**INTRODUCCION A LAS REDES PRIVADAS VIRTUALES (VPN)**
**BAJO GNU/LINUX**

Por Ramiro J. Caire <ramiro@lugro.org.ar>

## 1- INTRODUCCION:

*¿Que es una VPN?*
Consideraciones generales:

VPN (Virtual Private Network) es una extensión de una red local y privada que utiliza como medio de enlace una red publica como por ejemplo, Internet. También es posible utilizar otras infraestructuras WAN tales como Frame Relay, ATM, etc.
Este método permite enlazar dos o mas redes simulando una única red privada permitiendo así la comunicación entre computadoras como si fuera punto a punto.
También un usuario remoto se puede conectar individualmente a una LAN utilizando una conexión VPN, y de esta manera utilizar aplicaciones, enviar datos, etc. de manera segura.
Las Redes Privadas Virtuales utilizan tecnología de túnel (*tunneling*) para la transmisión de datos mediante un proceso de encapsulación y en su defecto de encriptación, esto es importante a la hora de diferenciar Redes Privadas Virtuales y Redes Privadas, ya que esta ultima utiliza líneas telefónicas dedicadas para formar la red. Mas adelante explicare mas en profundidad el funcionamiento del túnel.
Una de las principales ventajas de una VPN es la seguridad, los paquetes viajan a través de infraestructuras publicas(Internet) en forma encriptada y a través del túnel de manera que sea prácticamente ilegible para quien intercepte estos paquetes.
Esta tecnología es muy útil para establecer redes que se extienden sobre áreas geográficas extensas, por ejemplo diferentes ciudades y a veces hasta piases y continentes. Por ejemplo empresas que tienen oficinas remotas en puntos distantes, la idea de implementar una VPN haría reducir notablemente los costos de comunicación, dado que las llamadas telefónicas (en caso de usar dial-up) serian locales(al proveedor de Internet) o bien utilizar conexiones DSL, en tanto que de otra manera habría que utilizar líneas dedicadas las cuales son muy costosas o hacer tendidos de cables que serian mas costosos aun.

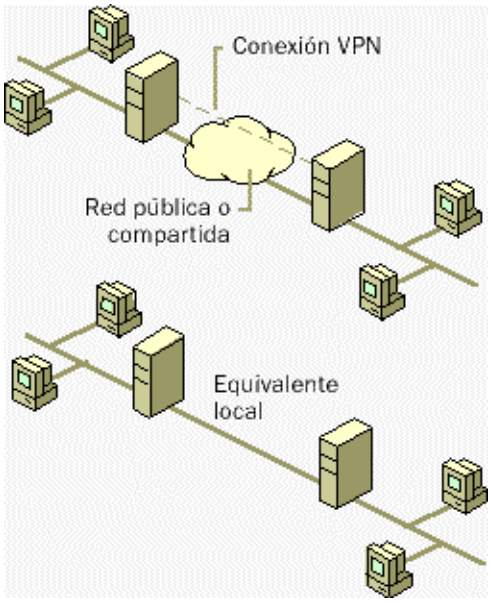



*Diagrama lógico de una VPN*

## 1.2- VENTAJAS DE UNA VPN:

-Seguridad: provee encriptación y Encapsulación de datos de manera que hace que estos viajen codificados y a través de un túnel.
-Costos: ahorran grandes sumas de dinero en líneas dedicadas o enlaces físicos.
-Mejor administración: cada usuario que se conecta puede tener un numero de IP fijo asignado por el administrador, lo que facilita algunas tareas como por ejemplo mandar impresiones remotamente, aunque también es posible asignar las direcciones IP dinámicamente si así se requiere.
-Facilidad para los usuarios con poca experiencia para conectarse a grandes redes corporativas transfiriendo sus datos de forma segura.

## 1.3 TIPOS DE VPN:

Las formas en que pueden implementar las VPNs pueden ser basadas en HARDWARE o a través de SOFTWARE, pero lo mas importante es el protocolo que se utilice para la implementación.
Las VPNs basadas en HARDWARE utilizan básicamente equipos dedicados como por ejemplo los routers, son seguros y fáciles de usar, ofreciendo gran rendimiento ya que todos los procesos están dedicados al funcionamiento de la red a diferencia de un sistema operativo el cual utiliza muchos recursos del procesador para brindar otros servicios, en síntesis, los equipos dedicados son de fácil implementación y buen rendimiento, solo que las desventajas que tienen son su alto costo y que poseen sistemas operativos propios y a veces también protocolos que son PROPIETARIOS.

**Existen diferentes tecnologías para armar VPNs:**
-DLSW: Data Link Switching(SNA over IP)
-IPX for Novell Netware over IP
-GRE: Generic Routing Encapsulation
-ATMP: Ascend Tunnel Management Protocol
-IPSEC: Internet Protocol Security Tunnel Mode
-PPTP: Point to Point Tunneling Protocol
-L2TP: Layer To Tunneling Protocol

entre los mas usados y con mejor rendimiento estarían Ipsec y PPTP, aunque a este ultimo se le conocen fallas de seguridad.

A continuación se detallan su funcionamiento:

**IPSEC (Internet Protocol Secure):**
Es un protocolo de seguridad creado para establecer comunicaciones que proporcionen confidencialidad e integridad de los paquetes que se transmiten a través de Internet.
IPsec puede utilizar dos métodos para brindar seguridad, ESP (Encapsulating Security Payload) o AH (Authentication Header).
La diferencia entre ESP y AH es que el primero cifra los paquetes con algoritmos de cifrado definidos y los autentica, en tanto que AH solo los autentica.
AH firma digitalmente los paquetes asegurándose la identidad del emisor y del receptor.
Ipsec tiene dos tipos de funcionamiento, uno es el modo transporte en el cual la encriptación se produce de extremo a extremo, por lo que todas las maquinas de la red deben soportar Ipsec, y el otro es el modo túnel, en el cual la encriptación se produce solo entre los routers de cada red.
Esta ultima forma seria la mas ordenada de organizar una red VPN basada en Ipsec.
Existen diferentes productos para implementar VPN con Ipsec en GNU/Linux, pero sin dudas el mas utilizado es el Freswan (http://www.freeswan.org).

**PPTP (Point to Point Tunneling Protocol):**
Este es uno de los protocolos mas populares y fue originalmente diseñado para permitir el transporte (de modo encapsulado) de protocolos diferentes al TCP/IP a través de Internet.
Fue desarrollado por el foro PPTP, el cual esta formado por las siguientes empresas: Ascend Communications, Microsoft Corporations, 3 Com, E.C.I. Telematics y U.S. Robotics(ahora 3 Com).

Básicamente, PPTP lo que hace es encapsular los paquetes del protocolo punto a punto PPP(Point to Point Protocol) que a su vez ya vienen encriptados en un paso previo para poder enviarlos a traves de la red.
El proceso de encriptación es gestionado por PPP y luego es recibido por PPTP, este ultimo utiliza una conexión TCP llamada conexión de control para crear el túnel y una versión modificada de la Encapsulación de Enrutamiento Generico (GRE, Generic Routing encapsulation) para enviar los datos en formato de datagramas IP, que serian paquetes PPP encapsulados, desde el cliente hasta el servidor y viceversa.

El proceso de autenticación de PPTP utiliza los mismos métodos que usa PPP al momento de establecer una conexión, como por ejemplo PAP (Password Authenticaction Protocol) y CHAP (Challenge-Handshake Authentication Protocol).
El método de encriptación que usa PPTP es el *Microsoft Point to Point Encryption*, MPPE, y solo es posible su utilización cuando se emplea CHAP (o MS-CHAP en los NT) como medio de autenticación.
MPPE trabaja con claves de encriptación de 40 o 128 bits, la clave de 40 bits es la que cumple con todos los estándares, en cambio la de 128 bits esta diseñada para su uso en Norte América. Cliente y servidor deben emplear la misma codificación, si un servidor requiere de mas seguridad de la que soporta el cliente, entonces el servidor rechaza la conexión.

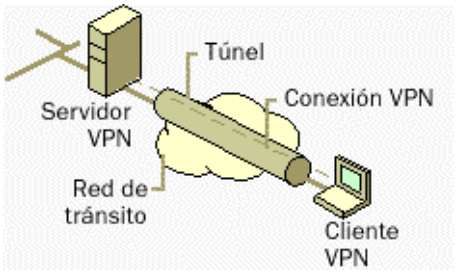
**NOTA:**
Es posible establecer conexiones mediante túneles sin encriptación, es decir, realizar solamente la Encapsulación, pero esto no esta considerado que sea una VPN ya que los datos viajan de forma insegura a través de la red.

**1.4- DIAGRAMAS:**

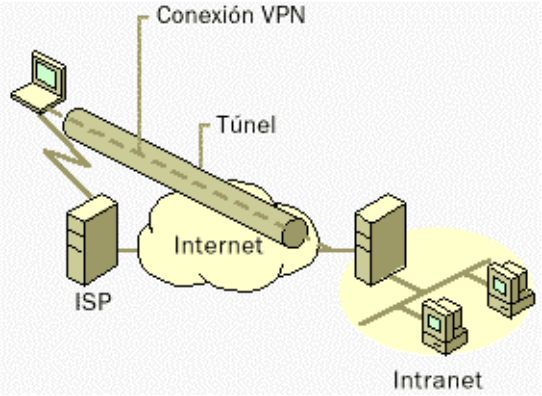
Hay varias posibilidades de conexiones VPN, esto será definido según los requerimientos de la organización, por eso es aconsejable hacer un buen relevamiento a fin de obtener datos como por ejemplo si lo que se desea enlazar son dos o mas redes, o si solo se conectaran usuarios remotos.

Las posibilidades son:

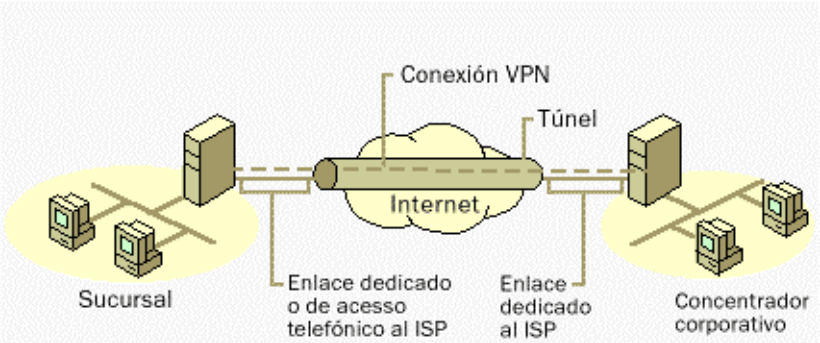
**DE CLIENTE A SERVIDOR(Client to Server):**
Un usuario remoto que solo necesita servicios o aplicaciones que corren en el mismo servidor VPN.




**DE CLIENTE A RED INTERNA (Client to LAN):**
Un usuario remoto que utilizara servicios o aplicaciones que se encuentran en uno o mas equipos dentro de la red interna.




**DE RED INTERNA A RED INTERNA (LAN to LAN):**
Esta forma supone la posibilidad de unir dos intranets a través de dos enrutadores, el servidor VPN en una de las intranets y el cliente VPN en la otra.
Aquí entran en juego el mantenimiento de tablas de ruteo y enmascaramiento.



## 1.5- REQUERIMIENTOS PARA EL ARMADO DE UNA VPN

Para el correcto armado de una VPN, es necesario cumplir con una serie de elementos y conceptos que a continuación se detallan:

***>Tener una conexión a Internet:*** ya sea por conexión IP dedicada, ADSL o dial-up.
***>Servidor VPN:*** básicamente es una pc conectada a Internet esperando por conexiones de usuarios VPN y si estos cumplen con el proceso de autenticación, el servidor aceptara la conexión y dará acceso a los recursos de la red interna.
***>Cliente VPN:*** este puede ser un usuario remoto o un enrutador de otra LAN, tal como se especifica en la sección 1.4 (Diagramas).
***>Asegurarse que la VPN sea capaz de:***

-Encapsular los datos
-Autentificar usuarios.
-Encriptar los datos.
-Asignar direcciones IP de manera estática y/o dinámica.

## 2- IMPLEMENTANDO VPNs CON GNU/LINUX.

Esta sección del documento pretende ayudarlo a configurar un servidor VPN bajo GNU/Linux y a establecer una conexión desde un cliente windows de manera simple.
Para este ejemplo de configuración me basare en el protocolo PPTP utilizando la distribución Debian SID, es decir una estructura CLIENT TO LAN.
Para la gestión de Redes Privadas Virtuales con PPTP bajo GNU/Linux existen diferentes herramientas, obviamente libres, pero sin duda el mas usado es el PPTPD (Point to Point Tunneling Protocol Daemon http://www.poptop.org) que oficia de servidor VPN y puede ser accesado tanto por clientes que corran windows como tambien GNU/Linux.
En tanto que para los clientes GNU/Linux, el programa a usar es el pptp-linux, que sirve para acceder a servidores VPN corriendo GNU/Linux , Windows NT o 2000 Server.

Los pasos a tratar son:

-Soporte del kernel para encriptación MPPE.
-Configurar PPP para encriptación con MPPE.
-Configuración del daemon PPTPD.
-Administración de usuarios.
-Filtrado de paquetes (básico).
-Conexión de clientes.

### 2.1- CONFIGURACION DEL KERNEL CON SOPORTE MPPE:

En todo momento asumo que trabajamos como usuario root.
Instale los fuentes del kernel para su posterior compilación, es aconsejable bajarse la ultima versión estable del kernel en http://www.kernel.org o bien, si su distribución es Debian puede instalarlo mediante apt. Al momento de escribir este documento se uso la versión 2.4.20.
También hay que conseguir el source del patch para MPPE para poder aplicárselo al kernel.
Si no desea recompilar el kernel, hay imágenes precompiladas que ya poseen el soporte para MPPE, debe buscar la versión que mas se adecue a la arquitectura de su equipo.

Obtenemos los fuentes del kernel y herramientas para la compilación y lo preparamos para la compilación (al modo Debian) :

```
# apt-get install kernel-package gcc bin86 bzip2 kernel-source-2.4.20
#cd /usr/src
#bzip2 -d kernel-source-2.4.20.tar.bz2
#tar -xvf kernel-source-2.4.20.tar.bz2
```

Siempre es conveniente, para evitar errores en la compilación, hacer un enlace simbólico llamado linux al directorio que contiene los sources.

```
#ln -s kernel-source-2.4.20 linux
#cd linux
#make-kpkg clean
# cp /boot/config-2.4.20 ./.config
```

Ahora le aplicaremos el patch para soporte MPPE:

```
# cd /usr/src
# wget http://quozl.netrek.org/pptp/ppp-2.4.2_cvs_20021120.tar.gz
# tar -xfz ppp-2.4.2_cvs_20021120.tar.gz
# cd ppp-2.4.2_cvs_20021120/linux/mppe
# chmod +x mppeinstall.sh
# ./mppeinstall.sh /usr/src/kernel-source-2.4.20
```

Editamos el archivo /usr/src/kernel-source-2.4.20/.config y buscamos la línea que hace referencia al PPP y la dejamos de la siguiente manera, CONFIG_PPP=m , luego buscamos otra línea que hace referencia al MPPE y quedaría así, CONFIG_PPP_MPPE=m.

El próximo paso es compilar el kernel e instalarlo:

```
# make-kpkg –append_to_version -mppe –initrd kernel-image
```

Este proceso puede tomar unos cuantos minutos, depende la maquina en la que se este compilando.

```
# dpkg -i kernel-image-2.4.20-mppe_10.00.Custom_i386.deb
```

Ahora solo resta reiniciar el equipo con el nuevo kernel(recuerde configurar su gestor de arranque).
Una vez reiniciado, asegúrese que esta corriendo el nuevo kernel:

```
#uname -r
```

Si el kernel es el correcto, entonces vamos a testear el soporte para MPPE:

```
#modprobe ppp_mppe
```

Aparecera un mensaje pero será solo de advertencia, no impedirá que trabaje bien.

**2.2- CONFIGURACION DE PPPD CON SOPORTE MPPE:**

Compilamos el nuevo pppd:

```
# cd /usr/src/ppp-2.4.2_cvs_20021120/linux/mppe
# make
```

Luego salvamos el pppd actual por si hay que volver atrás, para eso solo lo renombramos, y usamos el comando dpkg-divert el cual elimina la versión de pppd instalada por Debian, pero guardando los atributos en el archivo copiado.

```
# cp /usr/sbin/pppd /usr/sbin/pppd.debian
# dpkg-divert –divert /usr/sbin/pppd.debian /usr/sbin/pppd
# cp pppd/pppd /usr/sbin/pppd
```

**2.3- CONFIGURANDO EL SERVIDOR VPN (PPTPD).**

Descargar la ultima versión de los fuentes de PPTPD (al momento de escribir este documento es pptpd-1.4-b2.tar.gz).
Descompactar y compilar del siguiente modo

```
# tar -xvzf pptpd-1.4-b2.tar.gz
# cd poptop-1.1.4
# ./configure --prefix=/usr/sbin
# make
# make check
# make install
```

Copiar el archivo pptpd.init (ubicado en el directorio de la compilacion) a /etc/init.d (en Red Hat seria /etc/rc.d/init.d) y darle permisos de ejecución:

```
# chmod 755 pptpd.init
```

Luego hacer el link al runlevel adecuado:

```
#cd /etc/rc2.d/  (en debian el runlevel 2 es modo multiuser)
#ln -s ../init.d/pptpd.init S97vpn
```

Editar el archivo pptpd.init y cambiar la linea 23
"daemon /usr/sbin/pptpd" por "/usr/sbin/pptpd -d"

Editar el /etc/pptpd.conf y dejarlo de la siguiente manera

```
debug    #Esta línea puede ser removida una vez que la vpn esta funcionando
option /etc/ppp/options.pptp
localip 192.168.1.1
remoteip 192.168.1.10-50
```

**localip:** IP que tendrá el servidor vpn (IP del tunel), esto puede ser un rango, por lo cual asignara(en el servidor) una IP por cada túnel que se genere. A veces para mejor control es mejor que el server tenga la misma IP siempre.

**remoteip:** rango de IPs que se asignan a los usuarios que se conecten, en caso de proveer una sola IP, permitirá un cliente por vez, pero todos los clientes obtendrán esa misma IP. Es importante tener en cuenta la cantidad de conexiones que se van a permitir, porque si se requiere asignar direcciones IP estáticas a usuarios (esto se hace en el /etc/chap-secrets) las IPs deberán estar contempladas en el rango.

Ahora deberemos editar el archivo /etc/ppp/options.pptp que debería tener como mínimo las siguientes opciones:

```
 lock
 debug
+chap
+chapms
+chapms-v2
mppe-40
mppe-128
mppe-stateless
proxyarp
```

y con esto quedaría configurado el servidos PPTPD, ahora pasemos a dar de alta los usuarios VPN.

**2.4- CONFIGURANDO USUARIOS VPN.**

La configuración de usuarios se hace en el archivo /etc/ppp/chap-secrets, no es necesario que el usuario que se conecte por VPN tenga que ser usuario del sistema:

La primer columna corresponde a el nombre de usuario(para autenticar dominios de windows, poner el nombre de dominio mas dos barras invertidas antes del usuario)

En la segunda columna va el nombre del servidor (puede ir solamente un asterico asumiendo que es el mismo donde esta corriendo el pptp)

En la tercer columna va el password

En la cuarta columna va la IP, en caso de asignar estáticamente una a un usuario, de lo contrario, con un asterisco (*) el servidor le dará a ese usuario una IP arbitraria que va a estar comprendida en el rango que especifico en /etc/pptpd.conf ).

Ejemplo de /etc/ppp/chap-secrets:

```
juan        *        bostero     *
pedro       *        de3th58     192.168.1.22
alberto     *        123456      192.168.1.23
```

Para autenticar una maquina windows:

```
    Ej: dominio\\pedro          *          d3th58      192.168.1.22
```

Habrá tantos usuarios en el chap-secrets como conexiones se permitan realizar(no se pueden loguear dos usuarios al mismo tiempo) , recordemos también que en este archivo se encuentran los nombres de usuario y password de NUESTRO acceso a internet ;).

**2.5- FILTRADO DE PAQUETES.**

Si tenemos un firewall en el servidor VPN, deberemos agregar algunas reglas de iptables para permitir que se establezca el túnel, los puertos que nos interesan son el 1723 y el 47.

Reglas de entrada:

```
# iptables -A INPUT -p tcp --dport 1723 -j ACCEPT
# iptables -A INPUT -p 47 -j ACCEPT
```

Reglas de salida:

```
# iptables -A OUTPUT -p tcp --sport 1723 -j ACCEPT
# iptables -A OUTPUT -p 47 -j ACCEPT
```

Ahora solo resta por levantar el demonio pptpd del siguiente modo:

```
#/etc/init.d/pptpd.init start
```

o directamente

```
# pptpd
```

El servidor PPTPD se pondrá automáticamente en background escuchando por el puerto TCP 1723 en espera de conexiones entrantes.

**2.4- CONFIGURANDO CLIENTES VPN.**

Configuración de clientes bajo windows:

1- Primero deberemos dar soporte a windows para establecer conexiones vpn.
Desde las propiedades de red, agregar dispositivo(o adaptador) para redes privadas virtuales.

2- Ir a Panel de Control > Configuración de Acceso Telefónico a redes.
Crear un nuevo perfil de conexión utilizando el dispositivo con soporte para vpn, estableciendo el nombre o numero IP del servidor vpn, nombre de usuario que hemos establecido en el /etc/chap-secrets del equipo servidor y la misma password.

3- Luego de creada la conexión, entramos a Propiedades y le decimos (tildando las opciones) que queremos cifrado de datos y contraseña cifrada.

Llegados a este punto, ya se puede establecer la conexión cliente<->servidor, conectando como si lo hiciéramos a un isp normal.

---

Bueno, espero que tengan suerte con esta configuración que es algo basica, cualquier consulta pueden hacerla a ramiro@lugro.org.ar
Saludos.-