Creación de Redes Privadas Virtuales utilizando la Minicomputadora Raspberry Pi

M. en C. Cyntia E. Enríquez Ortiz

M. en C. Raúl Fernández Zavala

IPN-UPIITA

Academia de Telemática

Resumen

*Una **red privada virtual** (**VPN, Virtual Private Network**), es una tecnología que permite extender de manera segura el alcance de una red de área local (LAN) sobre una red pública no controlada como Internet, es decir, permite que cualquier dispositivo en la red de área local envíe y reciba datos sobre redes públicas como si fuera una red privada con toda su funcionalidad y seguridad. En este trabajo, se describe como implementar paso a paso una VPN mediante openVNP en la minicomputadora Raspberry Pi.*

Red Privada Virtual (VPN)

Generalmente, la red interna de una organización es una red de área local (LAN), la cual necesita con frecuencia conectarse a Internet mediante un equipo de interconexión, para permitir que las organizaciones se puedan comunicar con sucursales, clientes o personal que pueden estar alejados geográficamente.

Sin embargo, cuando los datos son enviados a través de Internet son mucho más vulnerables que cuando viajan por la red interna de la organización, ya que la ruta tomada no está definida por anticipado, y los datos deben atravesar una infraestructura de red pública que pertenece a distintas entidades administrativas. Debido a esto, es posible que a lo largo de la ruta seguida por los datos, algún usuario ajeno a la organización, pueda escucharlos o incluso alterarlos. Por lo tanto, la información confidencial de una organización no puede ser enviada bajo estas condiciones [1].

Una forma de solucionar este problema es el uso de una red privada virtual *(VPN),* la cual es una red privada construida dentro de una infraestructura de red pública, tal como Internet. Se dice que es *virtual* porque conecta dos redes físicas (redes de área local) a través de una conexión poco fiable como Internet y *privada* porque sólo los equipos que pertenecen a una red de área local de uno de los lados de la VPN pueden ver los datos transmitidos. Por lo tanto, el sistema VPN brinda una conexión segura a un bajo costo [2].

Las VPN proporcionan el mayor nivel de seguridad posible mediante el uso de IPsec (IP cifrado) o túneles VPN de SSL (Secure Sockets Layer) y tecnologías de autenticación. El objetivo de todas estas tecnologías es proteger a los datos que pasan por la red privada virtual contra accesos no autorizados. Además, con el uso de las VPN, las empresas pueden aprovechar la infraestructura de Internet y aumentar el alcance de su red privada sin incrementar significativamente su propia infraestructura.

Funcionamiento de una VPN

Una VPN se basa en un *protocolo túnel*, es decir, un protocolo que cifra los datos que se transmiten desde un lado de la VPN hacia el otro como se observa en la figura 1. La palabra *túnel* simboliza el hecho que los datos están cifrados desde el momento que entran a la VPN hasta que salen de ella y, por lo tanto, son incomprensibles para cualquiera que no se encuentre en uno de los extremos de la VPN, como si los datos viajaran a través de un túnel.

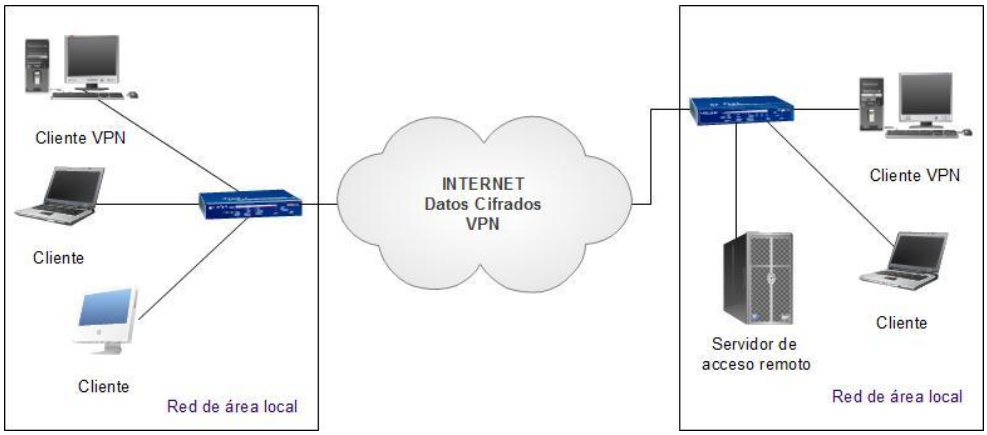

Figura 1. Red Virtual Privada

En una VPN, el cliente de VPN es la parte que cifra y descifra los datos del lado del usuario y el servidor VPN (comúnmente llamado servidor de acceso remoto) es el elemento que descifra los datos en el lado de la organización. De esta manera, cuando un usuario necesita acceder a la red privada virtual, el cliente VPN se conecta con la red remota mediante la infraestructura de red pública como intermediaria; pero transmite la solicitud de manera cifrada. El equipo remoto le proporciona los datos solicitados al servidor VPN en su red y éste envía la respuesta cifrada al cliente VPN. Cuando éste recibe los datos, los descifra y finalmente los entrega al usuario.

OpenVPN es una herramienta que permite crear una VPN, el tráfico entre los nodos de la red se transmite cifrado en una conexión SSL. El servidor puede verificar que los clientes que se conectan son clientes autorizados gracias al uso de certificados de seguridad. De la misma forma, los clientes pueden verificar que se están comunicando con el servidor real y no con otro equipo que pretende serlo.

Creación de una Autoridad Certificadora

OpenVPN se basa en OpenSSL para implementar criptografía SSL/TLS, esto se puede realizar de dos formas, mediante la configuración de una clave privada compartida o configurando un certificado de seguridad con el estándar X.509 basado en infraestructura de llave pública. En este trabajo, se presenta la segunda opción.

El primer paso consiste en la instalación de openVPN en Raspbian mediante el comando *$sudo apt-get install openvpn.* Enseguida, se debe acceder a */usr/share/doc/openvpn/examples* y copiar la carpeta *easy-rsa*, la cual permite crear una autoridad certificadora propia.

$ sudo cp -r /usr/share/doc/openvpn/examples/easy-rsa/2.0 /etc/openvpn/

Posteriormente dentro de esta carpeta se debe editar el archivo *vars* con información propia. Por ejemplo:

```
1  export KEY_COUNTRY="MX"
2  export KEY_PROVINCE="DISTRITO FEDERAL"
3  export KEY_CITY="Distrito Federal"
4  export KEY_ORG="UPIITA"
5  export KEY_EMAIL="micorreo@upiita.ipn.mx"
6  export KEY_CN= minombrecomun
7  export KEY_NAME=minombredeclave
8  export KEY_OU=IT
```

Una vez editado el archivo se deben exportar los datos mediante el comando *$ sudo #source vars,* una forma de verificar que las variables se han exportado correctamente, es comprobando el valor de cualquiera de ellas mediante el comando *#echo $nombre_var,* el cual debe devolver el nombre que se haya introducido a la variable previamente. Se recomienda antes de seguir con la creación de los certificados

ejecutar el script encargado de eliminar las posibles carpetas de claves creadas con anterioridad que es:

 # ./clean-all

El siguiente paso es crear el par de claves de la propia Autoridad Certificadora utilizando el script:

 # ./build-ca

Al ejecutar este script se preguntará acerca de los datos configurados en el archivo *vars,* una vez introducidos los datos solicitados, se debe crear el certificado para el servidor VPN (clave privada) y los parámetros Diffie-Hellman utilizados para establecer la conexión SSL/TLS. Para crear el certificado del servidor se ejecuta el script:

 # ./build-key-server raspberry.home.local

Se volverán a preguntar los mismos datos que cuando se generaron los certificados de la autoridad certificadora. Para crear los parámetros Diffie-Hellman el script que debe utilizarse es:

 # ./build-dh

Además, es necesario también crear la clave para cada uno de los usuarios que se conectarán vía VPN mediante el script:

 # ./build-key Nombre usuario

A continuación, se debe acceder a la carpeta */etc/openvpn/* donde se guardó el archivo *vars* y verificar que se encuentren los siguientes archivos:

- *ca.crt:* contiene el certificado público de la autoridad certificadora, el cual se tiene que usar en todos los clientes y el servidor.
- *raspberry.home.local.crt:* contiene el certificado público del servidor.
- *raspberry.home.local.key:* contiene el certificado privado del servidor.
- *dh1024.pem:* contiene los parámetros Diffie-Hellman y se debe ubicar sólo en la carpeta del servidor.
- *nombreusuario.crt:* contiene el certificado público de usuario.
- *nombreusuario.crt:* contiene el certificado privado de usuario.

Por último se deben distribuir estos archivos dentro del servidor:

 # cp ca.crt /etc/ssl/certs/UPIITA_CA.crt
 # cp raspberry.home.local.crt /etc/ssl/certs/
 # cp raspberry.home.local.key /etc/ssl/private/
 # cp dh1024.pem /etc/openvpn/

Configuración del Servidor OpenVPN en Raspbian

Una vez creada la autoridad certificadora y los certificados necesarios se debe configurar el servidor OpenVPN. Como primer paso se debe copiar el archivo de configuración por defecto para trabajar sobre él a */etc/openvpn/*:

$sudo cp /usr/share/doc/openvpn/examples/simple-config-files/server.conf.gz /etc/openvpn/

Posteriormente el archivo *server.conf.gz* se debe descomprimir y editar

$ cd /etc/openvpn

$ gunzip server.conf.gz

$ sudo nano server.conf

El archivo es muy extenso pero al menos se deben modificar las siguientes opciones:

 # Los certificados creados
ca /etc/ssl/certs/UPIITA_CA.crt
cert /etc/ssl/certs/raspberry.home.local.crt
key /etc/ssl/private/raspberry.home.local.key

 # Los parámetros Diffie-Hellman
dh /etc/openvpn/dh1024.pem

 # Habilitar a los clientes para que puedan acceder a la subred interna
push "route dir_IP mascara_subred"

Por último se debe reiniciar el servicio OpenVPN para crear el túnel *tun0* mediante el comando:

$ sudo /etc/init.d/openvpn restart

Configuración de IP forwarding en Raspbian

Una vez configurado el servidor VPN es necesario permitirles a los clientes alcanzar cualquier red fuera de la red local configurada, ya que por defecto, Raspian cuando recibe paquetes de un cliente en la interfaz virtual *tun0* no permite que estos viajen a ninguna otra red. Para poder darles acceso a los clientes a cualquier red externa es necesario utilizar un mecanismo conocido como *IP forwarding*, por lo que primero se

tiene que habilitar este mecanismo, para lo cual se debe editar el archivo */etc/sysctl.conf* y quitar el comentario de la línea:

net.ipv4.ip_forward = 1

Por último se debe configurar *IPtables* con las siguientes reglas para que el tráfico se enrute correctamente:

$ sudo iptables –A FORWARD –i eth0 –o tun0 –m state –state ESTABLISHED, RELATED –j ACCEPT
$ sudo iptables –A FORWARD –s 10.8.0.0/24 –eth0 –j ACCEPT
$ sudo iptables –t nat -A POSTROUTING –s 10.8.0.0/24 –o eth0 -j MASQUERADE

Por último para que estas reglas sean permanentes se debe instalar el paquete *iptables-persistent* que guardará la configuración actual y la volverá a cargar después de cada reinicio, con el comando:

$ sudo apt-get install iptables-persistent

Conclusiones

La minicomputadora Raspberry Pi es una buena opción para organizaciones de bajo presupuesto que requieran un servidor de acceso remoto que sea capaz de crear una VPN entre dos redes de área local, de manera que puedan disponer de una conexión virtual segura para la transferencia de su información confidencial.

Bibliografía

1. Carazo Gil Francisco J., “Ubuntu Linux. Instalación y Configuración Básica en Equipos y Servidores”, RA-MA, *2009*.

2. Forouzan Behrouz, “TCP/IP Protocols Suite”, Mc. Graw Hill, *2010*.

[3 Upton Eben, Halfacree Gareth, “Meet the Raspberry Pi”, Wiley, *2012*.

[4] Andrew Hudson, Paul Hudson, “La Biblia De Ubuntu”, Anaya Multimedia, *2008*.