## **Manual: Instalación y Configuración de una VPN con OpenVPN**

### Introducción: ¿Qué es una VPN y por qué usarla?

#### ¿Qué es una VPN?

Una VPN (Virtual Private Network, o Red Privada Virtual) es una tecnología que nos permite crear una conexión segura y cifrada entre un dispositivo (como un ordenador, móvil o tablet) y una red privada (como la red de nuestro servidor Ubuntu en Proxmox). Esta conexión actúa como un "túnel" seguro a través de Internet, protegiendo los datos que enviamos y recibimos.

#### Ventajas de usar una VPN

1. Seguridad:
   * Cifrado de datos: Toda la información que viaja a través de la VPN está cifrada, lo que la protege de posibles interceptaciones.
   * Protección contra ataques: Al acceder a la red interna a través de una VPN, reducimos el riesgo de ataques externos, ya que la conexión es privada y segura.
2. Acceso remoto:
   * Conectividad desde cualquier lugar: Nos permite acceder a los recursos de la red interna (como servidores, aplicaciones o archivos) desde fuera de la red local, como si estuviéramos físicamente en el lugar.
   * Flexibilidad: Es ideal para trabajar de manera remota o para dar acceso a usuarios externos de forma controlada.
3. Privacidad:
   * Ocultación de la IP: La VPN oculta nuestra dirección IP real, lo que aumenta la privacidad y el anonimato en Internet.
   * Evita el rastreo: El cifrado de la VPN dificulta que terceros rastreen nuestra actividad en línea.
4. Control de acceso:
   * Acceso restringido: Solo los usuarios autorizados (con los certificados y credenciales adecuados) pueden conectarse a la VPN.
   * Segmentación de redes: Podemos limitar el acceso a ciertos recursos dentro de la red, lo que mejora la seguridad.

#### ¿Por qué es una buena idea implementar una VPN en este proyecto?

En nuestro proyecto, queremos dar a conocer el ciclo de Administración de Sistemas Informáticos en Red (ASIR) y mostrar cómo se pueden implementar soluciones prácticas y seguras en un entorno real. Implementar una VPN nos permite:

* Demostrar habilidades técnicas: Configurar una VPN es una tarea avanzada que muestra nuestro dominio de redes, seguridad y sistemas.
* Proporcionar acceso seguro: Si queremos que los visitantes de las jornadas de puertas abiertas accedan a la página web o a otros recursos de forma segura, la VPN es la solución ideal.
* Fomentar la seguridad: Al usar una VPN, promovemos la importancia de la seguridad y la privacidad en el mundo digital, algo clave en la formación de un administrador de sistemas.

### Conceptos clave: Certificados, PKI, CA, Diffie-Hellman, Tunneling, Reglas de IPTables y Autenticación y Autorización

#### 1. Certificados Digitales

* Definición: Archivos digitales que utilizamos para verificar la identidad de un dispositivo, servidor o usuario en una red. Funcionan como una especie de "credencial digital" que garantiza que una entidad es quien dice ser.
* Importancia: Los certificados son esenciales para garantizar la autenticidad y seguridad de las conexiones VPN. Sin ellos, no podríamos confiar en que las conexiones sean seguras.

#### 2. PKI (Infraestructura de Clave Pública)

* Definición: Un sistema que gestiona la creación, distribución y revocación de certificados digitales.
* Importancia: La PKI es la base sobre la cual se construye la seguridad de la VPN. Nos permite generar y gestionar los certificados necesarios para autenticar a los dispositivos y cifrar las comunicaciones.

#### 3. CA (Autoridad Certificadora)

* Definición: Una entidad de confianza que emite y gestiona certificados digitales.
* Importancia: La CA es quien garantiza que los certificados son válidos y pertenecen a una entidad específica. En nuestro caso, creamos nuestra propia CA para emitir certificados para el servidor y los clientes.

#### 4. Diffie-Hellman

* Definición: Un protocolo criptográfico que permite a dos partes generar una clave secreta compartida sobre un canal no seguro.
* Importancia: Este protocolo es crucial para el intercambio seguro de claves en la VPN. Garantiza que, incluso si alguien intercepta el tráfico, no podrá descifrarlo sin la clave compartida.

#### 5. Tunneling

* Definición: La creación de un "túnel" seguro a través del cual viajan los datos entre el cliente y el servidor VPN.
* Importancia: El tunneling es lo que permite que los datos viajen de manera segura a través de Internet, protegiéndonos de posibles interceptaciones. En una VPN, los datos se encapsulan dentro de paquetes cifrados que viajan a través de este túnel.

#### 6. Reglas de IPTables

* Definición: IPTables es una herramienta en Linux que nos permite configurar reglas de firewall para controlar el tráfico de red.
* Importancia: En nuestro caso, utilizamos reglas de IPTables para:
  + Permitir el tráfico de la VPN: Aseguramos que el tráfico entre el cliente y el servidor VPN no sea bloqueado.
  + Habilitar el reenvío de IP: Permitimos que el tráfico de la VPN fluya entre el cliente y la red interna.
  + Proteger el servidor: Bloqueamos accesos no autorizados al servidor VPN.

#### 7. Autenticación y Autorización

* Definición:
  + Autenticación: Verificar la identidad de un usuario o dispositivo (por ejemplo, mediante certificados digitales).
  + Autorización: Determinar a qué recursos puede acceder un usuario o dispositivo autenticado.
* Importancia: En la VPN, la autenticación se realiza mediante certificados digitales, lo que garantiza que sólo los dispositivos autorizados puedan conectarse. La autorización se gestiona mediante reglas de red y permisos, asegurando que los usuarios solo accedan a los recursos permitidos.

### Paso 1: Preparación del servidor Ubuntu

1. Accedemos al servidor Ubuntu:
   * Nos conectamos al servidor Ubuntu (10.10.16.2) mediante SSH o directamente desde la consola de Proxmox.
2. Actualizamos el sistema:
   * Ejecutamos los siguientes comandos para asegurarnos de que el sistema esté actualizado:

sudo apt update

sudo apt upgrade -y

### Paso 2: Instalación de OpenVPN

1. Instalamos OpenVPN y Easy-RSA:
   * Easy-RSA es una herramienta que nos ayudará a gestionar los certificados SSL necesarios para la VPN.
   * Ejecutamos el siguiente comando:

sudo apt install openvpn easy-rsa -y

* + Explicación del comando:
    - **sudo apt install openvpn easy-rsa -y:** Instala los paquetes openvpn (para la VPN) y easy-rsa (para gestionar certificados).

1. Configuramos Easy-RSA:
   * Copiamos la plantilla de Easy-RSA a un directorio de trabajo:

mkdir ~/easy-rsa

cp -r /usr/share/easy-rsa/\* ~/easy-rsa/

cd ~/easy-rsa

* + Editamos el archivo de configuración de variables:

nano vars

* + Añadimos o modificamos las siguientes líneas con nuestros datos:

export KEY\_COUNTRY="ES"

export KEY\_PROVINCE="Gipuzkoa"

export KEY\_CITY="Irun"

export KEY\_ORG="Plaiaundi"

export KEY\_EMAIL="ikdxz@plaiaundi.net"

export KEY\_OU="ASIR"

export KEY\_NAME="servidor"

1. Generamos los certificados y claves:
   * Inicializamos el PKI (Public Key Infrastructure):

./easyrsa init-pki

* + Explicación del comando:
    - **./easyrsa init-pki:** Inicializa la infraestructura de clave pública (PKI) en el directorio actual.
  + Generamos la CA (Autoridad Certificadora):

./easyrsa build-ca

* + Explicación del comando:
    - .**/easyrsa build-ca:** Crea la Autoridad Certificadora (CA) y genera los certificados raíz. Nos pedirá una contraseña para proteger la CA.
  + Generamos el certificado y la clave para el servidor:

./easyrsa gen-req 10.10.16.2 nopass

./easyrsa sign-req server 10.10.16.2

* + Explicación de los comandos:
    - **./easyrsa gen-req server nopass:** Genera una solicitud de certificado para el servidor sin contraseña (nopass).
    - **./easyrsa sign-req server server:** Firma la solicitud de certificado para el servidor.
  + Generamos el certificado Diffie-Hellman (necesario para el intercambio de claves):

./easyrsa gen-dh

* + Explicación del comando:
    - **./easyrsa gen-dh:** Genera el archivo de parámetros Diffie-Hellman, que se utiliza para el intercambio seguro de claves.
  + Generamos la clave HMAC (para mejorar la seguridad):

openvpn --genkey secret ta.key

* + Explicación del comando:
    - **openvpn --genkey ta.key:** Genera una clave HMAC (Hash-based Message Authentication Code) para proteger contra ataques de repetición.

1. Movemos los archivos generados:
   * Copiamos los archivos generados al directorio de configuración de OpenVPN:

sudo cp pki/ca.crt pki/issued/10.10.16.2.crt pki/private/10.10.16.2.key pki/dh.pem ta.key /etc/openvpn/server/

### Paso 3: Configuración del servidor OpenVPN

1. Creamos el archivo de configuración del servidor:
   * Copiamos la plantilla de configuración:

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

* + Explicación de los comandos:
    - **sudo cp ... /etc/openvpn/server/:** Copia la plantilla de configuración del servidor.
  + Editamos el archivo de configuración:

sudo nano /etc/openvpn/server/server.conf

* + Aseguramos que las siguientes líneas estén configuradas correctamente:

port 1194

proto udp

dev tun

ca /etc/openvpn/server/ca.crt

cert /etc/openvpn/server/10.10.16.2.crt

key /etc/openvpn/server/10.10.16.2.key

dh /etc/openvpn/server/dh.pem

server 10.8.0.0 255.255.255.0

push "redirect-gateway def1 bypass-dhcp"

push "dhcp-option DNS 8.8.8.8"

keepalive 10 120

tls-auth /etc/openvpn/server/ta.key 0

cipher AES-256-GCM

auth SHA256

data-ciphers AES-256-GCM:AES-128-GCM

auth SHA256

data-ciphers AES-256-GCM:AES-128-GCM

user nobody

group nogroup

persist-key

persist-tun

status openvpn-status.log

verb 3

| **Resumen de la configuración** | |
| --- | --- |
| **Parámetro** | **Descripción** |
| *port 1194* | Puerto en el que el servidor escucha conexiones. |
| *proto udp* | Protocolo de transporte (UDP o TCP). |
| *dev tun* | Tipo de dispositivo de red virtual (tun para IP, tap para Ethernet). |
| *ca ca.crt* | Certificado de la Autoridad Certificadora (CA). |
| *cert server.crt* | Certificado del servidor. |
| *key server.key* | Clave privada del servidor. |
| *dh dh.pem* | Parámetros Diffie-Hellman para el intercambio de claves. |
| *server 10.8.0.0 255.255.255.0* | Subred para asignar direcciones IP a los clientes. |
| *push "redirect-gateway def1"* | Redirige todo el tráfico del cliente a través de la VPN. |
| *push "dhcp-option DNS 8.8.8.8"* | Mecanismo para mantener la conexión activa. |
| *keepalive 10 120* | Mecanismo para mantener la conexión activa. |
| *tls-auth ta.key 0* | Autenticación TLS adicional para mayor seguridad. |
| *cipher AES-256-CBC* | Algoritmo de cifrado utilizado. |
| *user nobody* | Grupo bajo el cual se ejecuta OpenVPN. |
| *group nogroup* | Grupo bajo el cual se ejecuta OpenVPN. |
| *persist-key* | Evita la relectura de la clave privada en reinicios. |
| *persist-key* | Evita el cierre y reapertura de la interfaz TUN/TAP en reinicios. |
| *status openvpn-status.log* | Nivel de detalle de los registros (logs). |
| *verb 3* | Nivel de detalle de los registros (logs). |

1. Habilitamos el reenvío de IP:
   * Editamos el archivo de configuración de red:

sudo nano /etc/sysctl.conf

* + Aseguramos que la siguiente línea esté descomentada:

net.ipv4.ip\_forward=1

* + Aplicamos los cambios:

sudo sysctl -p

* + Explicación de los comandos:
    - **sudo sysctl -p:** Aplica los cambios en la configuración del kernel.

1. Configuramos las reglas de iptables:
   * Añadimos las reglas necesarias para permitir el tráfico a través de la VPN:

sudo iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o ens18 -j MASQUERADE

* + Guardamos las reglas para que persistan después de un reinicio:

sudo apt install iptables-persistent

sudo netfilter-persistent save

* + Explicación de los comandos:
    - **sudo iptables -t nat -A POSTROUTING ...:** Añade una regla de NAT para redirigir el tráfico de la VPN.
    - **sudo apt install iptables-persistent:** Instala el paquete para guardar las reglas de iptables.
    - **sudo netfilter-persistent save:** Guarda las reglas actuales de iptables.

1. Habilitamos y reiniciamos OpenVPN:
   * Habilitamos el servicio para que se inicie automáticamente:

sudo systemctl enable openvpn@server

* + Reiniciamos el servicio:

sudo systemctl restart openvpn@server

* + Explicación de los comandos:
    - **sudo systemctl enable openvpn@servidor:** Habilita el servicio de OpenVPN para que se inicie automáticamente.
    - **sudo systemctl restart openvpn@servidor:** Reinicia el servicio de OpenVPN.

### Paso 4: Configuración de los clientes

1. Generamos los certificados para los clientes:
   * Volvemos al directorio de Easy-RSA:

cd ~/easy-rsa

* + Generamos un certificado para el cliente (por ejemplo, cliente1):

./easyrsa gen-req cliente1 nopass

./easyrsa sign-req client cliente1

* + Explicación de los comandos:
    - **./easyrsa gen-req asir1 nopass:** Genera una solicitud de certificado para el cliente sin contraseña.
    - .**/easyrsa sign-req client asir1:** Firma la solicitud de certificado para el cliente.

1. Creamos el archivo de configuración del cliente:
   * Copiamos la plantilla de configuración:

cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf ~/cliente1.ovpn

* + Editamos el archivo:

nano ~/cliente1.ovpn

* + Configuramos las siguientes líneas:

client

dev tun

proto udp

remote 10.10.13.202 1194

resolv-retry infinite

nobind

persist-key

persist-tun

ca ca.crt

cert cliente1.crt

key cliente1.key

tls-auth ta.key 1

cipher AES-256-GCM

auth SHA256

verb 3

1. Transferimos el archivo de configuración al cliente:
   * Copiamos el archivo cliente1.ovpn y los archivos de certificados (ca.crt, cliente1.crt, cliente1.key, ta.key) al dispositivo del cliente.

### Paso 5: Conexión desde el cliente

1. Instalamos OpenVPN en el cliente:
   * En el dispositivo del cliente, instalamos OpenVPN.
2. Importamos la configuración:
   * Importamos el archivo cliente1.ovpn en el cliente OpenVPN.

1. Conectamos a la VPN:
   * Iniciamos la conexión VPN desde el cliente. Si todo está bien configurado, el cliente debería conectarse al servidor y tener acceso a la red interna.

### Paso 6: Verificación

1. Comprobamos la conexión:
   * Desde el cliente, intentamos acceder a recursos internos (por ejemplo, la página web en 10.10.16.2).
   * Verificamos que el tráfico esté pasando a través de la VPN.







