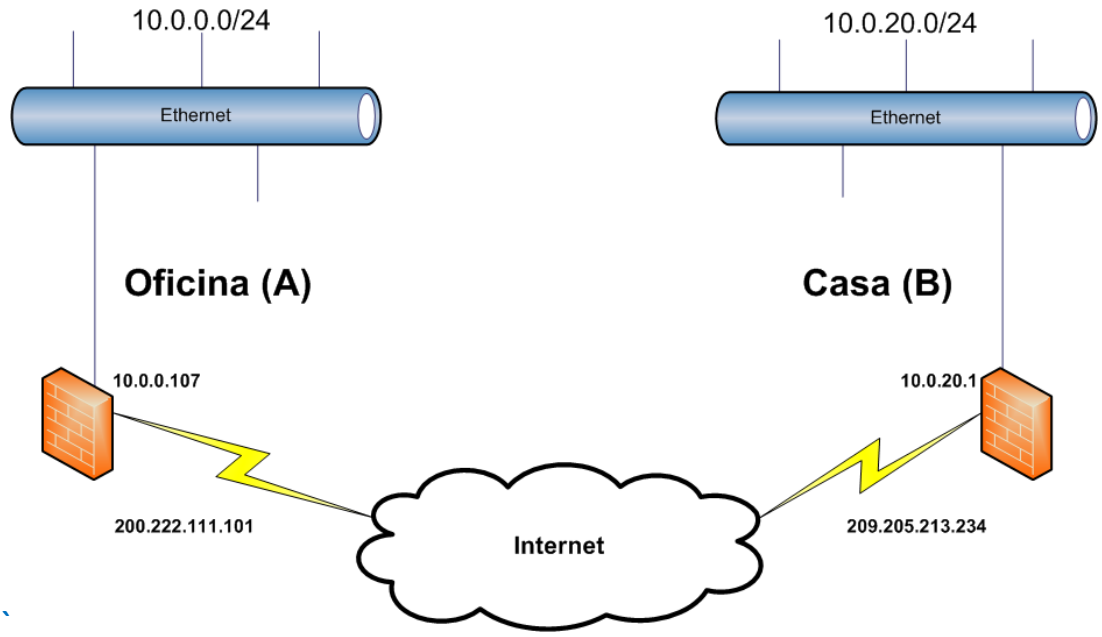
Conexión VPN

OpenVPN es una implementacion de VPN SSL la cual usa las extenciones OSI layer 2 o 3 para asegurar redes la cual usa los protocolos SSL/TLS, soporta diferentes medios de autenticacion como certificados, smart cards, y/o usuarios/contraseñas, y permite politicas de control de acceso para usaurios o grupos usando reglas de firewall aplicadas a las interfaces virtuales de la VPN. OpenVPN 2.0 permite multiples clientes conectar a un solo servidor (proceso) OpenVPN sobre un simple puerto TCP o UDP.

El proposito de la VPN es unir dos subredes en dos localizacioes, una la red de la oficina y la subred de mi red casera.



Configurando tu propia Autoridad Certificadora (CA - Certificate Authority) y generacion de certificados y par de llaves para el Servidor OpenVPN y un cliente VPN.

El primer paso al construir una VPN con OpenVPN 2.0 es establecer una PKI (Infraestructura de LLave Publica - Public Key Infrastructure), esta PKI consiste de:

- Un certificado aparte (tambien conocido como llave publica) y una llave privada para el servidor y cada cliente.
- Un Certificado Mastro para la Autoridad Certificadora (CA) y su llave la cual es usada para firmar cada certificado de el servidor y el cliente.

Generar la llave y el certificado Maestro para la Autoridad Certificadora (CA).

En esta seccion se generaran los certificados/llaves para la CA, el server y el cliente. Para la administracion de la PKI usaremos los scripts que vienen con OpenVPN (easy-rsa) pero en este caso usaremos la nueva version que tiene muchas mejoras, es esta easy-rsa 2.0.

Estos scripts de la version 2.0 de easy-rsa estan en: `/usr/doc/openvpn-2.0.6/easy-rsa/2.0/`

Se recomienda copiar el contenido de dicho directorio por ejemplo a `/etc/openvpn/easy-rsa-V2.0`.

Entonces haremos:

```
root@javier3:~# cd /etc/openvpn
```

```
root@javier3:/etc/openvpn# mkdir easy-rsa-V2.0
```

```
root@javier3:/etc/openvpn# cp -r /usr/doc/openvpn-2.0.6/easy-rsa/2.0/* /etc/openvpn/easy-rsa-V2.0
```

```
root@javier3:/etc/openvpn# cd /etc/openvpn/easy-rsa-V2.0
```

Ahora editaremos el archivo `vars` lo primero que se hara es definir la ruta para la variable `KEY_DIR` que por default estara asi: `/etc/openvpn/easy-rsa-V2.0/keys`, pero dicho directorio no existe por lo que primero lo crearemos:

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# mkdir -p /etc/openvpn/easy-rsa-V2.0/keys
```

Es en este directorio donde se almacenaran las llaves privadas, los archivos de requerimiento de certificado (`.csr`) y los certificados (`.crt`) y otros archivos como el serial y el index.txt.

Ahora configuraremos los parametros *`KEY_COUNTRY`*, *`KEY_PROVINCE`*, *`KEY_CITY`*, *`KEY_ORG`* y *`KEY_MAIL`*, no hay que dejar ninguno de estos parametros vacios, los valores de estas variables serán pasadas de manera determinada a los certificados que crearemos, por ejemplo:

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# vim vars
```

```
# These are the default values for fields
# which will be placed in the certificate.
# Don't leave any of these fields blank.
export KEY_COUNTRY="US"
export KEY_PROVINCE="CA"
export KEY_CITY="SanFrancisco"
export KEY_ORG="Fort-Funston"
export KEY_EMAIL="me@myhost.mydomain"
"vars" 64L, 1602C
```

Lo siguiente es inicializar la PKI, asi:

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# source ./vars
NOTE: If you run ./clean-all, I will be doing a rm -rf on /etc/openvpn/easy-rsa-V2.0/keys
```

Si se editaron los parametros correctamente veras algo como lo que salio arriba.

Ahora configuraremos un entorno nuevo.

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# ./clean-all
```

Conforme vayas creando certificados, keys, y requerimientos para firma de certificados, tendras que entender que solo los archivos.key deben de mantenerse confidenciales. Los archivos `.crt` y `.csr` pueden ser enviados sobre un canal inseguro como un email en texto plano.

- *Generando Parametros Diffie Hellman.*

Los parametros Diffie Hellman deben de ser generados para el Servidor OpenVPN:

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# ./build-dh
Generating DH parameters, 1024 bit long safe prime, generator 2
This is going to take a long time
.....
.....+.....+.....
.....+.....
.....+.....+.....
.....+.....+.....+
+*++*++*
root@javier3:/etc/openvpn/easy-rsa-V2.0#
```

Construiremos el certificado/key para la CA: Veremos algo asi:

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# ./pkitool --initca
Using CA Common Name: Fort-Funston CA
Generating a 1024 bit RSA private key
.....++++++
.....++++++
writing new private key to 'ca.key'
-----
```

Generacion de certificado y llaves para el servidor.

Lo siguiente es generar el certiicado y la llave privada par el servidor:

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# ./pkitool --server servidor
Generating a 1024 bit RSA private key
.....++++++
.....++++++
writing new private key to 'servidor.key'
-----
Using configuration from /etc/openvpn/easy-rsa-V2.0/openssl.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'US'
stateOrProvinceName   :PRINTABLE:'CA'
localityName          :PRINTABLE:'SanFrancisco'
organizationName      :PRINTABLE:'Fort-Funston'
commonName            :PRINTABLE:'servidor'
emailAddress          :IA5STRING:'me@myhost.mydomain'
Certificate is to be certified until Apr 13 15:36:00 2023 GMT (3650 days)
```

Generación de certificado y llave privada para un cliente.

Esto es muy similar a los pasos previos

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# ./pkitool cliente1
Generating a 1024 bit RSA private key
..............++++++
..++++++
writing new private key to 'cliente1.key'
-----
Using configuration from /etc/openvpn/easy-rsa-V2.0/openssl.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'US'
stateOrProvinceName   :PRINTABLE:'CA'
localityName          :PRINTABLE:'SanFrancisco'
organizationName      :PRINTABLE:'Fort-Funston'
commonName            :PRINTABLE:'cliente1'
emailAddress          :IA5STRING:'me@myhost.mydomain'
Certificate is to be certified until Apr 13 15:38:00 2023 GMT (3650 days)
```

Como pudimos ver lo todos los valores fueron tomados de el archivo `vars` y le agrego el valor de **commonName** el valor de el argumento que pusimos: ./pkitool --server **cliente1**, en este caso le puso **cliente1**.

Ahora crearemos un segundo certiicado para un nuevo cliente:

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# source ./vars
NOTE: If you run ./clean-all, I will be doing a rm -rf on /etc/openvpn/easy-rsa-V2.0/keys
```

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# ./pkitool cliente2
Generating a 1024 bit RSA private key
..............................++++++
..............................++++++
writing new private key to 'cliente2.key'
-----
Using configuration from /etc/openvpn/easy-rsa-V2.0/openssl.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'US'
stateOrProvinceName   :PRINTABLE:'CA'
localityName          :PRINTABLE:'SanFrancisco'
organizationName      :PRINTABLE:'Fort-Funston'
commonName            :PRINTABLE:'cliente2'
emailAddress          :IA5STRING:'me@myhost.mydomain'
Certificate is to be certified until Apr 13 15:40:00 2023 GMT (3650 days)
```

Conforme vayas agregando clientes lo haras con esta misma herramienta (pkitool) no hay que olvidar que cada vez que se vaya a usar el script pkitool se tiene que ejecutar el comando **sounce ./vars** antes de crear, o revocar algun certificado.

Archivos claves.

Ahora podremos encontrar nuestros nuevos certificados y llaves en el subdirectorio `keys` , esta es una explicacion de los archivos relevantes:

Nombre de Archivo	Necesario para	Proposito	Secreto
ca.crt	servidor + todos los clientes	Certificado para Root CA	NO
ca.key	Solo maquina con la llave para firmar	Llave para Root CA	**SI**
dh1024.pem	Solo Servidor	Parametros Diffie Hellman	NO
servidor.crt	Solo Servidor	Certificado para Servidor	NO
servidor.key	Solo Servidor	Llave privada para Servidor	**SI**
cliente1.crt	Solo cliente1	Certificado para Cliente1	NO
cliente1.key	Solo cliente1	Llave privada para Cliente1	**SI**
cliente2.crt	Solo cliente2	Certificado para Cliente2	NO
cliente2.key	Solo cliente2	Llave priada para Cliente2	**SI**

Creando archivos de configuracion para el servidor.

Consiguendo los archivos de configuracion de ejemplo.

Es recomendable usar los archivos de configuracion de ejemplo de OpenVPN como un punto inicial para tu propia configuracion. estos pueden ser encontrados en: `/usr/doc/openvpn-2.0.6/sample-config-files/` el archivo que necesitaremos es: `server.conf`

Editando el archivo de configuracion de el servidor.

El archivo de configuracion de ejemplo para el servidor es un punto de inicio ideal para la configuracion de un servidor OpenVPN. Creara una VPN usando una interfaz de red virtual TUN (para routed mode), escuchara conexiones de clientes en el puerto UDP 1194 (El numero de puerto oficial de OpenVPN), y distribuira direcciones virtuales de la subred 10.8.0.0/24 para los clientes que se conecten.

Copiamos el archivo de configuracion de el servidor:

```
root@javier3:/etc/openvpn/easy-rsa-V2.0# cd /etc/openvpn/
root@javier3:/etc/openvpn#
```

```
root@javier3:/etc/openvpn# cp /usr/doc/openvpn-2.0.9/sample-config-files/server.conf .
```

Editar el archivo `server.conf` y cambiar los valores de las lineas de los parametros: **ca**, **cert**, **key** y **dh** para que apunten a los archivos generados en la seccion anterior.

Por ejemplo quedaria asi:

```
root@javier3:/etc/openvpn# vim server.conf
```

```
# Any X509 key management system can be used.
# OpenVPN can also use a PKCS #12 formatted key file
# (see "pkcs12" directive in man page).
ca /etc/openvpn/easy-rsa-V2.0/keys/ca.crt
cert /etc/openvpn/easy-rsa-V2.0/keys/servidor.crt
key /etc/openvpn/easy-rsa-V2.0/keys/servidor.key  #

# Diffie hellman parameters.
# Generate your own with:
#   openssl dhparam -out dh1024.pem 1024
# Substitute 2048 for 1024 if you are using
# 2048 bit keys.
dh /etc/openvpn/easy-rsa-V2.0/keys/dh1024.pem

# Configure server mode and supply a VPN subnet
# for OpenVPN to draw client addresses from.
# The server will take 10.8.0.1 for itself,
```

Inicialización de la VPN y pruebas iniciales de conectividad.

Iniciando el Servidor.

Primero hay que asegurarse que el servidor OpenVPN es accesible desde el Internet, esto quiere decir:

- Abrir el puerto UDP 1194 en el firewall o configurar una regla de redireccionamiento de puerto (port forwarding) de el puerto UDP 1194 desde el gateway/firewall a la maquina servidor OpenVPN.
- Lo siguiete es asegurarse que la interfaz TUN no esta firewalleada.

Por simplicidad y para hacer pruebas iniciales, es recomendable iniciar el servidor OpenVPN desde la linea de comando, en lugar de iniciarlo como un servicio (daemon).

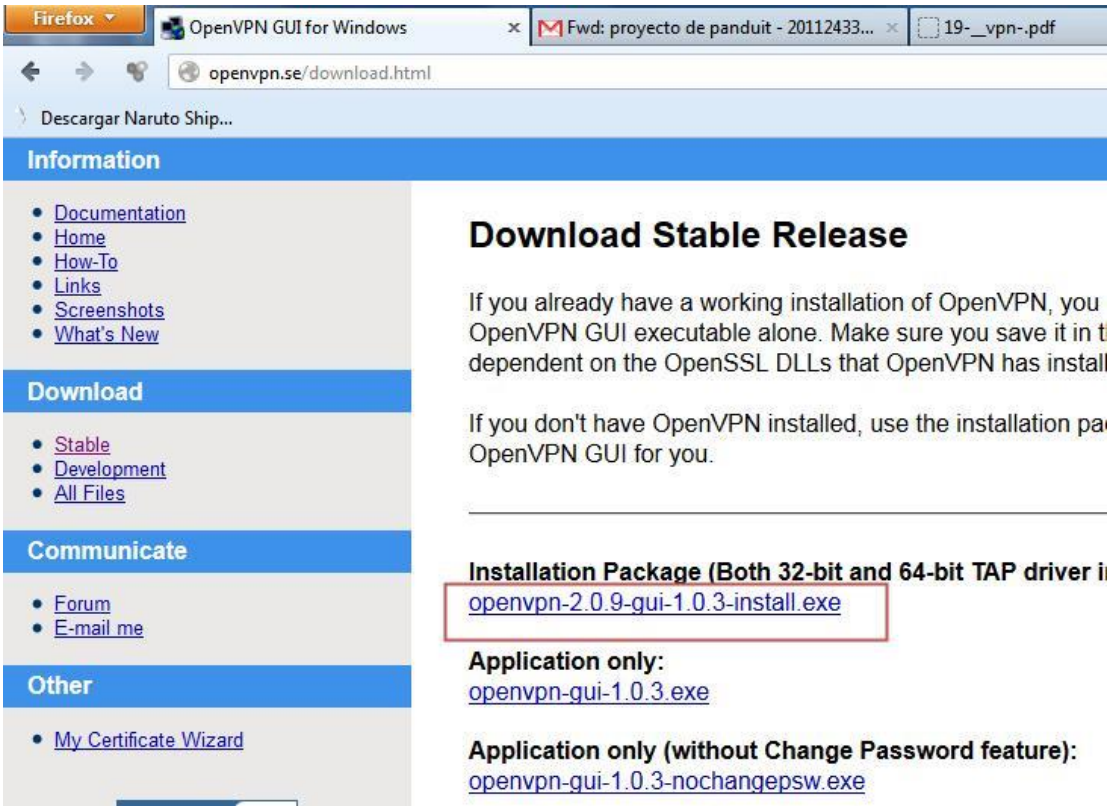
```
root@javier3:~# cd /etc/openvpn/
root@javier3:/etc/openvpn# openvpn server.conf
Mon Apr 15 22:12:05 2013 OpenVPN 2.0.9 i486-slackware-linux [SSL] [LZO] [EPOLL]
built on Jun 11 2007
Mon Apr 15 22:12:05 2013 Diffie-Hellman initialized with 1024 bit key
Mon Apr 15 22:12:05 2013 TLS-Auth MTU parms [ L:1542 D:138 EF:38 EB:0 ET:0 EL:0
]
Mon Apr 15 22:12:05 2013 TUN/TAP device tun0 opened
Mon Apr 15 22:12:05 2013 /sbin/ifconfig tun0 10.8.0.1 pointopoint 10.8.0.2 mtu 1
500
Mon Apr 15 22:12:05 2013 /sbin/route add -net 10.8.0.0 netmask 255.255.255.0 gw
10.8.0.2
Mon Apr 15 22:12:05 2013 Data Channel MTU parms [ L:1542 D:1450 EF:42 EB:135 ET:
0 EL:0 AF:3/1 ]
Mon Apr 15 22:12:05 2013 UDPv4 link local (bound): [undef]:1194
Mon Apr 15 22:12:05 2013 UDPv4 link remote: [undef]
Mon Apr 15 22:12:05 2013 MULTI: multi_init called, r=256 v=256
Mon Apr 15 22:12:05 2013 IFCONFIG POOL: base=10.8.0.4 size=62
Mon Apr 15 22:12:05 2013 IFCONFIG POOL LIST
Mon Apr 15 22:12:05 2013 cliente1,10.8.0.4
Mon Apr 15 22:12:05 2013 Initialization Sequence Completed
```

Configurar el Cliente en Windows XP

En el caso de Windows XP se debe instalar openVPN-GUI

Entonces abrimos nuestro explorador y entramos al sitio web de openvpn (http://openvpn.se).

Hacer click en el link de descarga del archivo:

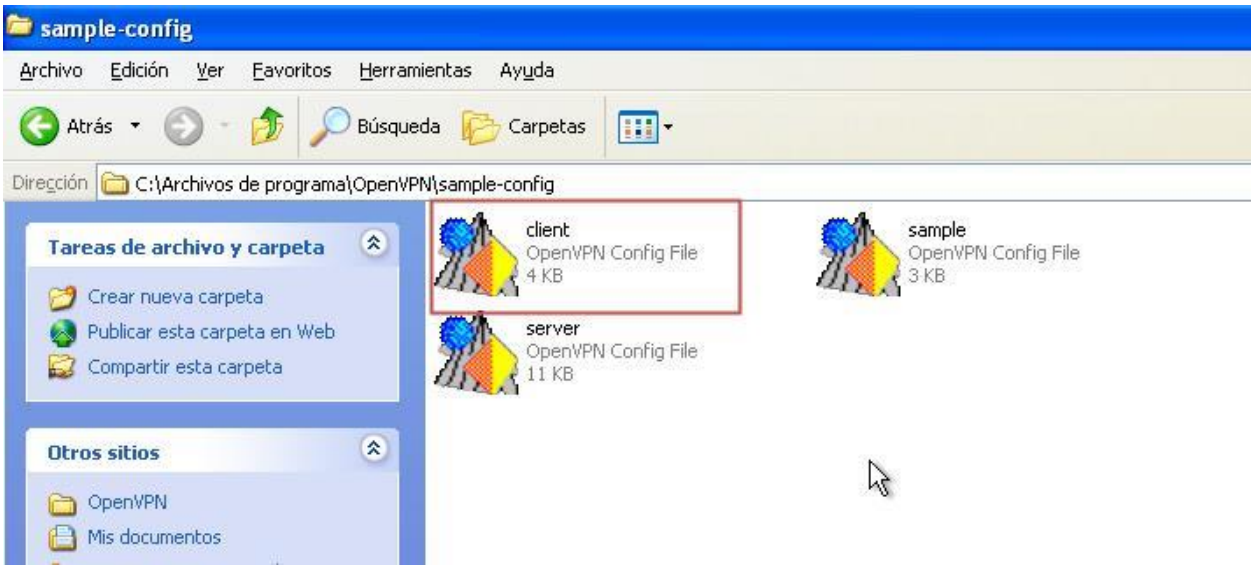


Una vez instalados copiamos los archivos entregados por el servidor (ca.crt, cliente1.crt, cliente1.key y todos los demás archivos) a un directorio en Windows en específico, por ejemplo:

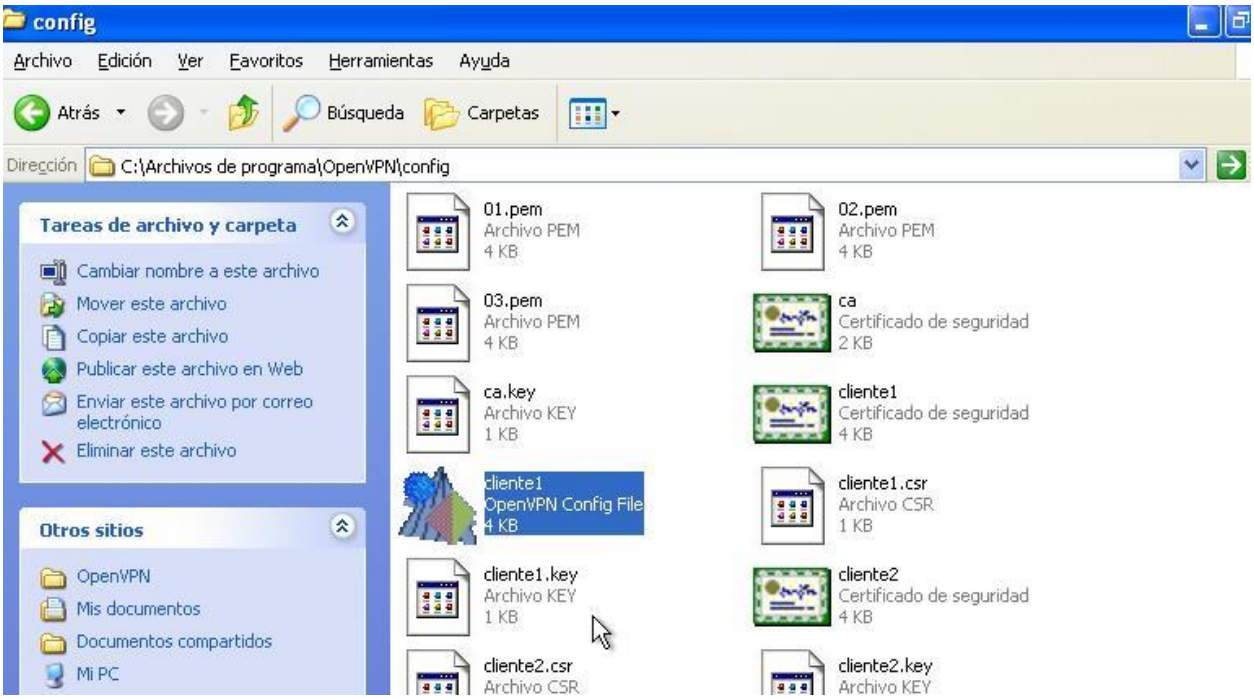
C:\Archivos de programa\OpenVPN\config

Una vez en el directorio C: \Archivos de programa\OpenVPN\sample-config y copiamos el archivo client a C:\Archivos de programa\OpenVPN\config.

.



Una vez en C:\Archivos de programa\OpenVPN\config le cambiamos el nombre de client por cliente1 lo abrimos con un editor de texto (notepad sirve) y apuntamos la configuración a la dirección en donde pusimos los certificados y la llave privada.



.

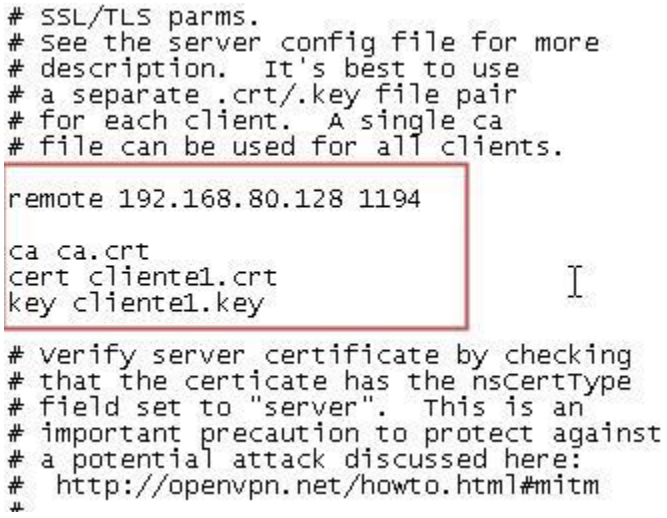
Ubicamos en el documento donde están estos archivos:

ca ca.crt
cert client1.crt
key client1.key

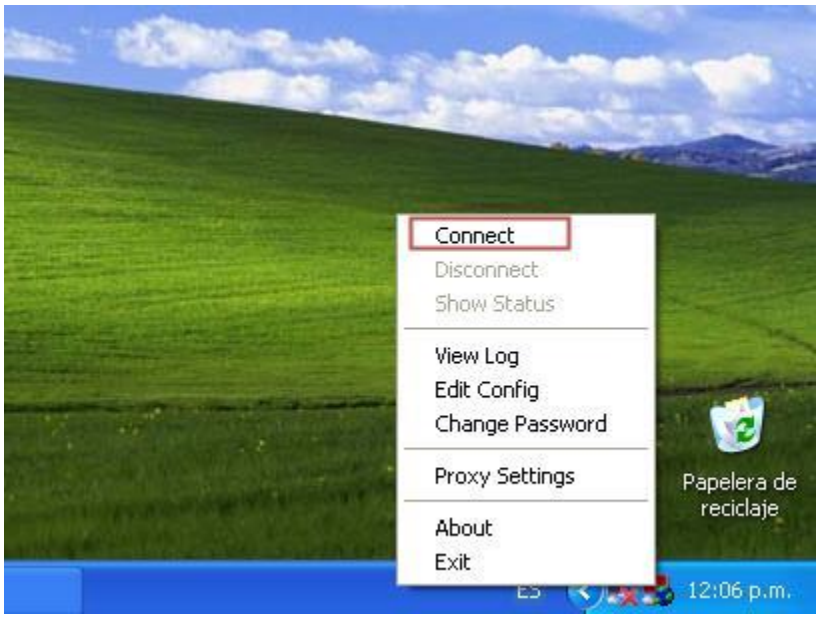
Luego modificamos estas líneas de manera que quede así:

ca ca.crt
cert cliente1.crt
key cliente1.key

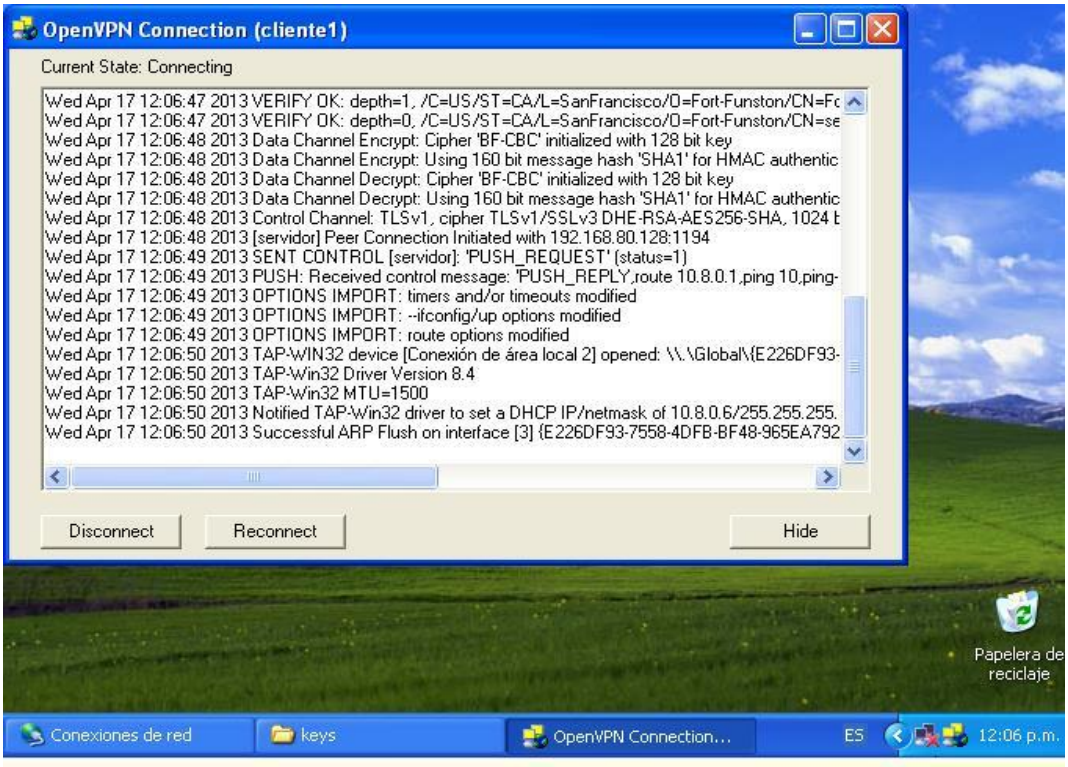
remote 192.168.80.128 1194



Veremos un icono de dos computadores rojos en la bandeja del sistema, si hacemos click derecho veremos un menú en donde la primera opción es connect.



Al pulsar conectar aparecerá una consola mostrando algunos datos y al finalizar aparecerá un mensaje avisando que la conexión se ha realizado:




cliente1 is now connected.
Assigned IP: 10.8.0.6
apelera de
reciclaje
ES
12:07 p.m.

: