Redes Privadas Virtuales

Preguntas

1. ¿Qué es una Red Privada Virtual?

Una red privada virtual permite "unir" o "extender" varias redes privadas separadas físicamente por una red pública (Internet), como si fueran una única red privada.

Para ello los paquetes de una red privada que van hacia otra red privada utilizarán protocolos de túnel que autentificarán todos los extremos y cifran la información para que atraviese Internet sin peligro. Algunos de estos protocolos de túnel son GRE, L2PT, OpenVPN (TLS) , IPSec, WireGuard, etc.

Los encargados de realizar tales tareas se llaman "VPN endpoints" y pueden ser encaminadores, computadoras, o hardware especializado.

Cada paquete que sale de una red privada hacia otra red privada es "encapsulado" dentro de otro paquete, por el "VPN endpoint". Es decir, el paquete a enviar forma parte de la información enviada dentro de un paquete más grande. Al llegar al otro extremo, dicha información se desempaqueta y circula por la correspondiente red.

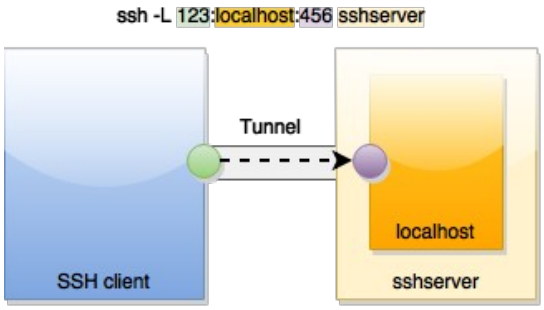
Existen dos tipos de redes privadas virtuales, según el acceso:

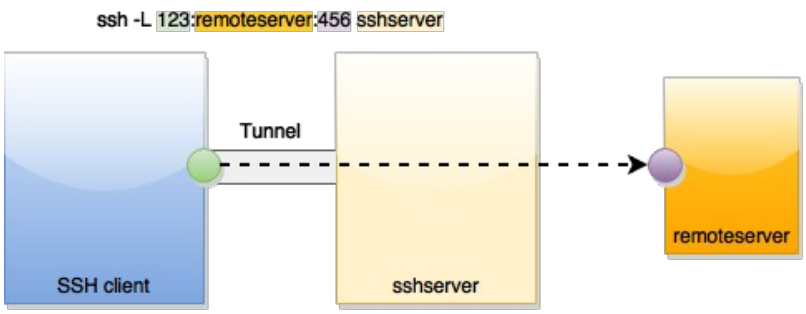
- Site-to-Site VPN: Las que unen varias redes privadas.

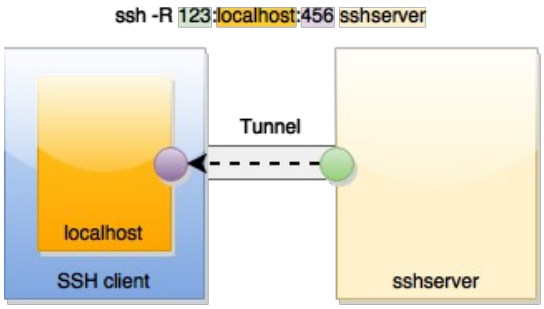
- Host-to-Site o Remote Access VPN: las que unen individualmente empleados en lugares remotos con la red privada de la empresa.

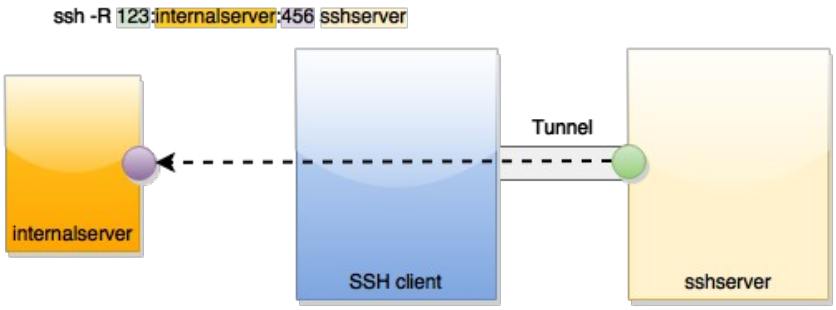
2. ¿Qué es un tunel SSH?

Un tunel SSH consiste en utilizar una conexión SSH para enviar tráfico de una aplicación (-normalmente no encriptado-) por un canal encriptado. También sirve para pasar el filtrado de cortafuegos y proxies.









3. ¿Qué es IPSec?

Es un protocolo de encapsulamiento que funciona en la capa 3 de los niveles OSI.

- More widely used in industry

- Available with proprietary routers (most proprietary routers do not support OpenVPN)

- Arguably more secure, since OpenVPN users can (and sometimes do) set their passwords empty, allowing a connection without a passphrase.

- Formally standardized via IETF RFC 3193

- De Facto standard for Microsoft products.

4. ¿Qué es OpenVPN?

Es un protocolo de encapsulamiento que en el servidor funciona en la capa 3 de los niveles OSI (puerto 1994/udp) y que usa SSL/TLS. Permite el uso de certificados para autentificar a los extremos, pero también autentificar mediante usuario y contraseña.

- Easier to set up and configure

- Less likely to be blocked by intermediate routers

- Much better for site-to-site connections (where an entire network is connected to another network)

- Ability to do Ethernet-layer tunneling (not possible with IPSec)

- More stable, and troubleshooting is generally simpler.

- Standard for OpenSource projects

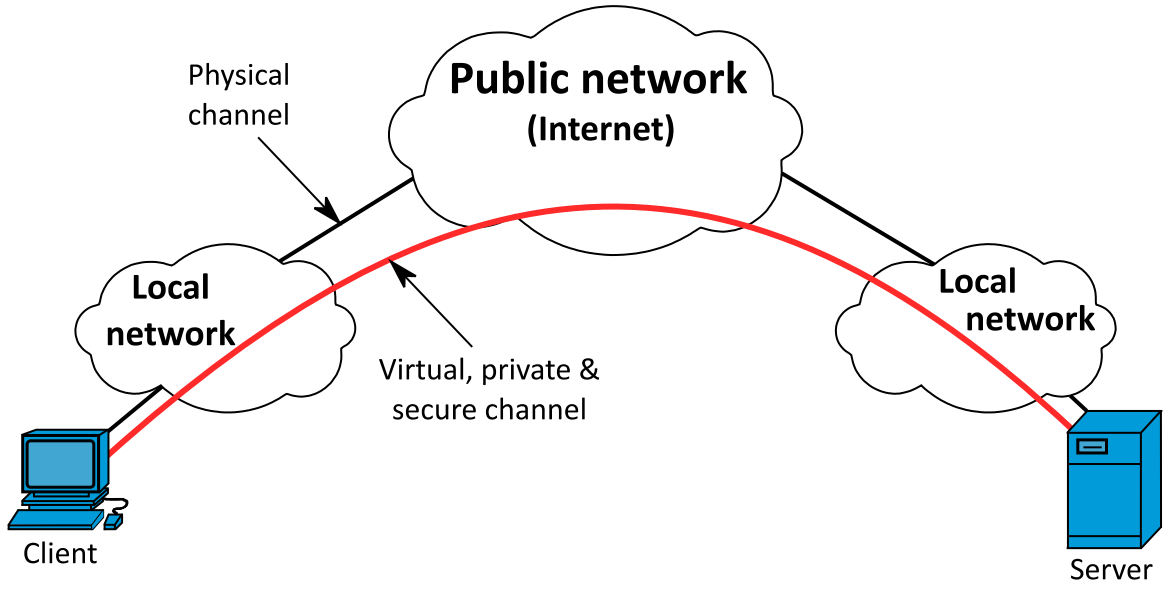
5. ¿Qué es Wireguard?

Es un protocolo de encapsulamiento que funciona en la capa 3 de los niveles OSI, mediante datagramas udp. De reciente aparición, es más rápido, estable y eficiente que IPSec y

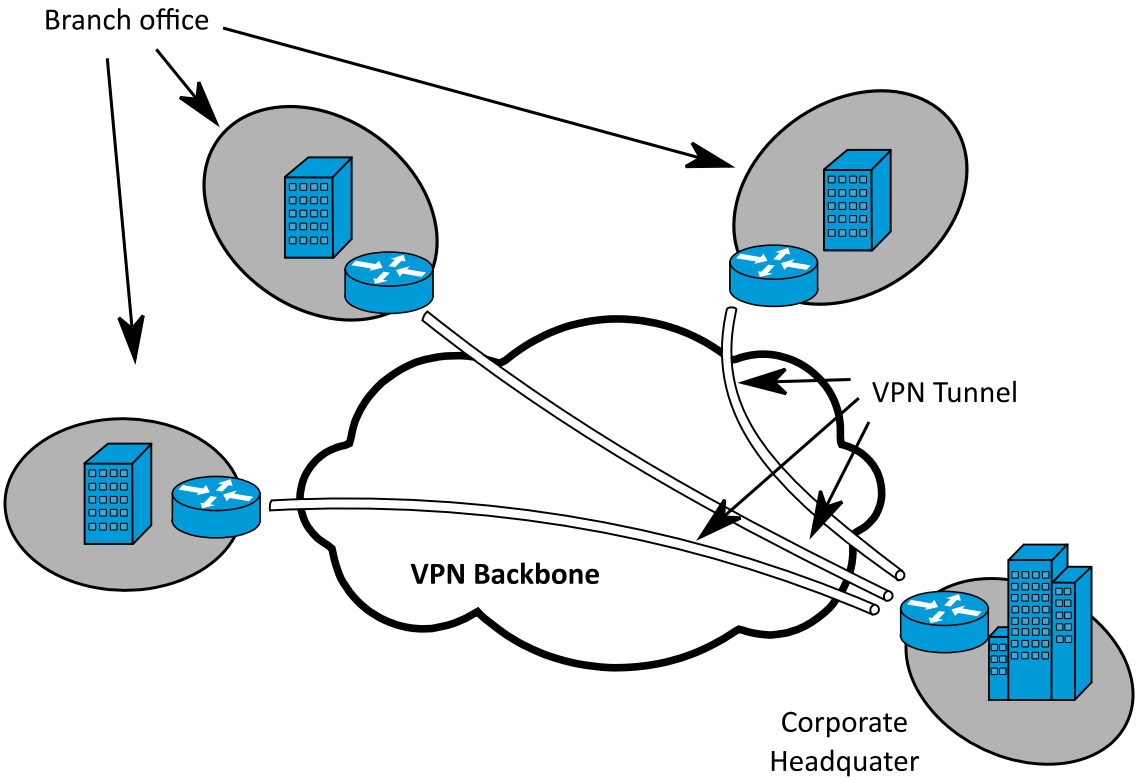
OpenVPN, y con un código mejor auditado. También tiene mejor encriptación y es más simple de configurar. Se espera que en breve los reemplace. Se ha incorporado como módulo dentro de los núcleos de Linux por encima del kernel 5.6 y también a los núcleos BSD a partir de FreeBSD 13.

6. Dibuja el esquema de varias redes físicas distantes que se comunican mediante un tunel VPN.

El túnel VPN puede unir de manera segura un equipo fuera de la red de la empresa con dicha red, como si realmente estuviera dentro de ella.



El túnel VPN puede unir también las redes locales de varias sedes en una única red lógica segura.



7. ¿Viene algún software de servidor de VPN con Windows Server? ¿Desde dónde se instala y desde dónde se administra? ¿Y para Windows no server?

En Windows Server debes instalar el rol de acceso remoto, en el panel de administración del servidor.

Para conectarte des de los clientes Windows, en éstos debes ir a Conexión de red y decir que quieres una conaxión VPN nueva.

8. ¿Cuál es el software de servidor de VPN más conocido para Unix/Linux?

En Linux está disponible OpenVPN para cualquier distribución, y Wireguard en cualquier núcleo 5.6 o superior.

Además hay numerosas distribuciones Linux y BSD especializadas en hacer de router y extremo VPN que se configuran fácilmente via web: IPFire, OpnSense, PfSense, etc.

9. Al instalar una VPN , ¿Qué parámetros a configurar piensas que serán los más importantes?

Básicamente deberemos generar certificados para autentificar a las partes.

10. Al instalar un servidor de VPN en una red local, ¿Qué pruebas piensas que deberás hacer para comprobar el buen funcionamiento?

- Comprobar antes de crear la VPN que el cliente y el servidor se "ven" haciendo ping de uno a otro.

- Comprobar que cliente y servidor han recibido direcciones ip compatibles en su nueva interfaz de red de "túnel", y que el cliente puede hacer ping a ordenadores de la red local.

- Si no es así, comprobar en los logs del servidor VPN qué falló en la autentificación del cliente.

Datos de la práctica "Host-to-Site VPN"

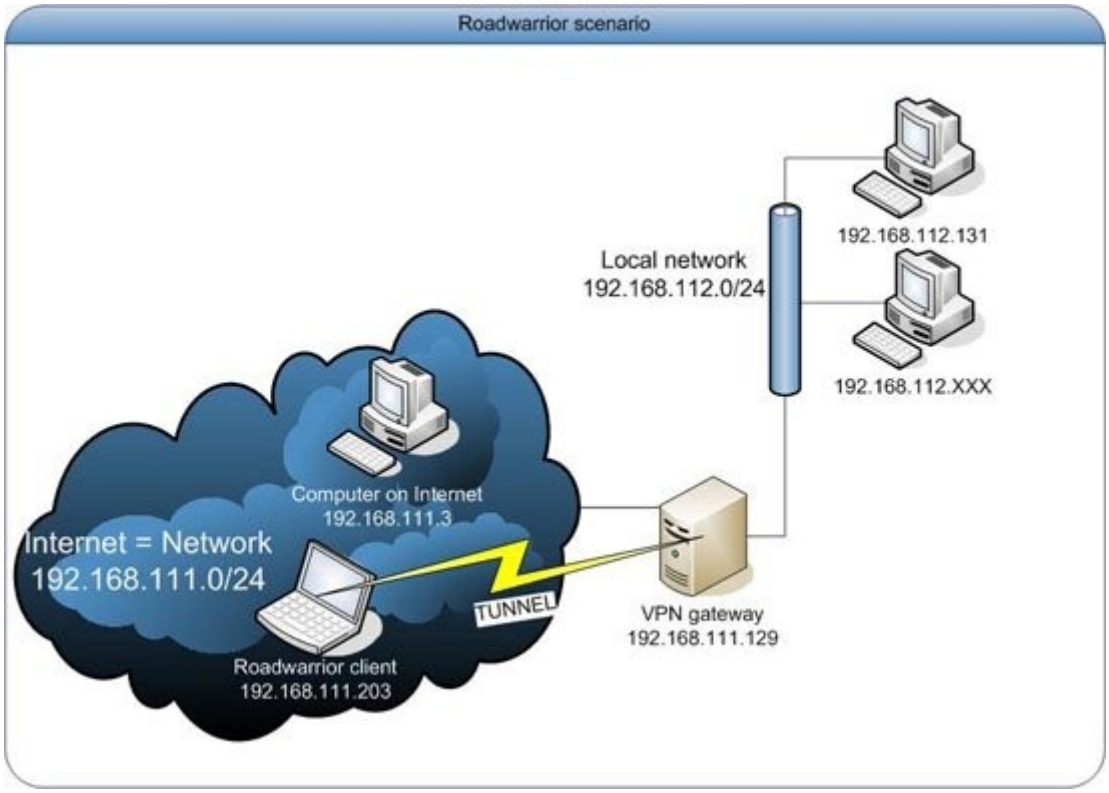
Crea el siguiente montaje de red para conectar un "roadwarrior" a la red de la empresa por "mobile VPN":

- Una máquina virtual que será un endpoint VPN, con dos tarjetas de red:
  - Una tarjeta de red para la red de la empresa (LAN) en red interna. Escoge la IPs que quieras.
  - Una tarjeta de red para Internet (WAN) en modo puente.
- Una máquina virtual con servicio ssh y web que estará en la red de la empresa. Escoge su IP.
- El cliente que creará un túnel con el endpoint y que intentará acceder a la web y ssh del servidor. Estará conectada a Internet (WAN) en modo puente.

  Dicho cliente puede ser:

- Otra máquina virtual en el mismo anfitrión, en red con el endpoint VPN.
- Una máquina virtual de un compañero, en otro anfitrión.
- Tu teléfono móvil con una app gratuita y confiable para conexión remota.

Para hacerte una idea del montaje, mira las imágenes (tus rangos de IP pueden variar):



Puedes hacer la práctica por parejas.

Una opción es utilizar como VPN endpoint una distribución Linux o BSD especializada, como IPFire, OPNsense o pfSense, que puedes administrar fácilmente des de un navegador.

Otra opción es utilizar como VPN endpoint un Linux e instalar y administrar a pelo OpenVPN.

Otra opción es utilizar Windows tanto en el cliente como en endpoint VPN.

Práctica: "Host-to-Site VPN" con Windows

Instalaremos VPN para Windows con las opciones de Microsoft que incorpora en Windows Server y Windows 10, sin instalar software adicional para el VPN y conseguirlo. En este caso el cliente será miembro del dominio, y se pueden aplicar GPOs al cliente que se conecta por VPN.

- https://msftwebcast.com/2020/02/how-to-setup-l2tp-ipsec-vpn-on-windows-server-2019.html
- https://www.youtube.com/watch?v=eTzHH8CQX_8

1. Instalamos el rol "Remote Access Server":

   Server role → remote acces

Features → DirectAccess and VPN

2. Configuramos el servicio “Routing and Remote Access”:

   Configure Remote Access , Getting Started Wizard

   - Deploy VPN Only

   - Configure and Enable Routing and Remote Access

   - Custom configuration

   - VPN access

   - Start Service

   - Server → Properties → IPv4 → Static Address pool → add 10.10.10.1 to 10.10.10.10

   - Ports → Properties

3. Creamos las cuentas de usuarios y grupos en el servidor:

   Active directory Users and Computers

   - Crear usuarios y grupo de prueba

   - Usuario -> propiedades -> Dial-in -> Control access through NPS Network Policy

4. Activas la política de acceso a la red (NPS):

   Tools → Network Policy Server → Policies → Network Policies

   - New → Type: Remote Access Server (VPN-Dial up) → next

   - Specify Condition → add → Windows Groups → grup dels usuaris → Access Granted → uncheck less secure MSCHAP y MSCHAPv2

   - Constrains → Authentication methods → EAP → add EAP MSCHAPv2

5. Configuremos el reenvío de puertos en el router:

   En el router debemos habilitar forwarding de PPTP (puerto tcp/1723 ip servidor) y protocolo 47 GRE (PPTP passthrough).

6. Probamos la funcionalidad VPN des de un cliente:

   En Windows 10 → Network Connections → VPN → Add VPN connection

   - Provider: Windows

   - Server name or address: IP pública del server (la IP del router)

   - VPN type: PPTP

   Clica en el nuevo icono de la VPN y selecciona "connect"

   - username@dominio + password → ¡ya estás dentro!

   En el servidor puedes ver como están las cosas:

   - en Tools → Routing and Remote Access →Remote Access Clients → ver la conexión

- en Tools → Routing and Remote Access →Ports -> Status → ver la IP

En el cliente puedes ver como están las cosas:

- en Network Connections → icono de la VPN → Status → Details

Práctica: "Mobile VPN" con Linux/BSD

- Con OpenVPN y pfSense: https://www.youtube.com/watch?v=K26Pir7xu5s
- Con OpenVPN y OPNsense: https://www.youtube.com/watch?v=YYCUbA8vpDY
- Con OpenVPN y IPFire: https://www.youtube.com/watch?v=pMMxZEjL-k8

A pesar de lo fácil e intuitivo que es montar una VPN con una distribución especializada para encaminadores y que se administra vía navegador web, como IPFire, OPNsense o pfSense, montaremos una VPN "a pelo" con OpenVPN sobre Debian 11, y nos conectaremos des de un cliente Linux con NetworkManager. Teóricamente, el proceso no debería ser mucho más complicado que instalar el programa, ajustar algunos parámetros y generar el certificado para el cliente que se conectará. Utilizaré como referencia la información que se encuentra en:

- https://wiki.archlinux.org/index.php/OpenVPN
- https://wiki.archlinux.org/index.php/Easy-RSA

Aviso: para realizar esta práctica debes tener claro el montaje de red que vas a realizar con las máquinas virtuales.

- Para simular la red local, el servidor VPN tendrá una tarjeta de red en red interna con una máquina que tenga algún servicio, por ejemplo HTTP o SSH. El cliente externo a la red local no será capaz de acceder a dichos servicios sin conectarse a la VPN, pero será capaz de acceder cuando se haya conectado. Para la red local podemos utilizar el rango de IPs del anterior diagrama, 192.168.112.0/24, u otro como por ejemplo 172.16.0.0/24.

- Para simular Internet, el servidor VPN tendrá otra tarjeta de red con el cliente que se conectará a la VPN. Para Internet podemos utilizar el rango de IPs del anterior diagrama, 192.168.111.0/24, u otro como por ejemplo 192.168.206.0/24.

- Dentro del tunel, el cliente y el servidor VPN tendrán otras IPs asociadas a la tarjeta de red *tun0* virtual que se crea para el tunel, por ejemplo 10.8.0.0/24.

No continúes la práctica hasta que tengas un diagrama claro con el montaje que vas a realizar.

1. Activamos el enrutamiento:

```
# nano /etc/sysctl.conf
```

```
net.ipv4.ip_forward=1
```

```
# sysctl -p
```

2. A continuación instalamos el programa:

   # **apt update**

   # **apt install openvpn**

   # **cp -r /usr/share/easy-rsa /etc/openvpn/**

3. Generamos la autoridad certificadora, la clave y el certificado para el servidor:

   # **cd /etc/openvpn/easy-rsa**

   # mv vars.example vars

   # nano vars

```
set_var EAYRSA_REQ_COUNTRY="ES"
set_var EAYRSA_REQ_PROVINCE="Barcelona"
set_var EAYRSA_REQ_CITY="Barcelona"
set_var EAYRSA_REQ_ORG="mired"
set_var EAYRSA_REQ_EMAIL="admin@mired.org"
set_var EAYRSA_REQ_OU="ASIX"
```

   Inicializamos la infraestructura de clave pública:

   # **./easyrsa init-pki**

   Creamos la autoridad certificadora, sin contraseña (en “common name” respondí “server”):

   # **./easyrsa build-ca nopass**

   Generamos la clave privada para el servidor:

   # **./easyrsa gen-req *servidor* nopass**

   Firmamos el certificado del servidor:

   # **./easyrsa sign-req server *servidor***

   Creamos las clave Diffie-Hellman para el intercambio de la clave simétrica de cifrado:

   # **./easyrsa gen-dh**

   Generamos la firma HMAC:

   # **openvpn --genkey secret ta.key**

   Y por último copiamos los ficheros generados al directorio anterior /etc/openvpn:

   # **cp ta.key /etc/openvpn/**

   # **cp pki/ca.crt /etc/openvpn/server/**

   # **cp pki/private/servidor.key /etc/openvpn/server/**

   # **cp pki/issued/servidor.crt /etc/openvpn/server/**

   # **cp pki/dh.pem /etc/openvpn/server/**

4. Generamos la clave y el certificado para el cliente:

   # **./easyrsa gen-req *cliente* nopass**

   Firmamos el certificado del cliente:

   # **./easyrsa sign-req client *cliente***

   Y por último copiamos los ficheros generados al directorio /etc/openvpn/client:

   # **cp pki/ca.crt /etc/openvpn/client/**

cp pki/issued/cliente.crt /etc/openvpn/client/

cp pki/private/cliente.key /etc/openvpn/client/

5. Configuramos el servidor OpenVPN:

   # **cp /usr/share/doc/openvpn/examples/sample-config-files/server.conf /etc/openvpn/server.conf**

   # **nano /etc/openvpn/server.conf**

```
port 1194
proto udp
dev tun

ca   /etc/openvpn/server/ca.crt
cert /etc/openvpn/server/servidor.crt
key  /etc/openvpn/server/servidor.key
dh   /etc/openvpn/server/dh.pem

server 10.8.0.0 255.255.255.0
keepalive 10 120

tls-auth /etc/openvpn/ta.key 0
cipher AES-256-CBC

user nobody
group nogroup

persist-key
persist-tun

status      /var/log/openvpn/openvpn-status.log
log         /var/log/openvpn/openvpn.log
log-append  /var/log/openvpn/openvpn.log
verb 3
explicit-exit-notify 1
```

   # **systemctl start openvpn@server**

   # **systemctl status openvpn@server**

   # **systemctl enable openvpn@server**

   Ahora deberíamos ver la nueva interfaz *tun0* para la conexión VPN:

   # **ip a show tun0**

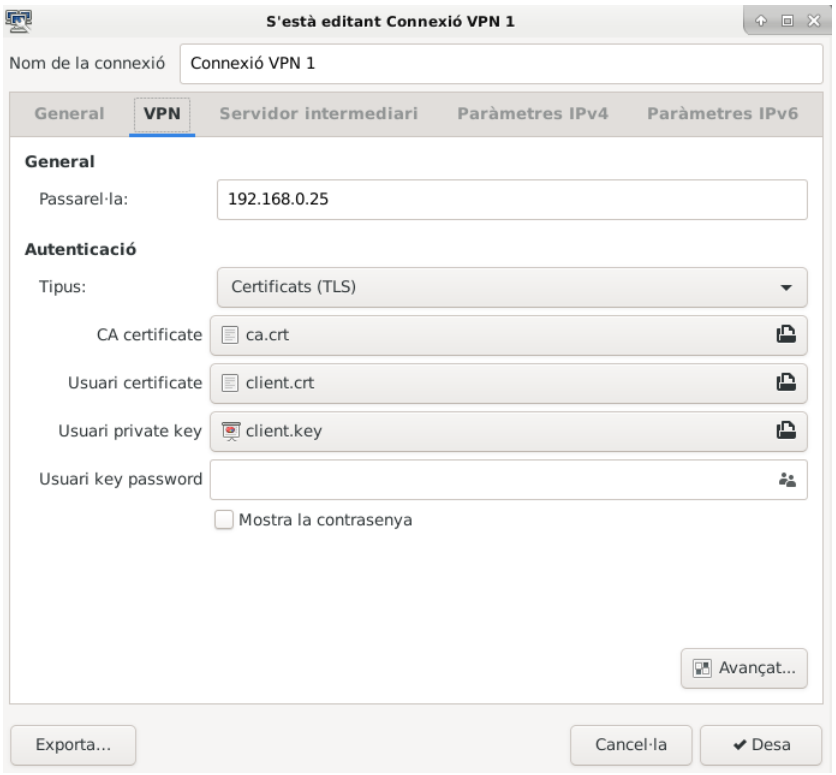
   Pero si algo ha fallado, revisa los logs en:

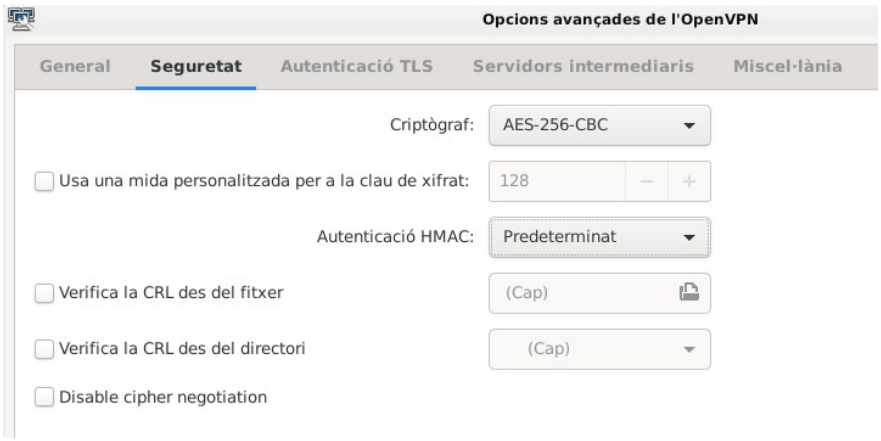
   # **cat /var/log/openvpn/openvpn.log**

6. Ahora que ya tenemos el servidor OpenVPN listo, vamos a por el cliente que se conectará. Debemos mover a dicho cliente los fichero ca.crt , cliente.crt y cliente.key que se encuentran en la carpeta /etc/openvpn/client/ del servidor, y también el fichero ta.key que se encuentran en la carpeta /etc/openvpn/ del servidor. Para ello puedes utilizar un USB, carpetas compartidas, scp, sftp, etc.

   Una vez ya tengas dichos ficheros en el cliente, configuramos la conexión del cliente, en nuestro caso des del Network Manager en el escritorio, que nos pedirá el certificado y clave del cliente:

```
cliente # apt install network-manager-openvpn-gnome
```





Iniciamos la conexión VPN Comprobamos que funciona accediendo a servicios de la red. En mi caso detrás del servidor VPN he puesto otro servidor en red local con él, con servicio SSH e IP 192.168.112.131:

```
cliente # ip a show tun0
cliente # ssh root@192.168.112.131
```

Datos de la práctica "Site-to-Site VPN"

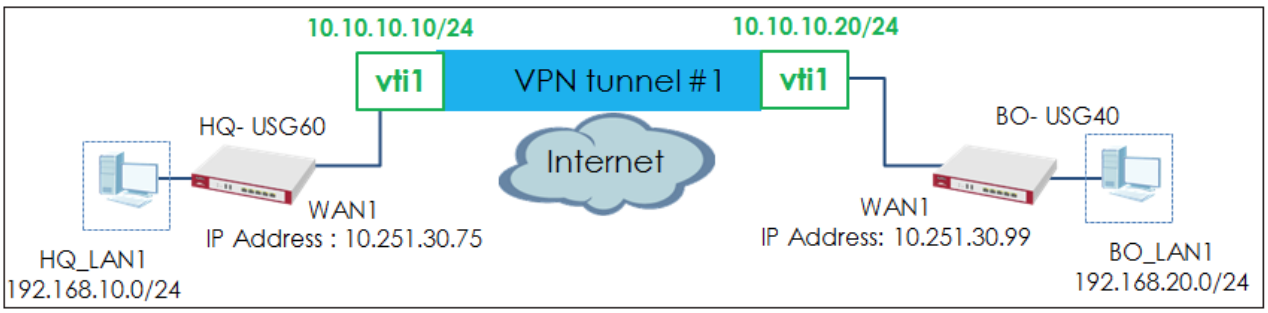
Crea el siguiente montaje de red para unir mediante VPN "site to site" dos o más redes locales de una empresa separadas físicamente.

Cada red local estará simulada en un anfitrión diferente. Dicho anfitrión contendrá:

- Una máquina virtual que será un endpoint VPN, con dos tarjetas de red:
  - Una tarjeta de red para la red de la empresa (LAN) en red interna. Escoge la IPs que quieras.
  - Una tarjeta de red para Internet (WAN) en modo puente.
- Una máquina virtual con servicio ssh y web que estará en la red de la empresa. Escoge la IP.

La tecnología del tunel del endpoint VPN será WireGuard sobre Linux. Aviso: recomiendo núcleo 5.6 o superior.

Para hacerte una idea del montaje, mira las imágenes (tus rangos de IP pueden variar):



Realiza la práctica por parejas. Cada miembro de la pareja monta la red local en su anfitrión y luego unís las dos redes por VPN.

Otra opción es que cada grupo de clase monta una red local de una sede de la empresa, y luego unimos todas las sedes mediante VPN.

Práctica: "Site-to-Site VPN" con Linux/BSD (por probar)

Con WireGuard y pfSense:

- https://www.youtube.com/watch?v=ZY49EAMnniY

- https://www.youtube.com/watch?v=YfP0Kx4tdBI

Con WireGuard y OPNsense :

- https://www.youtube.com/watch?v=RoXHe5dqCM0

A pesar de lo fácil e intuitivo que es montar una VPN con una distribución especializada para encaminadores y que se administra vía navegador web, como OPNsense o pfSense, montaremos una VPN “a pelo” con Wireguard sobre Debian 11, y nos conectaremos des de otra red también con Wireguard sobre Debian 11. Utilizaré como referencia la información que se encuentra en:

- https://wiki.archlinux.org/index.php/WireGuard

1. Activamos el enrutamiento:

   # **nano /etc/sysctl.conf**

   ```
   net.ipv4.ip_forward=1
   ```

   # **sysctl -p**

   Y a continuación instalamos el programa:

   # **apt update**

   # **apt install wireguard**

2. Generamos las claves pública y privada para el servidor:

   # **cd /etc/wireguard/**

   # umask 077

   # wg genkey | tee privatekey | wg pubkey > publickey

   # **ls -l privatekey publickey**

   Necesitaras recordar las dos claves, para escribirlas en los ficheros de configuración:

   # **cat privatekey**

   # **cat publickey**

3. Configuramos el servidor WireGuard:

   # **nano /etc/wireguard/wg0.conf**

   ```
   [Interface]

   ## IP del servidor dentro del tunel VPN ##
   Address = 10.10.10.10/32

   ## Puerto del servidor para el tunel VPN con WireGuard ##
   ListenPort = 9999

   ## Clave secreta del servidor VPN ##
   PrivateKey = copia/pega la clave de /etc/wireguard/privatekey

   [Peer]
   ```

```
## Clave pública del servidor VPN del otro sitio ##
PublicKey = copia/pega la clave de /etc/wireguard/publickey del otro

## ACL ##
AllowedIPs = 10.10.10.20/32

## IP pública y puerto de Wireguard del otro servidor ##
Endpoint = 10.251.30.99:9999

## Mantiene la conexión viva si estás detrás de NAT ##
PersistentKeepalive = 15
```

```
# systemctl start wg-quick@wg0
# systemctl status wg-quick@wg0
# systemctl enable wg-quick@wg0
```

Ahora deberíamos ver la nueva interfaz *wg0* para la conexión VPN:

```
# ip a show wg0
# wg
```

4. La configuración del otro sitio será muy parecida. Se realizan los mismos pasos, pero las claves públicas y privadas, y las IPs cambiarán. Todo debería funcionar y podríamos hacer ping des de las máquinas que están dentro de una red hasta las máquinas que están dentro de la otra red.

Referencias

- https://en.wikipedia.org/wiki/Virtual_private_network
- https://www.youtube.com/watch?v=_wQTRMBAvzg
- https://www.cisco.com/c/en/us/support/docs/security-vpn/ipsec-negotiation-ike-protocols/14106-how-vpn-works.html
- https://en.wikipedia.org/wiki/Tunneling_protocol#Secure_Shell_tunneling
- https://en.wikipedia.org/wiki/OpenVPN
- https://en.wikipedia.org/wiki/IPsec
- https://en.wikipedia.org/wiki/WireGuard
- https://www.ivpn.net/pptp-vs-ipsec-ikev2-vs-openvpn-vs-wireguard/