PRÁCTICA N° 2

CONFIGURACIÓN DE UNA VPN CON OPENVPN.

# OBJETIVOS

* 1. Configurar un servidor VPN utilizando OpenVPN en Ubuntu.
  2. Garantizar comunicaciones seguras mediante el uso de certificados y claves.
  3. Implementar un cliente VPN y verificar la conexión al servidor.
  4. Proteger el tráfico con un firewall y reenvío de paquetes.

# MARCO TEÓRICO

## OpenVPN

OpenVPN es una herramienta de código abierto diseñada para crear redes privadas virtuales (VPN), ofreciendo conexiones seguras y encriptadas a través de redes públicas o privadas. Su portabilidad destaca como una de sus principales ventajas, permitiendo una fácil integración en distintos sistemas y dispositivos.

Este software es ampliamente reconocido por establecer redes confiables en entornos como Internet y redes corporativas, que suelen presentar riesgos para la seguridad. OpenVPN se basa en la biblioteca OpenSSL para proporcionar cifrado sólido y mecanismos de autenticación, asegurando tanto la confidencialidad como la integridad de los datos transmitidos. Asimismo, OpenVPN es compatible con una variedad de plataformas, como Windows, macOS, Linux, iOS y Android, lo que facilita su uso en numerosos dispositivos y sistemas operativos. Su combinación de flexibilidad y robustez lo posiciona como una solución preferida para crear túneles seguros y proteger comunicaciones en entornos poco confiables.

Por otro lado, una autoridad de certificación (CA) es una entidad encargada de verificar la identidad digital de sitios web, direcciones de correo electrónico, empresas o individuos. Esto se logra mediante el uso de certificados digitales, que actúan como herramientas criptográficas para confirmar la autenticidad. En consecuencia, se puede usar para una VPN.

Además, las CA se clasifican de la siguiente manera:

* CA pública: Estas entidades ofrecen servicios a nivel global, emitiendo certificados que son reconocidos y aceptados internacionalmente. Son ideales para proteger sitios web, facilitar transacciones en línea y otros casos de uso digital. Aunque el número de CA públicas es limitado, tienen una sólida relación de confianza con los principales navegadores web, lo que garantiza su aceptación generalizada.
* CA privada: Estas autoridades están destinadas al uso exclusivo dentro de una organización. Emiten certificados para propósitos internos, como redes privadas, VPN, autenticación de usuarios o firma de código. A diferencia de las CA públicas, una CA privada es "de confianza" únicamente dentro de la empresa que la utiliza y rara vez extiende certificados a entidades externas. Por esta razón, también se las conoce como “CA locales”.

Por otro lado, es importante señalar que las autoridades de certificación (CA) son esenciales en la Infraestructura de Clave Pública (PKI), ya que garantizan la autenticidad de las identidades digitales y verifican que las entidades en línea sean legítimas. Al emitir un certificado digital, la CA asocia una clave pública con la identidad confirmada de su propietario, promoviendo la confianza en las transacciones electrónicas y las comunicaciones en línea.

Por consiguiente una PKI o Infraestructura de Clave Pública es un sistema compuesto por procesos, tecnologías y normativas que habilitan el cifrado y la firma digital de datos. Este sistema permite la generación de certificados digitales utilizados para verificar la identidad de usuarios, dispositivos o servicios. Gracias a estos certificados, se pueden establecer conexiones seguras tanto en sitios web públicos como en entornos privados, como redes VPN, redes Wi-Fi internas y otros servicios que demandan autenticación multifactor . Por lo tanto, resulta crucial para asegurar la confidencialidad, integridad y autenticación en las interacciones digitales, ofreciendo un fundamento firme para la protección en entornos vinculados a Internet.

# TRABAJO PREPARATORIO

* 1. Consultar que es una VPN, su funcionalidad y los protocolos VPN más comunes.
  2. Escribir un breve resumen sobre OpenVPN y las ventajas que posee ante otros protocolos.
  3. Revisar y comprender acerca de la configuración de redes en Ubuntu.
  4. Consultar el uso de ufw para gestionar reglas de firewall.

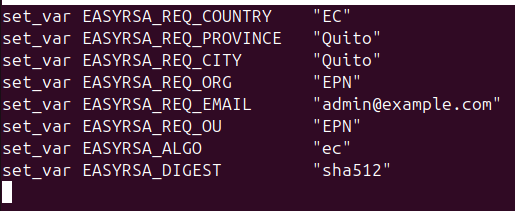
# EQUIPO Y MATERIALES

* Oracle VM Virtual Box o VMWare Player.
* Dos máquinas virtuales con distribución Ubuntu/Debian que actúen como servidor OpenVPN y servidor CA.
* Dispositivo móvil que actúe como cliente (puede ser Windows, Linux, macOS o Android) para probar la conexión VPN.

# PROCEDIMIENTO

**Instalación del servidor CA**

* Primero se debe instalar EasyRSA, que servirá para configurar el servidor CA con el siguiente comando: sudo apt install easy-rsa.
* Crear un directorio para configurar el PKI, con el comando: mkdir ~/easy-rsa
* Generar enlaces simbólicos que direccionen a los archivos del paquete easy-rsa instalado, con lo siguiente: ln -s /usr/share/easy-rsa/\* ~/easy-rsa/.
* Cambiar los permisos del directorio PKI e iniciarlo con los siguientes comandos respectivamente: chmod 700 /home/<username>/easy-rsa, ./easyrsa init-pki. Para inicializar es necesario cambiarse al directorio ~/easy-rsa con cd.
* Configurar la CA abriendo el archivo vars para ingresar la información correspondiente, para esto se usa el comando nano ~/easy-rsa/vars.



* Ejecutar ./easyrsa build-ca para crear las claves raíz privada y pública.

A computer screen shot of a computer program

Description automatically generated

Se debe ingresar una contraseña para el par de claves y un nombre común Además, para que no se solicite la contraseña cada que se use la CA se utiliza lo siguiente: ./easyrsa build-ca nopass.

* Luego crear dos archivos ca.crt, el cual es el archivo de certificado público de la CA y el ca.key, es la clave privada utilizada por la CA para firmar certificados para el servidor y cliente OpenVPN.

A screenshot of a computer program

Description automatically generated

**Instalación de OpenVPN y EasyRSA en el servidor OpenVPN**

* Se instalan los paquetes OpenVPN y Easy-RSA con el siguiente comando: sudo apt install openvpn easy-rsa.

**A screen shot of a computer screen

Description automatically generated**

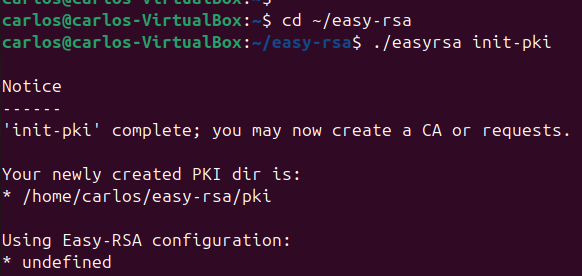
* Crear el directorio ~/easy-rsa para configurar el PKI en el servidor OpenVPN tal y como se hizo anteriormente cambiando los permisos y creando en enlace simbólico.

A computer screen with text

Description automatically generated

.

* Configurar un PKI, que será usado para almacenar certificados y solicitudes de certificados, para el servidor OpenVPN tal como se hizo para el anterior servidor. La configuración se muestra en a continuación



* Editar el archivo vars contenido en el directorio pki y realizar la siguiente configuración en dicho archivo.

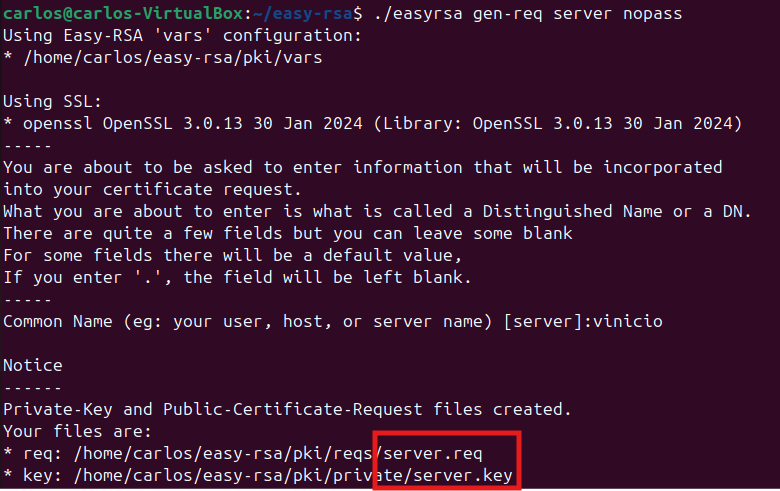
A black background with white text

Description automatically generated

.

**Creación de la solicitud de certificado y la clave privada del servidor OpenVPN**

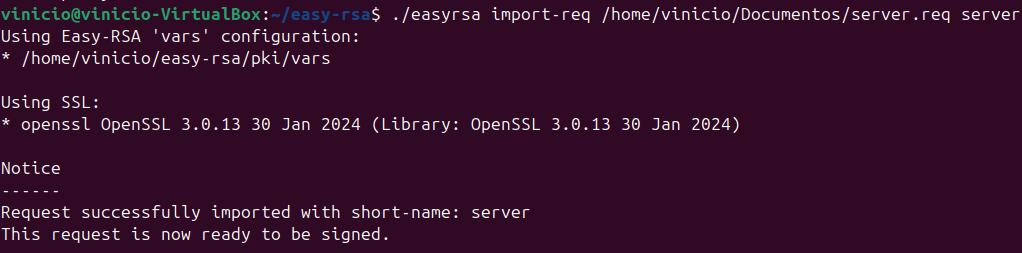
* Se debe generar una clave privada y una solicitud de firma de certificado (CSR) con el siguiente comando: ./easyrsa gen-req cn nopass, donde cn=server para el caso.



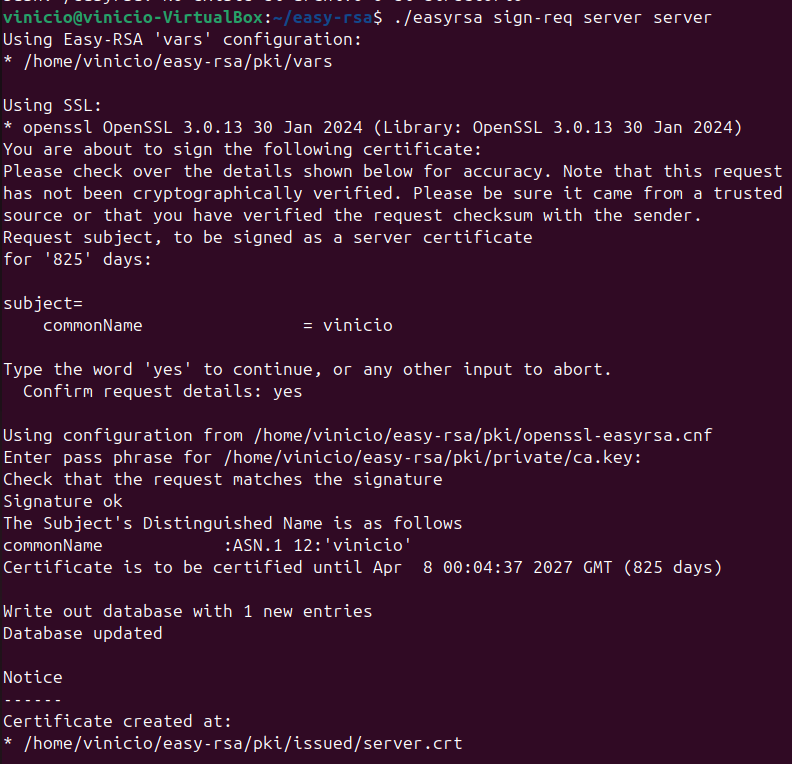
En consecuencia, se crean los archivos server.req, server.key. Además, se debe copiar la clave del servidor (server.key) en el directorio /etc/openvpn/server, para que la CSR esté lista para ser firmada.

**Firmar la CSR del servidor OpenVPN**

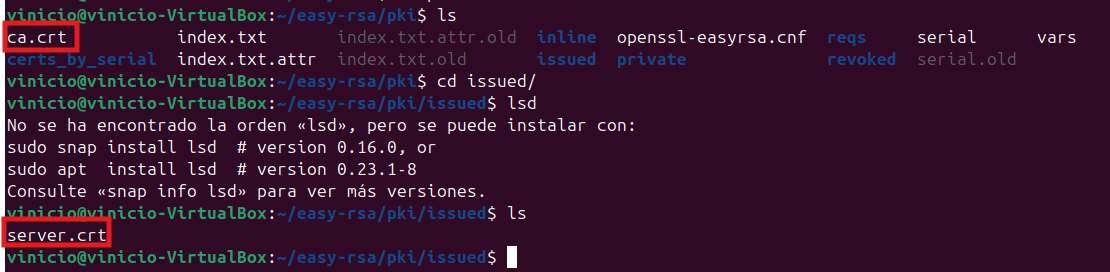
* Es necesario copiar el archivo server.req en un directorio cualquiera del servidor CA para poder importarlo y firmarlo. Se importa con el comando ./easyrsa import-req /<directorio>/server.req server.

****

* Para firmar el archivo se usa el siguiente comando ./easyrsa sign-req server server. En este caso se pide la contraseña ingresada cuando se generaron el par de claves,



* Se obtienen los archivos ca.cart, server.crt, los cuales contienen la firma del servidor CA y la clave de cifrado pública para el servidor OpenVPN.



Los archivos resultantes se deben copiar hacia el servidor OpenVPN en el directorio /etc/openvpn/server.

**Configurar el material criptográfico OpenVPN**

* Generar una clave Diffie-Hellman para el intercambio de claves con el comando ./easyrsa gen-dh en el directorio ~/easy-rsa y una clave precompartida con el comando sudo openvpn --genkey secret ta.key. Los archivos generados se copian al directorio /etc/openvpn/server.

**A screenshot of a computer program

Description automatically generated**

**Generar un certificado de cliente y un par de claves**

* Crear un directorio para almacenar el certificado del cliente y los archivos de claves, además se deben restringir los accesos. Para el caso ~/client-configs/keys.

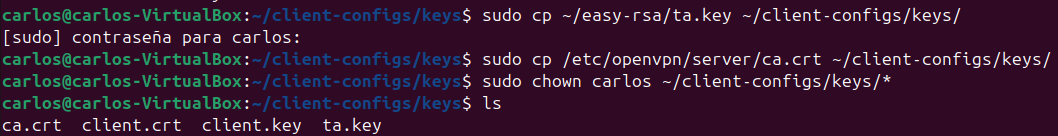
****

* Generar una clave para el cliente donde se puede usar cualquier CN, para esto se usa el comando usado para generar las claves para el servidor. Después se copia el archivo client.key al directorio ~/client-configs/keys.

A screenshot of a computer program

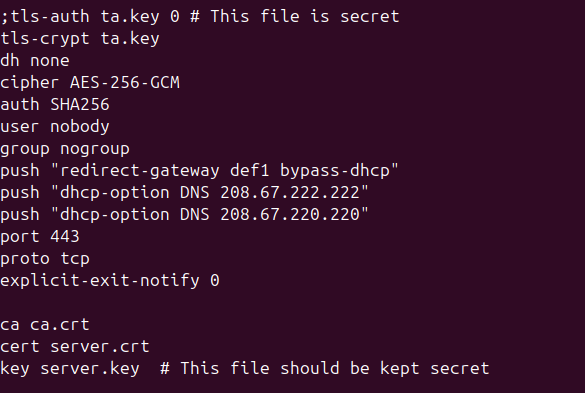
Description automatically generated

Luego el archivo client.req debe ser firmado de igual forma que se hizo con el archivo server.req, de lo que se obtiene el archivo client.crt que debe ser ubicado en el directorio ~/client-configs/keys. Además, los archivos ca.crt, ta.key también deben ser copiados a dicho directorio, así como establecer al usuario *carlos* como propietario el directorio con el comando chown.



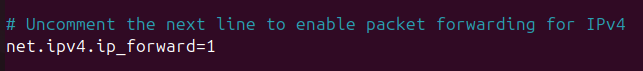
**Configurar OpenVPN**

* Para configurar OpenVPN primero se debe copiar el archivo server.conf ubicado en el path /usr/share/doc/openvpn/examples/sample-config-files/ hacia el directorio /etc/openvpn/server/. Y editarlo de la siguiente forma como se muestra en la figura 2.32.

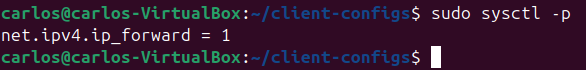


**Configuración de red**

* Para configurar la red del servidor y que enrute correctamente se debe configurar el reenvió de puertos en el archivo sysctl.conf ubicado en el directorio /etc, de la siguiente manera:

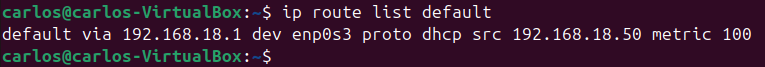


Además, se debe cargar y leer el archivo para la sesión con el comando sudo sysctl -p.



**Configurar el firewall**

* Se debe habilitar el enmascaramiento para que el tráfico de la VPN atraviese el firewall. Para esto primero se debe conocer la interfaz de red pública del servidor con el siguiente comando: ip route list default



* Luego se edita el archivo before.rules ubicado en el path /etc/ufw/. Se agrega las siguientes líneas:

# START OPENVPN RULES

# NAT table rules

\*nat

:POSTROUTING ACCEPT [0:0]

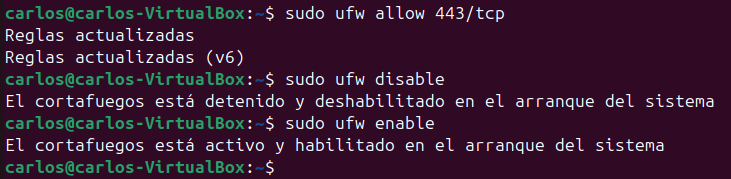
# Allow traffic from OpenVPN client to ens3 (change to the interface you discovered!)

-A POSTROUTING -s 10.8.0.0/8 -o ens3 -j MASQUERADE

COMMIT

# END OPENVPN RULES

* Después se configura el archivo ufw ubicado en el directorio etc/default/ para permitir paquetes reenviados de forma predeterminada. Para ello se debe establecer la línea DEFAULT\_FORWARD\_POLICY en ACCEPT.
* Por último, se habilita el puerto 443 y luego deshabilitar y habilitar el firewall para que se aplique la nueva configuración con los comandos: sudo ufw disable, sudo ufw enable respectivamente.



**Iniciar el servicio Open VPN**

* Se inicia el servicio con el comando sudo systemctl start openvpn-server@server.service, y se consulta el estado con el comando sudo systemctl status openvpn-server@server.service. Despues se establece para que el servicio arranque a penas se encienda el dispositivo con el comando sudo systemctl -f enable openvpn-server@server.service.

Si existiera algún cambio en la configuración se debe reiniciar el servicio con el comando sudo systemctl restart openvpn-server@server.service.

**Crear la configuración del cliente**

* Para probar la VPN se debe crear un archivo ejecutable para que el cliente pueda conectarse a la VPN. Para ello se crea un directorio para guardar los archivos de configuración del cliente (mkdir -p ~/client-configs/files), después se copia el archivo de configuración de ejemplo (client.config) al directorio ~/client-configs con el comando cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf ~/client-configs/base.conf. Por último, se edita el archivo base.conf de la siguiente forma.

A screenshot of a computer

Description automatically generated

* Después se crea un script para compilar la configuración base con los archivos de certificado, clave y cifrado en el directorio ~/client-configs/, con el comando nano ~/client-configs/make\_config.sh y se cambia los permisos para que sea ejecutable (chmod 700). El código para el script se muestra a continuación.

#!/bin/bash

# First argument: Client identifier

KEY\_DIR=/home/<username>/client-configs/keys

OUTPUT\_DIR=/home/<username>/client-configs/files

BASE\_CONFIG=/home/<username>/client-configs/base.conf

cat ${BASE\_CONFIG} \

<(echo -e '<ca>') \

${KEY\_DIR}/ca.crt \

<(echo -e '</ca>\n<cert>') \

${KEY\_DIR}/${1}.crt \

<(echo -e '</cert>\n<key>') \

${KEY\_DIR}/${1}.key \

<(echo -e '</key>\n<tls-crypt>') \

${KEY\_DIR}/ta.key \

<(echo -e '</tls-crypt>') \

> ${OUTPUT\_DIR}/${1}.ovpn

* Por último, se crea el archivo de configuración del cliente (cliente.ovpn) que posee certificados y archivos de claves, con el comando ./make\_config.sh client

A screen shot of a computer code

Description automatically generated

# INFORME

* 1. Documentar las configuraciones aplicadas.
  2. Presentar las capturas de pantalla, con la debida explicación de los resultados mostrados. Además, indicar el impacto en la seguridad.
  3. Modificar parámetro para maximizar el ancho de banda disponible y configurar autenticación de dos factores (2FA) para el cliente.

# REFERENCIAS

**[11] "Cómo instalar y configurar el servidor OpenVPN en Debian 12" Linux-Console.net. https://es.linux-console.net/?p=30676 (accessed Dec 08. 2024)**