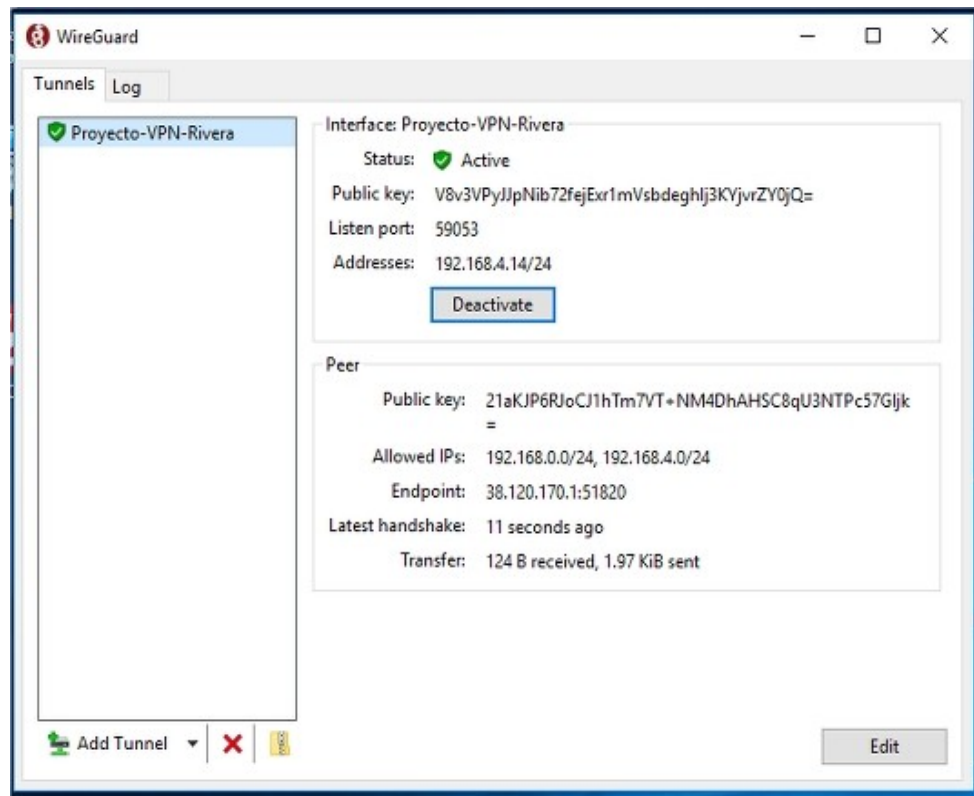


Figura 63: Activar interfaz cliente

Fuente : Propia

- ◆ Una vez activada, el túnel se creará comenzando la comunicación con el servidor VPN. Una vez recibido el paquete, lo descifra y comprueba la lista de pares autenticándolo si puede y si es así acepta el paquete.



*Figura 64: Cliente conectado correctamente*

Fuente : Propia

- ◆ Para obtener más información en el cliente se dispone de la pestaña de Logs, donde se puede consultar algunos detalles del proceso de conexión.

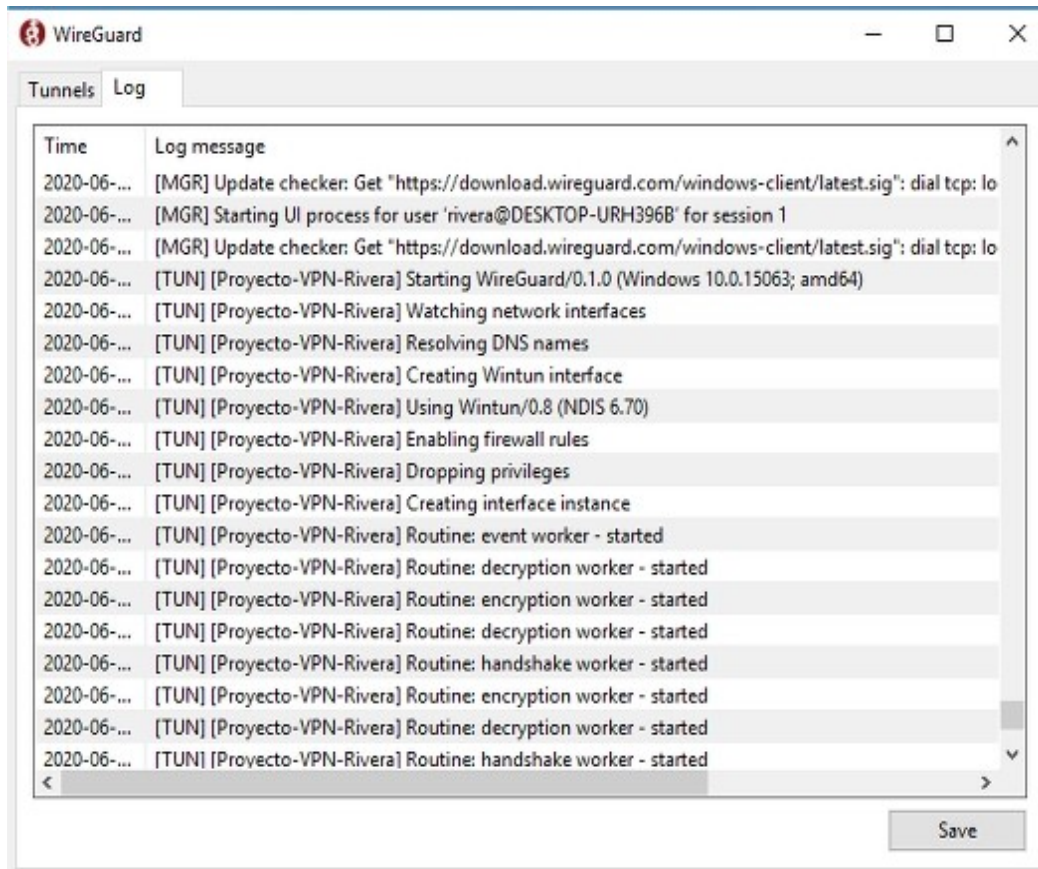


Figura 65: Logs

Fuente : Propia

- ◆ Si se tienen problemas con la conexión, y en el apartado de transferencia no hay paquetes recibidos, un detalle a tener en cuenta es la apertura de puertos en el cortafuegos. Tanto en el cliente como en el servidor.



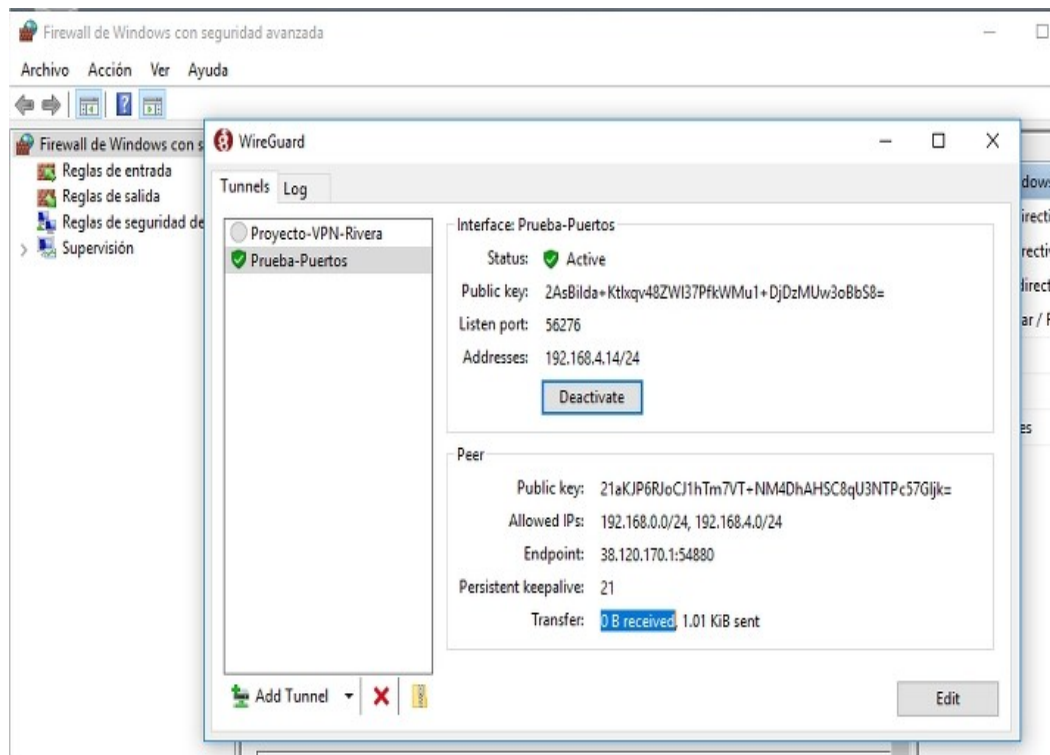


Figura 66: Túnel sin paquetes recibidos

Fuente : Propia

- ◆ Al encontrarnos en una situación en la cual el servidor VPN está situado detrás del cortafuegos se tendrán que configurar algunas reglas más con iptables. Se añadirán al script de cortafuegos-on las siguientes reglas :

# Permitimos el tráfico VPN entrante al puerto de escucha.

```
iptables -A FORWARD -p udp -m udp -dport 51820 -j ACCEPT
```

# Permitimos el reenvío de paquetes en el túnel VPN

```
iptables -A FORWARD -i wg0 -o wg0 -j ACCEPT
```

- ◆ En Windows se tendrá que acceder a la configuración avanzada y crear una regla para poder abrir el puerto UDP necesario para esa conexión.

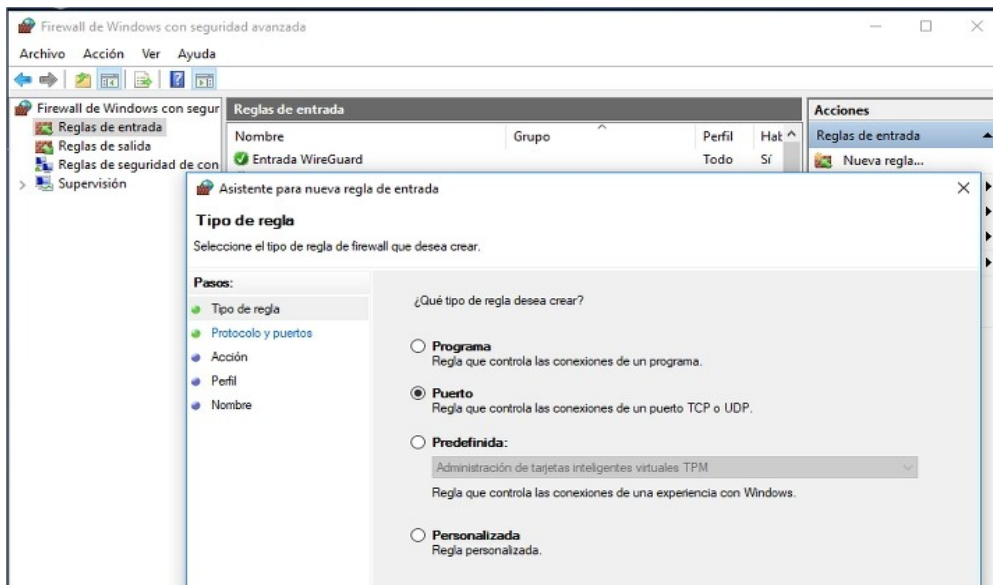


Figura 67: Abrir puertos en Windows

Fuente : Propia

- ◆ Un problema importante con muchas configuraciones de VPN es que el DNS tiene fugas. Esto termina filtrando la conexión del cliente y los detalles de ubicación. Una buena práctica sería instalar “Unbound<sup>14</sup>”, que es un producto de resolución de DNS de validación, recursivo y de almacenamiento en caché.
- ◆ Unbound ha suplantado a Berkeley Internet Name Daemon ( BIND ) como el servidor de nombres predeterminado del sistema base en FreeBSD y OpenBSD.

14 <https://nlnetlabs.nl/projects/unbound/about/>

## **6. CONCLUSIONES**

- ◆ Debido al rápido avance tecnológico y a las múltiples y continuas mejoras que vive el mundo de la informática, no podría asegurar que WireGuard esté entre nosotros tanto años como llevan OpenVPN o Ipsec, aunque actualmente este protocolo los mejora notablemente. Pienso que las pymes cada vez están más mentalizadas en invertir en tecnología y algunas de ellas ya han comenzado a digitalizarse.
- ◆ Con la llegada de la computación cuántica todo lo relacionado con las claves de cifrado pública y privada, como pueden ser los certificados de páginas web, el software de firmas digitales, la comunicación cifrada de la banca electrónica y otros tipos de aplicaciones, tiene un futuro incierto. Mientras un ordenador actual tardaría muchísimos años en descifrar la clave privada, los ordenadores cuánticos lo procesarían en unos pocos minutos. Por este motivo algunas organizaciones importantes como el Instituto Nacional de Estándares y Tecnología están estudiando nuevos algoritmos para remplazar el sistema RSA o el protocolo ECDH.
- ◆ A nivel personal este proyecto me ha servido para afianzar los conocimientos adquiridos en los dos años del ciclo, algo que me mueve a mostrar más interés por seguir aprendiendo y hace que me sienta muy satisfecho por haber elegido estos estudios.

### **6.1 LIMITACIONES ENCONTRADAS**

- ◆ Al realizar este proyecto he tenido algunos problemas con la virtualización al disponer solo de un ordenador portátil y no disponer de tanta memoria RAM. También al ser WireGuard un protocolo relativamente nuevo surgieron algunos fallos con los repositorios. Otras complicaciones han sido las relacionadas con la configuración del cortafuegos ASA en Packet Tracer.

## **7. ÍNDICE DE FIGURAS**

### **Índice de figuras**

Figura 1: Comparativa protocolos.....	6
Figura 2: Diagrama de Gannt.....	7
Figura 3: Topología de red (PYME).....	8
Figura 4: Máquinas creadas en VirtualBox.....	9
Figura 5: Fichero /etc/network/interfaces.....	11
Figura 6: ACLs Packet tracer.....	12
Figura 7: Fichero resolv.conf.....	12
Figura 8: Reinicio de servicio.....	13
Figura 9: Fichero /etc/network/interfaces.....	13
Figura 10: Fichero script “ cortafuegos-on “.....	14
Figura 11: Fichero script “ cortafuegos-off “.....	15
Figura 12: Fichero del servicio.....	15
Figura 13: Habilitar servicio.....	16
Figura 14: Fichero /etc/network/interfaces.....	16
Figura 15: Comando instalar bum.....	17
Figura 16: Herramienta gráfica “ boot-up manager.....	18
Figura 17: Fichero /etc/dnsmasq.conf.....	18
Figura 18: Fichero /etc/dnsmasq.conf.....	18
Figura 19: Fichero named.conf.local.....	19
Figura 20: Fichero zona directa.....	20
Figura 21: Fichero zona inversa.....	20
Figura 22: Comprobar zonas.....	21
Figura 23: Comprobar named.conf.....	21
Figura 24: URI servidor.....	22
Figura 25: Dominio y cuenta admin.....	23

Figura 26: Fichero /etc/network/interfaces.....	24
Figura 27: Creación de carpeta y estado de la conexión.....	25
Figura 28: Autorizar claves.....	26
Figura 29: Estado del servicio ssh.....	26
Figura 30: Generar claves ssh.....	27
Figura 31: Copiar clave publica.....	27
Figura 32: Conexión al servidor SFTP.....	27
Figura 33: Conexión al servidor SFTP.....	28
Figura 34: Conexión al servidor SFTP.....	29
Figura 35: Configuración SquirrelMail.....	30
Figura 36: Configuración Imap server.....	30
Figura 37: Configuración del dominio.....	31
Figura 38: Enlace simbólico.....	31
Figura 39: Reinicio Apache.....	31
Figura 40: Cliente de correo web.....	32
Figura 41: Fichero /etc/network/interfaces.....	32
Figura 42: Instalación LAMP metapaquete.....	33
Figura 43: Securizar MySQL.....	34
Figura 44: Contraseña MySQL.....	34
Figura 45: Enlace simbólico.....	35
Figura 46: Servidor web.....	35
Figura 47: Inicio phpmyadmin.....	36
Figura 48: Creación base de datos y tablas.....	36
Figura 49: Interfaces router-casa.....	37
Figura 50: Interfaces Server_VPN.....	38
Figura 51: Claves servidor VPN.....	39
Figura 52: Configuración inicial Wg0.conf.....	40
Figura 53: Habilitar interfaz Wg0.....	40
Figura 54: Interfaz Wg0 activa.....	41
Figura 55: interfaz Wg0 deshabilitada.....	41

Figura 56: Wg0 en Systemd.....	41
Figura 57: Instalación Windows 10.....	42
Figura 58: Software WireGuard en Windows 10.....	43
Figura 59: Intercambio de claves en Filezilla.....	44
Figura 60: Configuración túnel cliente.....	45
Figura 61: Añadir cliente a Wg0.conf.....	47
Figura 62: Ayuda de Wg.....	47
Figura 63: Activar interfaz cliente.....	48
Figura 64: Cliente conectado correctamente.....	49
Figura 65: Logs.....	50
Figura 66: Túnel sin paquetes recibidos.....	51
Figura 67: Abrir puertos en Windows.....	52



## **8. GLOSARIO**

- **ACLs:**  
Lista de control de acceso. Las ACL permiten controlar el flujo del tráfico en equipos de redes, tales como enrutadores y conmutadores.
- **BLAKE2:**  
Función hash criptográfica rápida y segura.
- **ChaCha20:**  
Cifrado de flujo. Un cifrado de flujo es un cifrado de clave simétrica.
- **Curve25519:**  
Es una curva elíptica que ofrece 128 bits de seguridad y está diseñada para usarse con el esquema de acuerdo clave de la curva elíptica Diffie-Hellman (ECDH).
- **DMZ:**  
En seguridad informática una zona desmilitarizada o red perimetral es una red local que se ubica entre la red interna de una organización y una red externa.
- **ECDH:**  
Es un protocolo de establecimiento de claves anónimo.
- **HKDF:**  
Función para derivar claves de diferente longitud.
- **IMAP:**  
Protocolo de acceso a mensajes de Internet.
- **NAT-P:**  
Traducción de Direcciones de Red por Puerto.
- **Noise protocol framework:**  
Protocolos criptográficos simples, rápidos y seguros
- **Poly1305:**  
Es un código de autenticación de mensajes de clave secreta de última generación adecuado para una amplia variedad de aplicaciones.
- **POP3:**  
(Protocolo de Oficina Postal) en clientes locales de correo para obtener los mensajes de correo electrónico almacenados en un servidor remoto.

- **Roaming:**  
Concepto utilizado en telecomunicaciones para referirse a la posibilidad de un dispositivo inalámbrico de utilizar una cobertura de red distinta de la principal.
- **SipHash24:**  
Es una familia de funciones pseudoaleatorias (también conocidas como funciones hash con clave) optimizadas para la velocidad en mensajes cortos.
- **UDP:**  
Es un protocolo del nivel de transporte basado en el intercambio de datagramas
- **UNIX:**  
Es un sistema operativo portable, multitarea y multiusuario.
- **VPN:**  
Virtual Private Network, VPN es una tecnología de red de ordenadores que permite una extensión segura de la red de área local (LAN) sobre una red pública o no controlada como Internet.

## 9. BIBLIOGRAFÍA

- Libro de Redes Locales e Internet (conceptos y práctica)  
Arturo Mora Rioja, M.ª Mercedes Rodríguez Villafáfila
- Libro de Servicios de Red e Internet  
Grupo editorial Garceta
- Apuntes y Prácticas 2º ASIR
- <https://www.incibe.es/>
- <https://www.wireguard.com/> <https://www.wireguard.com/performance/>
- <https://www.cisco.com/c/en/us/products/security/vpn-endpoint-security-clients/what-is-vpn.html>
- <https://www.opendns.com/>
- [https://wiki.archlinux.org/index.php/Systemd\\_\(Español\)](https://wiki.archlinux.org/index.php/Systemd_(Español))
- [https://wiki.archlinux.org/index.php/WireGuard#Manual\\_WireGuard\\_setup](https://wiki.archlinux.org/index.php/WireGuard#Manual_WireGuard_setup)
- [https://www.gamificafp.com/pluginfile.php/8111/mod\\_resource/content/0/2-Servidor-DHCP-y-DNS.pdf](https://www.gamificafp.com/pluginfile.php/8111/mod_resource/content/0/2-Servidor-DHCP-y-DNS.pdf)
- <http://somebooks.es/administrar-servicios-demonios-de-ubuntu-con-boot-up-manager/>
- <https://www.ssh.com/ssh/sftp/>
- <https://www.ecured.cu/RSA>
- [https://wiki.archlinux.org/index.php/Postfix\\_\(Español\)](https://wiki.archlinux.org/index.php/Postfix_(Español))
- <https://wiki.archlinux.org/index.php/Dovecot>
- <https://wiki.archlinux.jp/index.php/Squirrelmail>
- <https://www.linuxadictos.com/que-son-los-meta-paquetes-de-linux.html>
- <https://www.wireguard.com/install/>

## **9. ANEXO**

- Topología de Red Packet Tracer ( Archivo .pkt )
- Script (Cortafuegos-on)