Cómo configurar un servidor de OpenVPN

Autor: *Joel Barrios Dueñas*
Correo electrónico: *darkshram en gmail punto com*
Sitio de Red: *http://www.alcancelibre.org/*
Jabber ID: *darkshram@jabber.org*

Introducción.

Acerca de OpenVPN.

OpenVPN es una solución de conectividad basada sobre equipamiento lógico (*software*): SSL(Secure Sockets Layer) VPN (Virtual Private Network, o red virtual privada), OpenVPN ofrece conectividad punto-a-punto con validación, jerárquica de usuarios y host conectados remotamente, resulta una muy buena opción en tecnologías Wi-Fi (redes inalámbricas EEI 802.11) y soporta una amplia configuración, entre éstas el balanceo de cargas, entre otras muchas cosas más.

URL: http://openvpn.net

Breve explicación de lo que se logrará con este documento.

Este documento describe la configuración de una **VPN** tipo **Intranet**.

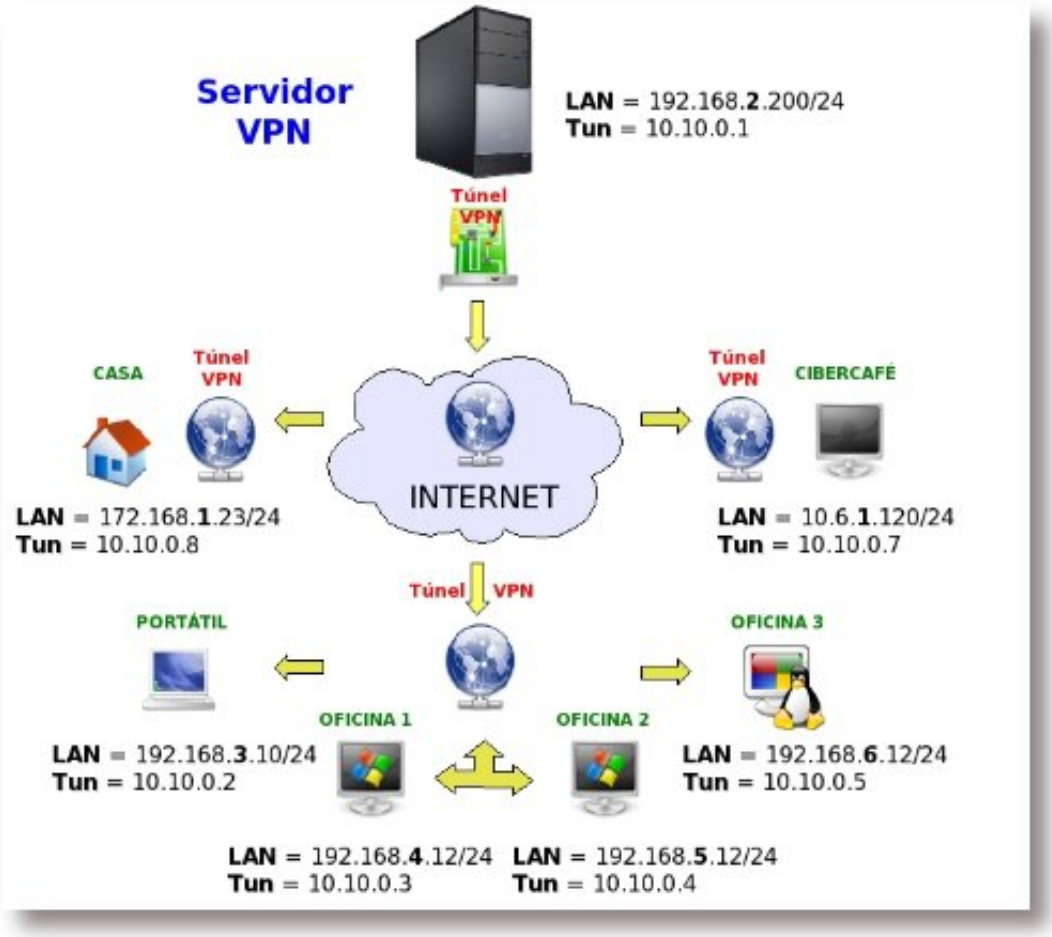
Este tipo de redes es creado entre una oficina central (servidor) y una o varias oficinas remotas (clientes). El acceso viene del exterior. Se utiliza este tipo de VPN cuando se necesita enlazar a los sitios que son parte de una compañía, en nuestro caso será compuesto por un servidor Central que conectará a muchos clientes VPN entre si.

La información y aplicaciones a las que tendrán acceso los directivos móviles en el VPN, no serán las mismas que aquellas en donde pueden acceder los usuarios que efectúan actividades de mantenimiento y soporte, esto como un ejemplo de lo que se podrá realizar con esta configuración.

Ademas de que podrá conectarse a través de **Terminal Server** (***en el caso de clientes Linux***)

a terminales Windows de la red VPN así como de Clientes Windows a computadoras con el mismo sistema operativo (mediante RDP).

Nota Importante: Enfocado a esta configuración .. Una vez que los clientes (***Windows/Linux***) se conecten a la red VPN quedarán automáticamente sin conexión a Internet, lo cual NO podrán acceder a la red mundial. Esto puede ser modificable en el servidor VPN.



Servidor de Pasarela OpenVPN con clientes (Windows/Linux) remotos

El servidor VPN **hace de pasarela** para que todos los clientes (Windows/Linux) puedan estar **comunicados** a través del túnel OpenVPN, estos al conectarse por medio de Internet al túnel automáticamente quedan **sin linea**a la red mundial quedando como una **red local**, esto claro esta a través de la VPN.

Cada cliente se encuentra en lugares diferentes (ciudad/estado/país) con diferentes tipos de segmento de red, al estar conectados mediante el túnel VPN se crea un red virtual y se asigna un nuevo segmento de red proporcionada por el servidor principal en este caso con segmento (por ejemplo 10.10.0.0/255.255.255.0no 192.168.37.0/255.255.255.0).

Instalación del equipamiento lógico necesario.

Fedora 9 en adelante incluye el paquete **openvpn** en sus depósitos Yum, por lo que solo es necesario instalarlo desde la terminal a través del mandato **yum**. El siguiente procedimiento solo es necesario para **CentOS 5**.

Instalación en CentOS 5.

Como el usuario **root**, desde una terminal, crear el archivo **/etc/yum.repos.d/AL-Server.repo**,

utilizando cualquier editor de texto. En el siguiente ejemplo se utiliza **vi**.

```
vi /etc/yum.repos.d/AL-Server.repo
```

Añadir a este **nuevo archivo** el siguiente contenido:

```
[AL-Server]
name=AL Server para Enterprise Linux $releasever
mirrorlist=http://www.alcancelibre.org/al/el$releasever/al-server
gpgcheck=1
gpgkey=http://www.alcancelibre.org/al/AL-RPM-KEY
```

Importar la firma digital de **Alcance Libre** ejecutando lo siguiente desde la terminal:

```
rpm --import http://www.alcancelibre.org/al/AL-RPM-KEY
```

Luego de importar la firma digital de Alcance Libre, instalar el equipamiento lógico (*software*) necesario con el mandato **yum**. Se requieren los paquetes RPM de OpenVPN, Shorewall y vim-enhanced (la versión mejorada de Vi):

```
yum -y install openvpn shorewall vim-enhanced
```

Procedimientos.

Si fuera necesario, cambiarse al usuario **root** utilizando el siguiente mandato:

```
su -l
```

A fin de poder utilizar inmediatamente la versión mejorada de **Vi** (instalado con el paquete **vim-enhanced**), ejecutar desde la terminal lo siguiente:

```
alias vi="vim"
```

Cambiarse al directorio, desde la terminal, ejecutar lo siguiente para cambiarse al directorio **/etc/openvpn**:

```
cd /etc/openvpn/
```

NOTA: Todos los procedimientos necesarios para configurar un servidor con **OpenVPN** se

realizan sin salir de **/etc/openvpn/**. Por favor, **evite cambiar de directorio** hasta haber finalizado los procedimientos descritos en este documento.

A fin de facilitar los procedimientos, se copiarán dentro del directorio **/etc/openvpn/** los archivos **openssl.cnf**, **whichopensslcnf**, **pkitool** y **vars**, que se localizan en **/etc/openvpn/easy-rsa/2.0/**:

```
cp /usr/share/openvpn/easy-rsa/2.0/openssl.cnf ./
cp /usr/share/openvpn/easy-rsa/2.0/whichopensslcnf ./
cp /usr/share/openvpn/easy-rsa/2.0/pkitool ./
cp /usr/share/openvpn/easy-rsa/2.0/vars ./
```

Utilizar el editor de texto y abrir el archivo **/etc/openvpn/vars**:

```
vi /etc/openvpn/vars
```

De este archivo, solamente editar las últimas líneas, que corresponden a lo siguiente:

```
export KEY_COUNTRY="US"
export KEY_PROVINCE="CA"
export KEY_CITY="SanFrancisco"
export KEY_ORG="Fort-Funston"
export KEY_EMAIL="me@myhost.mydomain"
```

Reemplazar por valores reales, como los del siguiente ejemplo:

```
export KEY_COUNTRY="MX"
export KEY_PROVINCE="DF"
export KEY_CITY="Mexico"
export KEY_ORG="servidor.mi-dominio.com"
export KEY_EMAIL="fulanito@mi-dominio.com"
```

Se requiere ejecutar del siguiente modo el archivo **/etc/openvpn/vars** a fin de que carguen las variables de entorno que se acaban de configurar.

```
source /etc/openvpn/./vars
```

Cada vez que se vayan a generar nuevos certificados, debe ejecutarse el mandato anterior a fin de que carguen las variables de entorno definidas.

Se ejecuta el archivo **/usr/share/openvpn/easy-rsa/2.0/clean-all** a fin de limpiar cualquier firma digital que accidentalmente estuviera presente.

```
sh /usr/share/openvpn/easy-rsa/2.0/clean-all
```

Lo anterior realiza un **rm -fr** (eliminación recursiva) sobre el directorio **/etc/openvpn/keys**, por lo que se eliminarán todas los certificados y firmas digitales que hubieran existido con anterioridad.

A fin de crear el certificado del servidor, se crea un certificado:

```
sh /usr/share/openvpn/easy-rsa/2.0/build-ca
```

Se crea el archivo dh1024.pem, el cual contendrá los parámetros del protocolo **D**iffie-**H**ellman, de 1024 bits:

```
sh /usr/share/openvpn/easy-rsa/2.0/build-dh
```

El protocolo **D**iffie-**H**ellman permite el intercambio secreto de claves entre dos partes que sin que éstas hayan tenido contacto previo, utilizando un canal inseguro, y de manera anónima (sin autenticar). Se emplea generalmente como medio para acordar claves simétricas que serán empleadas para el cifrado de una sesión, como es el caso de una conexión VPN.

Para generar la firma digital, se utilizan el siguiente mandato:

```
sh /usr/share/openvpn/easy-rsa/2.0/build-key-server server
```

Finalmente se crean los certificados para los clientes. En el siguiente3 ejemplo se crean los certificados para **cliente1**, **cliente2**, **cliente3**, **cliente4**, **cliente5**, y **cliente6**:

```
sh /usr/share/openvpn/easy-rsa/2.0/build-key cliente1
sh /usr/share/openvpn/easy-rsa/2.0/build-key cliente2
sh /usr/share/openvpn/easy-rsa/2.0/build-key cliente3
sh /usr/share/openvpn/easy-rsa/2.0/build-key cliente4
sh /usr/share/openvpn/easy-rsa/2.0/build-key cliente5
sh /usr/share/openvpn/easy-rsa/2.0/build-key cliente6
```

A fin de utilizar los certificados y que se configure el sistema, se crea con el editor de texto el archivo **/etc/openvpn/servidorvpn-udp-1194.conf**, donde *servidorvpn* se reemplaza por el nombre de anfitrión del sistema:

```
vi /etc/openvpn/servidorvpn-udp-1194.conf
```

Para la **VPN** se recomienda utilizar una red privada que sea poco usual, a fin de poder permitir a los clientes conectarse sin conflictos de red. Un ejemplo de una red poco utilizada sería 192.168.37.0/255.255.255.0, lo cual permitirá conectarse a la **VPN** a 253 clientes. Tomando en cuenta lo anterior, el contenido del archivo **/etc/openvpn/servidorvpn-udp-1194.conf**, debe ser el siguiente:

ser el siguiente:

```
port 1194
proto udp
dev tun
#---- Seccion de llaves -----
ca keys/ca.crt
cert keys/server.crt
key keys/server.key
dh keys/dh1024.pem
#----------------------------
server 192.168.37.0 255.255.255.0
ifconfig-pool-persist ipp.txt
keepalive 10 120
comp-lzo
persist-key
persist-tun
status openvpn-status-servidorvpn-udp-1194.log
verb 3
```

Descripción de los parámetros anteriores:

Port: Especifica el puerto que será utilizado para que los clientes vpn puedan conectarse al servidor.

Proto: tipo de protocolo que se empleará en a conexión a través de VPN

dev: Tipo de interfaz de conexión virtual que se utilizará el servidor openvpn.

ca: Especifica la ubicación exacta del archivo de Autoridad Certificadora [.ca].

cert: Especifica la ubicación del archivo [.crt] creado para el servidor.

key: Especifica la ubicación de la llave [.key] creada para el servidor openvpn.

dh: Ruta exacta del archivo [.pem] el cual contiene el formato de Diffie Hellman (requirerido para **--tls-server**solamente).

server: Se asigna el rango IP virtual que se utilizará en la red del túnel VPN.

Ifconfig-pool-persist: Archivo en donde quedarán registrado las direcciones IP de los clientes que se encuentran conectados al servidor OpenVPN.

Keepalive 10 120 : Envía los paquetes que se manejan por la red una vez cada

10 segundos; y asuma que el acoplamiento es abajo si ninguna respuesta ocurre por 120 segundos.

comp-lzo: Especifica los datos que recorren el túnel vpn será compactados durante la trasferencia de estos paquetes.

persist-key: Esta opción soluciona el problema por llaves que persisten a través de los reajustes SIGUSR1, así que no necesitan ser releídos.

Persist-tun: Permite que no se cierre y re-abre los dispositivos TAP/TUN al correr los guiones up/down

status: archivo donde se almacenará los eventos y datos sobre la conexión del servidor [.log]

verb: Nivel de información (default=1). Cada nivel demuestra todo el Info de los niveles anteriores. Se recomienda el nivel 3 si usted desea un buen resumen de qué está sucediendo.

0 --No muestra una salida excepto errores fatales. **1 to 4 –**Rango de uso normal. **5 --**Salida **R**y **W**caracteres en la consola par los paquetes de lectura y escritura, mayúsculas es usada por paquetes TCP/UDP minúsculas es usada para paquetes TUN/TAP.

Si **SELinux** está activo, es necesario que el directorio **/etc/openvpn** y sus contenidos, tengan los contextos apropiados de esta implementación de seguridad (system_u:object_r:**openvpn_etc_rw_t** para **ipp.txt** y **openvpn-status-servidorvpn-udp-1194.log** y system_u:object_r:**openvpn_etc_t** para el resto del contenido del directorio).

Se utiliza luego el mandato **restorecon** sobre el directorio **/etc/openvpn** a fin de asignar los contextos adecuados.

```
restorecon -R /etc/openvpn/
```

Se crean los archivos **ipp.txt** y **openvpn-status-servidorvpn-udp-1194.log**:

```
cd /etc/openvpn/
touch ipp.txt
touch openvpn-status-servidorvpn-udp-1194.log
```

Si se tiene activo SELinux, estos últimos dos archivos requieren se les asigne contexto de lectura y escritura (**openvpn_etc_rw_t**).

```
cd /etc/openvpn/
chcon -u system_u ipp.txt
chcon -u system_u openvpn-status-servidorvpn-udp-1194.log
chcon -r object_r ipp.txt
chcon -r object_r openvpn-status-servidorvpn-udp-1194.log
chcon -t openvpn_etc_rw_t ipp.txt
chcon -t openvpn_etc_rw_t openvpn-status-servidorvpn-udp-1194.log
```

Los anterior cambia los contextos a usuario de sistema (**system_u**), rol de objeto (**object_r**) y tipo configuración de OpenVPN de lectura y escritura (**openvpn_etc_rw_t**).

Para iniciar el servicio, se utiliza el mandato **service** del siguiente modo:

```
service openvpn start
```

Para que el servicio de OpenVPN esté activo en el siguiente inicio del sistema, se utiliza el mandato **chkconfig** de la siguiente forma:

```
chkconfig openvpn on
```

Configuración de muro cortafuegos con Shorewall.

El siguiente procedimiento considera que se ha configurado un muro cortafuegos apropiadamente, de acuerdo a las indicaciones descritas en el documento titulado **Cómo configurar un muro cortafuegos con Shorewall y tres interfaces de red**.

Independientemente del contenido, en el archivo **/etc/shorewall/zones**, se añade la zona **rem** con el tipo **ipv4**, antes de la última línea.

```
# OpenVPN ----
rem     ipv4
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE
```

Independientemente del contenido, en el archivo **/etc/shorewall/interfaces**, se añade la zona **rem** asociada a la interfaz **tun0**, con la opción **detect**, para detectar automáticamente el número de dirección **IP** de difusión (*broadcast*) y la opción **dhcp**. También debe definirse antes de la última línea del archivo.

```
# OpenVPN ----
rem     tun0            detect          dhcp
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE
```

Independientemente del contenido, en el archivo **/etc/shorewall/policy**, se añade la política deseada para permitir el acceso de los miembros de la **VPN** hacia las zonas que se consideren

apropiadas. En el siguiente ejemplo, se define una política que permite el acceso de las conexiones originadas desde la zona **rem** hacia el cortafuegos, la red pública y la red local. Todo debe definirse antes de la última línea del archivo.

```
fw              all             ACCEPT
loc             all             ACCEPT
# OpenVpn ----
rem             fw              ACCEPT
rem             net             ACCEPT
rem             loc             ACCEPT
# ------------
net             all             DROP    info
all             all             REJECT  info
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE
```

Independientemente del contenido, en el archivo **/etc/shorewall/rules**, se debe abrir en el cortafuegos el puerto 1194 por UDP, para todas las zonas desde las cuales se pretenda conectar clientes a la **VPN**.

```
ACCEPT  net                     fw                      udp     1194
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE
```

Finalmente, se edita el archivo **/etc/shorewall/tunnels** a fin de definir el túnel SSL que será utilizado para el servidor de **VPN** y que permita conectarse desde cualquier ubicación.

```
#TYPE                   ZONE    GATEWAY         GATEWAY
#                                               ZONE
openvpnserver:1194      rem     0.0.0.0/0
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE
```

En lugar de **0.0.0.0/0**, se puede especificar una dirección IP o bien una red desde la cual se quiera establecer las conexiones **VPN**.

Para aplicar los cambios, es necesario reiniciar **shorewall** con el mandato **service**, del siguiente modo:

```
service shorewall restart
```

Configuración de clientes Windows.

A través de OpenVPN GUI.

Instalar **OpenVPN GUI** desde http://openvpn.se/. Se requiere instalar la versión de desarrollo **1.0.3** de **OpenVPN GUI**, compatible con OpenVPN 2.1.x. El cliente es **estable**, siempre que se verifique que funcione adecuadamente la configuración utilizada antes de poner en marcha en un entorno productivo.

Crear el archivo **cliente1-udp-1194.ovpn**, con el siguiente contenido, donde es importante que las rutas definidas sean las correctas, y las diagonales invertidas sean dobles:

```
client
dev tun
proto udp
remote dominio-o-ip.del.servidor.vpn 1194
float
resolv-retry infinite
nobind
persist-key
persist-tun
#------ SECCION DE LLAVES --------
ca "C:\\Archivos de Programa\\OpenVPN\\config\\ca.crt"
cert "C:\\Archivos de Programa\\OpenVPN\\config\\cliente1.crt"
key "C:\\Archivos de Programa\\OpenVPN\\config\\cliente1.key"
ns-cert-type server
#---------------------------------
comp-lzo
verb 3
```

Descripción de los parámetros anteriores:

client: Especifica el tipo de configuración, en este caso tipo cliente OpenVPN.

Port: Especifica el puerto que será utilizado para que los clientes VPN puedan conectarse al servidor.

Proto: tipo de protocolo que se empleará en a conexión a través de VPN

dev: Tipo de interfaz de conexión virtual que se utilizará el servidor openvpn.

remote: Host remoto o dirección IP en el cliente, el cual especifica al servidor OpenVPN.

El cliente OpenVPN puede tratar de conectar al servidor con **host:port** en el orden especificado de las opciones de la opción **--remote**.

float: Este le dice a OpenVPN aceptar los paquetes autenticados de cualquier dirección, no solamente la dirección cuál fue especificado en la opción **--remote**.

resolv-retry: Si la resolución del nombre del anfitrión (*hostname*) falla para **--remote**, la resolución antes de fallar hace una re-comprobación de n segundos.

nobind: No agrega bind a la dirección local y al puerto.

ca: Especifica la ubicación exacta del archivo de Autoridad Certificadora [.ca].

cert: Especifica la ubicación del archivo [.crt] creado para el servidor.

key: Especifica la ubicación de la llave [.key] creada para el servidor OpenVPN.

remote: Especifica el dominio o IP del servidor así como el puerto que escuchara las peticiones para servicio VPN.

comp-lzo: Especifica los datos que recorren el túnel VPN será compactados durante la trasferencia de estos paquetes.

persist-key: Esta opción soluciona el problema por llaves que persisten a través de los reajustes SIGUSR1, así que no necesitan ser releídos.

Persist-tun: Permite que no se cierre y re-abre los dispositivos TAP/TUN al correr los guiones up/down

verb: Nivel de información (default=1). Cada nivel demuestra toda la Información de los niveles anteriores. Se recomienda el nivel 3 si usted desea un buen resumen de qué está sucediendo.

0 --No muestra una salida excepto errores fatales. **1 to 4 –**Rango de uso normal. **5 --**Salida **R**y **W**caracteres en la consola par los paquetes de lectura y escritura, mayúsculas es usada por paquetes TCP/UDP minúsculas es usada para paquetes TUN/TAP.

El cliente necesitará que los archivos **ca.crt**, **cliente1.crt**, **cliente1.key** y **cliente1-udp-1194.ovpn** estén presentes en el directorio **"C:\Archivos de Programa\OpenVPN\config\"**. Estos archivos fueron creados, a través de un procedimiento descrito en este documento, dentro del directorio **/etc/openvpn/keys/** del servidor.

Si se quiere que los clientes de la **VPN** se puedan conectar a la red local, es importante considerar las implicaciones de seguridad que esto conlleva si alguno de los certificados es robado, o bien el cliente se ve comprometido en su seguridad por una intrusión, virus, troyano o gusano. Es preferible que la red de la **VPN** sea independiente a la red local y cualquier otra red, uniendo los servidores y clientes a la **VPN**, independientemente de si éstos están en la red local o una red pública.

Si es imperativo hacer que los clientes de la **VPN** se conecten a la red local, la red desde la cual se conectan los clientes debe ser diferente a la red utilizada en la red local. Por ejemplo: si la red local detrás del servidor de **VPN** es 192.168.0.0/255.255.255.0, 10.0.0.0/255.0.0.0 o

la red local detrás del servidor de **VPN** es 192.168.0.0/255.255.255.0, 10.0.0.0/255.0.0.0 o 172.16.0.0/255.255.0.0, los clientes que se conecten a la **VPN** detrás de un modem ADSL o Cable e intenten establecer conexiones con la red local, muy seguramente tendrán conflictos de red.

Para permitir a los clientes de la **VPN** poder establecer conexiones hacia la red local, se añaden las siguientes líneas en el archivo de configuración de OpenVPN para los clientes, y que definen la ruta para la red local y un servidor DNS que debe estar presente y configurado para permitir **consultas recursivas** a la red de la **VPN**:

```
route 192.168.0.0 255.255.255.0
dhcp-option DNS 192.168.0.1
```

Opcionalmente, también se puede definir un servidor Wins.

```
dhcp-option WINS 192.168.26.1
```

Ejemplo, considerando que la red local es **192.168.26.0/255.255.255.0**:

```
client
dev tun
proto udp
remote dominio-o-ip.del.servidor.vpn 1194
float
resolv-retry infinite
nobind
persist-key
persist-tun
route 192.168.26.0 255.255.255.0
dhcp-option DNS 192.168.26.1
dhcp-option WINS 192.168.26.1
#------ SECCION DE LLAVES --------
ca "C:\\Archivos de Programa\\OpenVPN\\config\\ca.crt"
cert "C:\\Archivos de Programa\\OpenVPN\\config\\cliente1.crt"
key "C:\\Archivos de Programa\\OpenVPN\\config\\cliente1.key"
ns-cert-type server
#---------------------------------
comp-lzo
verb 3
```

Clientes GNU/Linux.

A través del servicio openvpn.

Este es el método que funcionará en prácticamente todas las distribuciones de de GNU/Linux basadas sobre **Red Hat**, **CentOS** y **Fedora**. Se requiere instalar el paquete **openvpn**:

```
yum -y install openvpn
```

Para **CentOS 5**, se requiere haber configurado previamente el depósito de **AL Server**, descrito con anterioridad en este mismo documento.

Para los clientes con GNU/Linux utilizando el servicio **openvpn**, básicamente se utiliza el mismo archivo para **OpenVPN GUI** para Windows, pero definiendo rutas en el sistema de archivos de GNU/Linux. Ejemplo:

```
client
dev tun
proto udp
remote dominio-o-ip.del.servidor.vpn 1194
float
resolv-retry infinite
nobind
persist-key
persist-tun
#------ SECCION DE LLAVES --------
ca /etc/openvpn/keys/ca.crt
cert /etc/openvpn/keys/cliente1.crt
key /etc/openvpn/keys/cliente1.key
ns-cert-type server
#--------------------------------
comp-lzo
verb 3
```

Este archivo se guarda como **/etc/openvpn/cliente1-udp-1194.ovpn**. Requiere que los certificados definidos en la configuración estén en las rutas especificadas dentro del directorio **/etc/openvpn/keys/**.

Para iniciar la conexión hacia la **VPN**, simplemente se inicia el servicio **openvpn**:

```
service openvpn start
```

Para que la conexión se establezca automáticamente cada vez que se inicie el sistema, se utiliza el mandado **chkconfig** de la siguiente manera:

```
chkconfig openvpn on
```

A través de NetworkManager.

NetworkManager es una implementación que permite a los usuarios configurar interfaces de red de todos los tipos, sin necesidad de contar con privilegios de administración en el sistema. Es la forma más flexible, sencilla y práctica de conectarse a una red **VPN**.

Se requiere que los clientes Linux tengan instalado el paquete **NetworkManager-openvpn**, mismo que debe estar incluido en los depósitos Yum de **Fedora 9** en adelante y distribuciones recientes de GNU/Linux. **CentOS 5** carece del soporte para utilizar **NetworkManager-openvpn**, por lo que solo podrá conectarse a la **VPN** a través del método anterior, con el servicio **openvpn**.

Para instalar a través del mandato **yum** en distribuciones basadas sobre **Fedora 9** en adelante, se hace de la siguiente manera:

```
yum -y install NetworkManager-openvpn
```

Se puede reiniciar el sistema para que tengan efectos los cambios, o simplemente reiniciar el servicio **NetworkManager**:

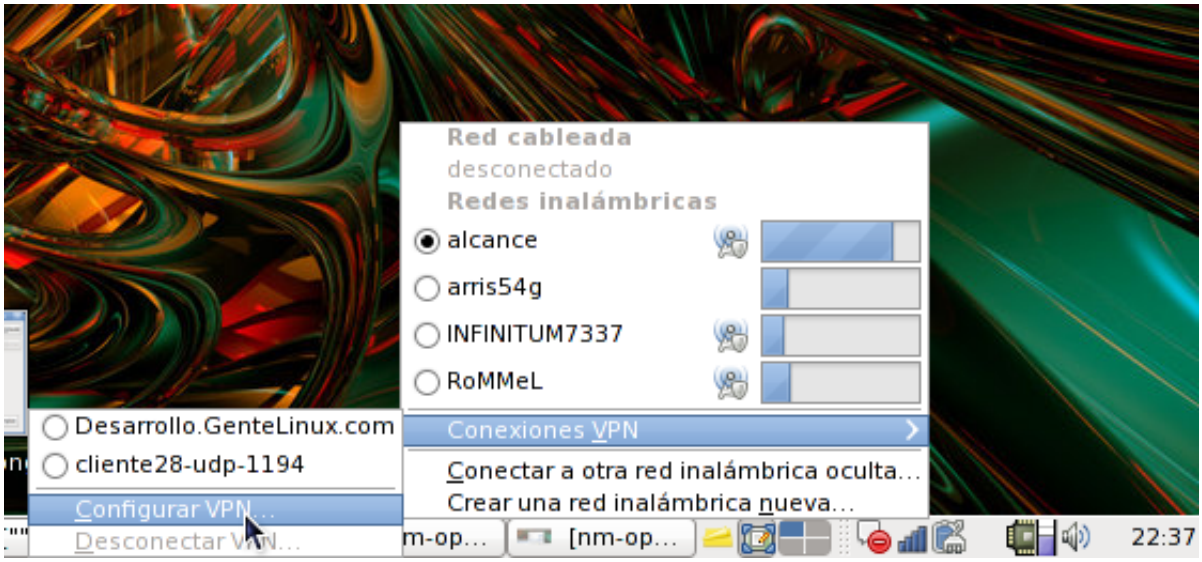
```
service NetworkManager restart
```

Lo anterior cerrará y volverá a establecer las conexiones de red existentes.

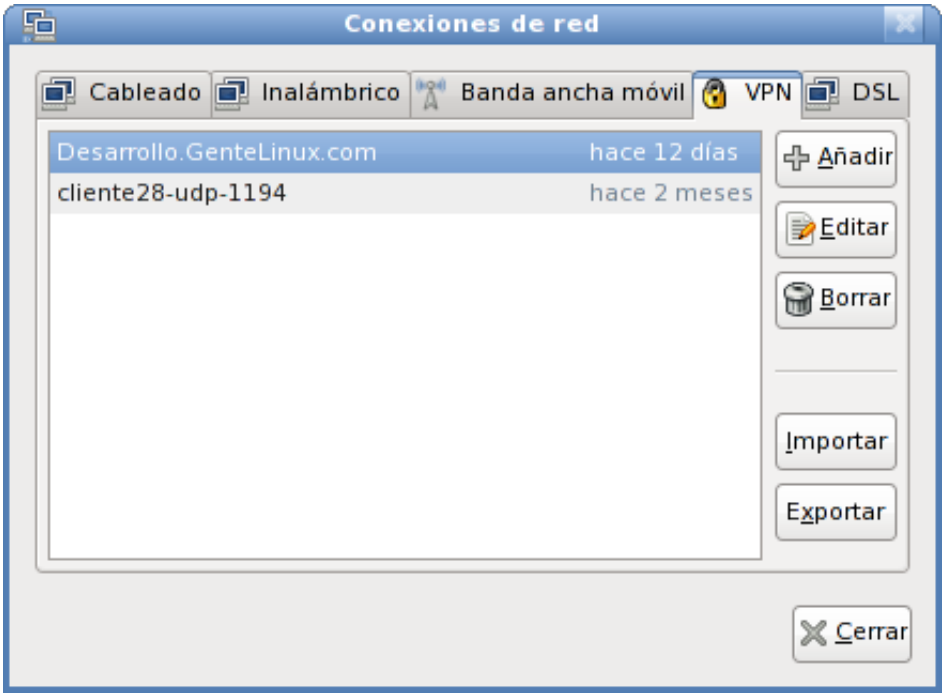
Al igual que el método anterior, para los clientes con GNU/Linux con NetworkManager, básicamente se utiliza el mismo archivo para **OpenVPN GUI** para Windows, pero definiendo rutas en el sistema de archivos de GNU/Linux. Ejemplo:

```
client
dev tun
proto udp
remote dominio-o-ip.del.servidor.vpn 1194
float
resolv-retry infinite
nobind
persist-key
persist-tun
#------ SECCION DE LLAVES --------
ca /etc/openvpn/keys/ca.crt
cert /etc/openvpn/keys/cliente1.crt
key /etc/openvpn/keys/cliente1.key
ns-cert-type server
#--------------------------------
comp-lzo
verb 3
```

Este archivo se puede utilizar con la interfaz gráfica de **NetworkManager**. Solo hay que hacer clic sobre el icono en el **Área de notificación** del panel de GNOME y luego hacer clic en **Configurar VPN**.



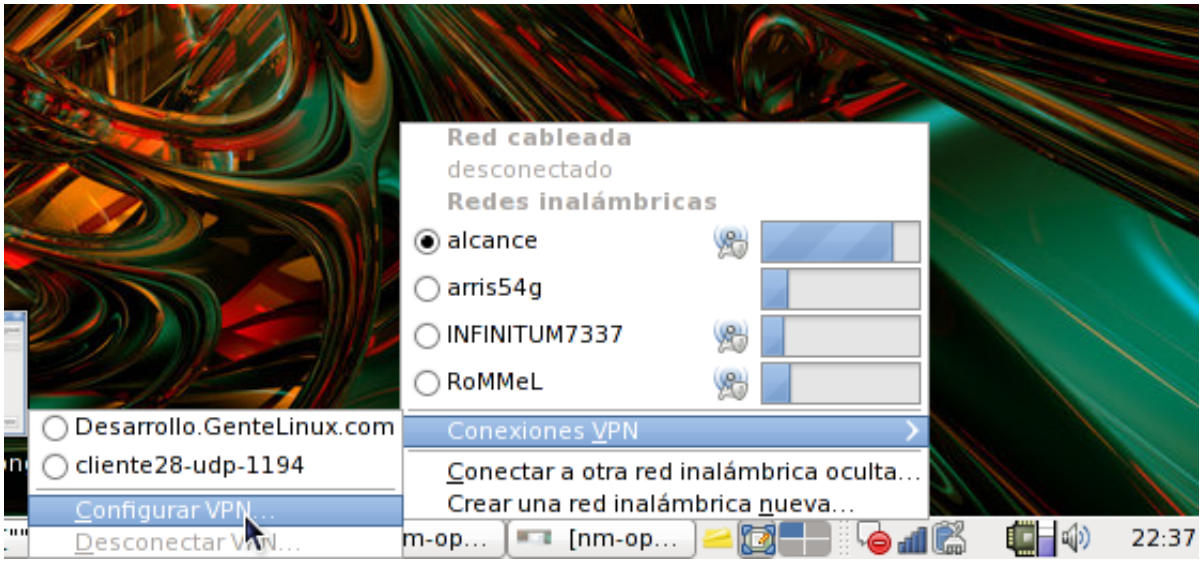
En la ventana que abre a continuación, hay un botón que permite importar el archivo de configuración.



Si los certificados y firma digital son colocados en la ruta **/etc/openvpn/keys/** con SELinux activo, éstos funcionarán adecuadamente. Si los certificados y firma digital son almacenados dentro del directorio de inicio del usuarios, es necesario establecer la política **openvpn_enable_homedirs** con valor **1** (que equivale a **on**, o activa):

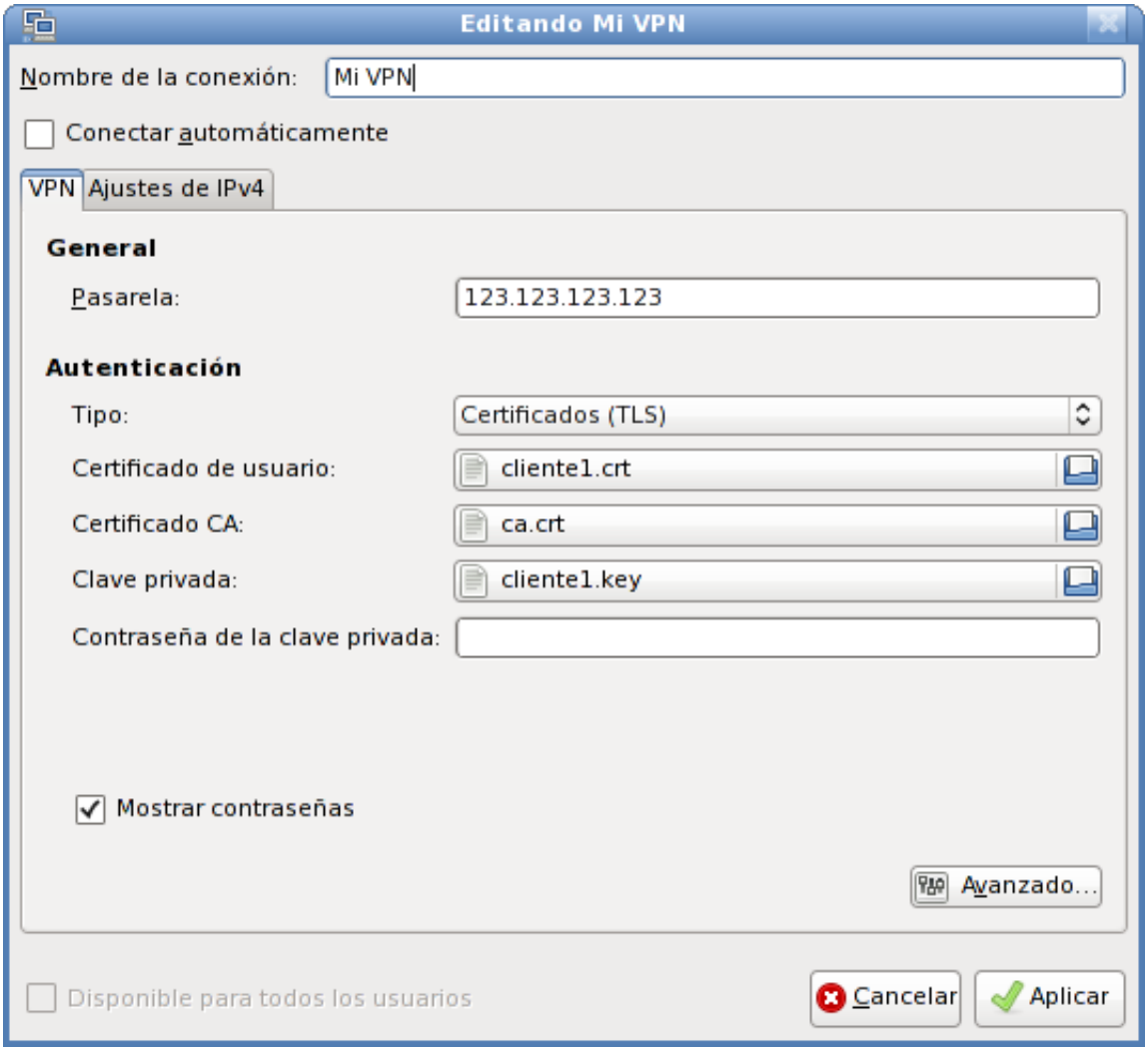
```
setsebool -P openvpn_enable_homedirs 1
```

Personalmente recomiendo crear una configuración nueva desde la interfaz de **NetworkManager**. Desde la ventana de redes VPN de la interfaz de **NetworkManager**, hacer clic en **Añadir**.

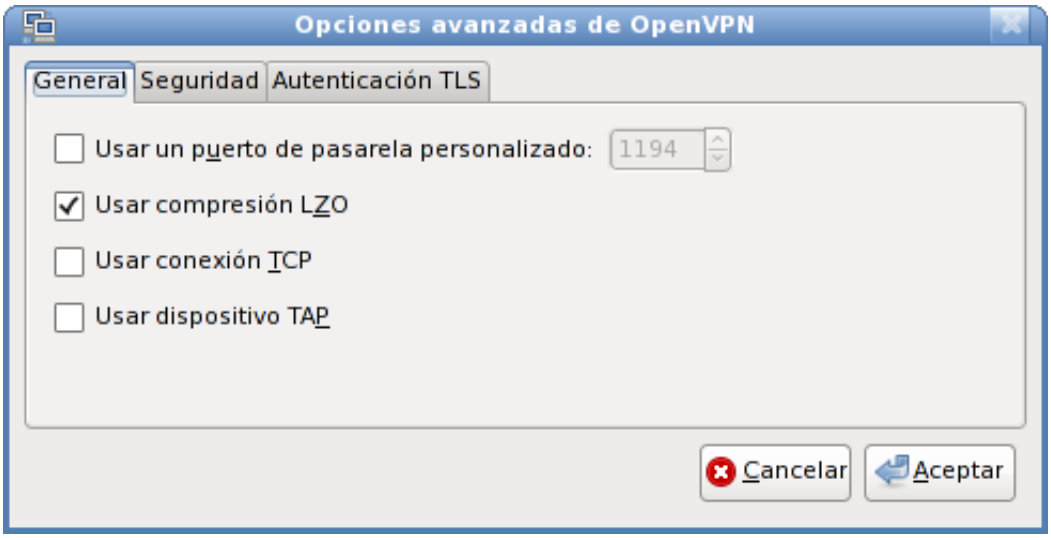


Aparecerá un diálogo donde se debe seleccionar que se trata de una **VPN** con **OpenVPN**.

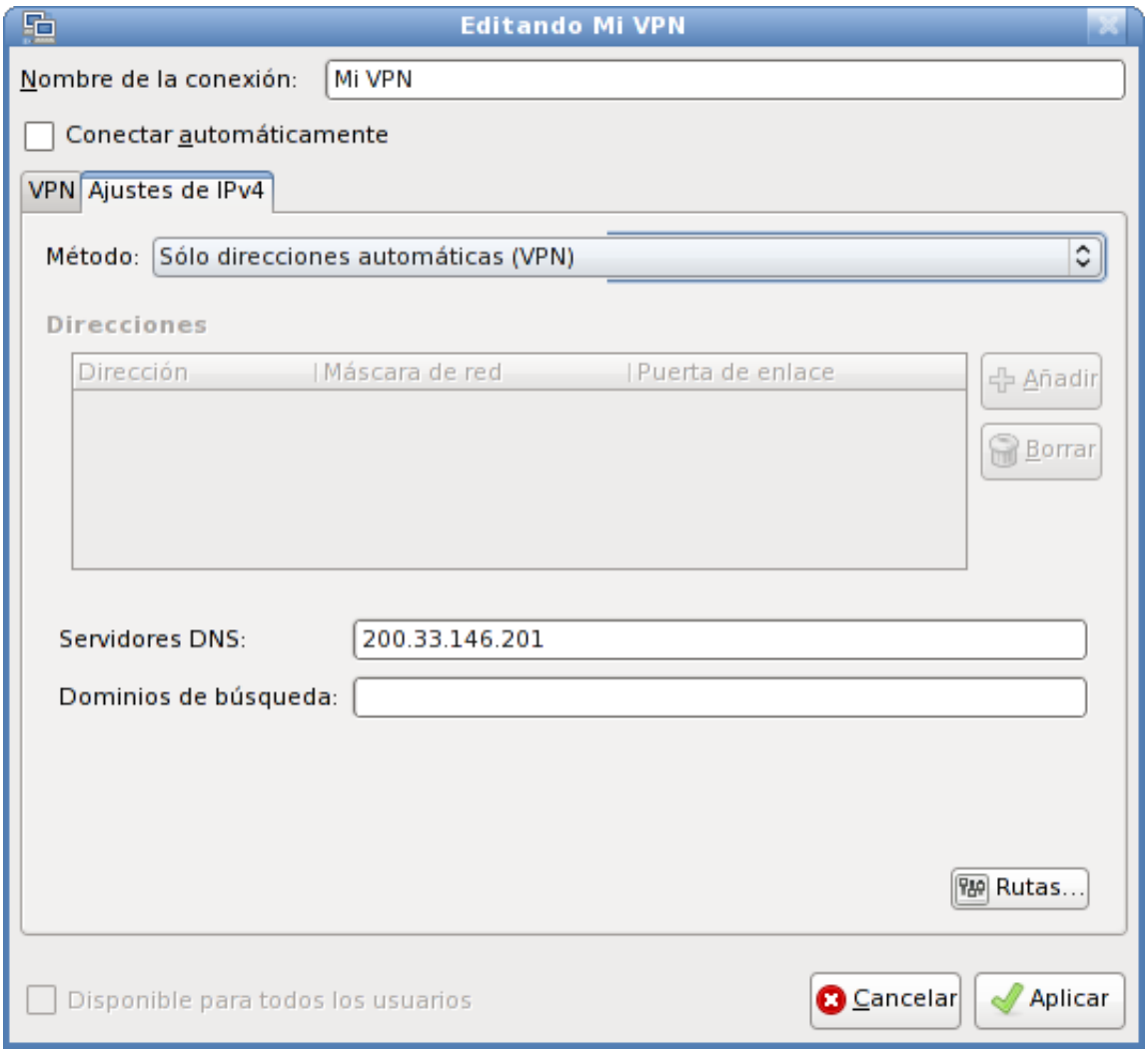
En la siguiente ventana de diálogo, se define el nombre de la conexión, dirección IP o nombre del servidor donde está instalado OpenVPN, y los certificados a utilizar. Si se siguieron los procedimientos de ese documento, se **deja en blanco** el campo **Contraseña de clave privada**.



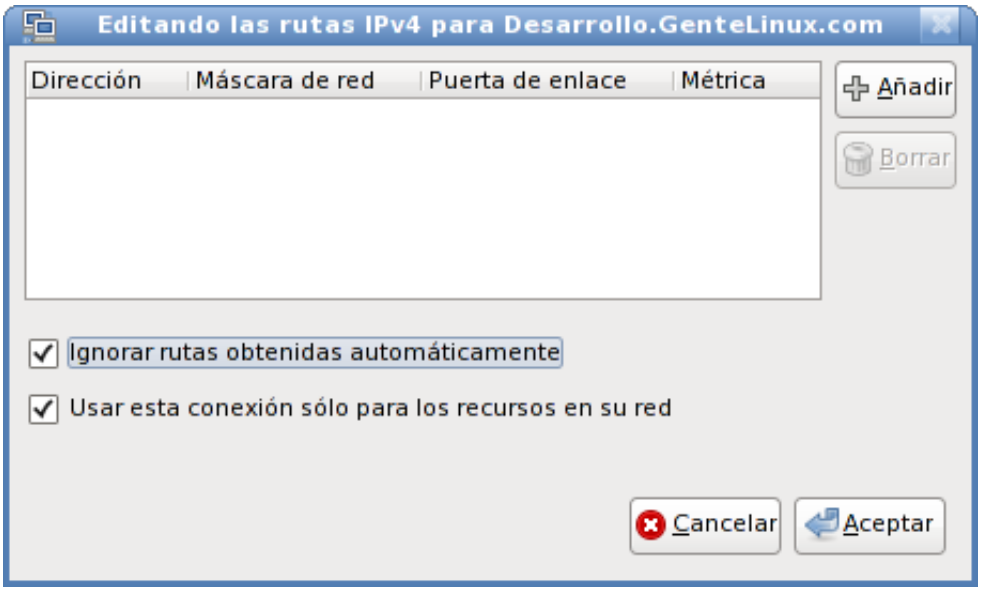
Luego, se hace clic en **Avanzado** para especificar que se utilizará compresión **LZO**.



Para evitar conflictos de conectividad, se hace clic en la pestaña **Ajustes IPV4**, y se define un servidor DNS que permita al cliente navegar a través de Internet y dentro de la red de la **VPN**.

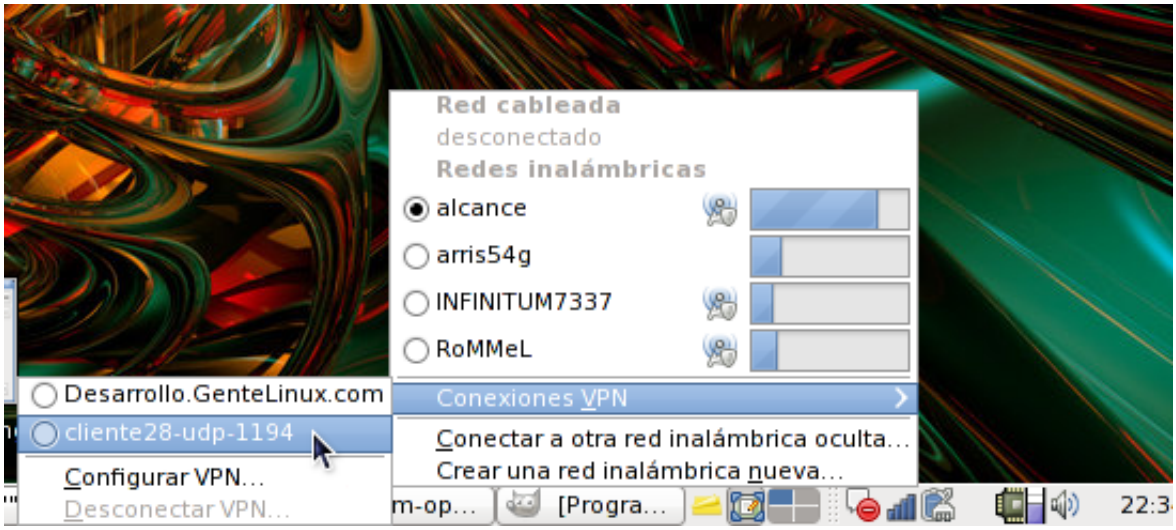


Se hace clic en **Rutas** para abrir otra ventana de diálogo y se seleccionan las casillas de las opciones **Ignorar las rutas obtenidas automáticamente** y **Usar esta conexión solo para los recursos de su red**. Opcionalmente se pueden añadir las rutas estáticas para tener conectividad con la red local detrás del servidor de **VPN**, tomando en cuenta que la red local desde la cual se está conectado el cliente debe ser diferente a la de la red local detrás del servidor de **VPN**, a fin de evitar conflictos de red.



Finalmente se hace clic en aplicar. Para conectarse a la red **VPN**, solo basta hacer clic sobre el icono de **NetworkManager** en el **Área de notificación** del panel de GNOME y seleccionar la

red **VPN** recién configurada.



Bibiliografía.

Este documento se basa sobre los manuales titulados VPN en servidor Linux y clientes Windows/Linux con OpenVPN + Shorewall [Parte 1] y VPN en servidor Linux y clientes Windows/Linux con OpenVPN + Shorewall [Parte 2], por **William López Jiménez**, publicados en **Alcance Libre**, cumpliendo cabalmente con los términos de la licencia **Creative Commons** Reconocimiento-NoComercial-CompartirIgual 2.1.

Alcance Libre
http://www.alcancelibre.org/staticpages/index.php/como-openvpn-server-centos5

()