Réseau Privé Virtuel.

Présentation :

Un Réseau privé virtuel appelé VPN est un système permettant de créer un lien direct entre des ordinateurs distants. On utilise notamment ce terme dans le travail à distance, ainsi que pour l'accès à des structures de type cloud computing. Un VPN permet d'accéder à des ordinateurs distants comme si l'on était connecté au réseau local. On peut ainsi avoir un accès au réseau interne (réseau d'entreprise, par exemple).

Un VPN dispose généralement aussi d'une passerelle permettant d'accéder à l'extérieur, ce qui permet de changer l'adresse IP source apparente de ses connexions. Cela rend plus difficile l'identification et la localisation approximative de l'ordinateur émetteur par le fournisseur de service.

Objectif :

Installer et configurer un VPN avec le logiciel libre OpenVPN.

Pré requis :

- ➔ Deux ordinateurs sur Linux (ici, on utilisera la Debian 8.2) et un ordinateur sur Windows.
- ➔ Avoir une connexion internet
- ➔ Avoir une IP fixe pour le serveur
- ➔ Mon serveur s'appelle squid et son @IP est 192.168.1.137/24

Sommaire :

I. Installation d'OpenVPN

Avant l'installation, mettre à jour les paquets :

```
root@serveurvpn:~# apt-get update
```

Puis installer les paquets :

```
root@serveurvpn:~# apt-get install openvpn openssh-server openssl
```

OpenVPN utilise les protocoles TLS et SSL et écoute sur les ports UDP ou TCP.

II. Construction d'une PKI

On va créer deux répertoires et copier les scripts dans ce repértoire :

```
root@serveurvpn:~# mkdir /etc/openvpn/easy-rsa
root@serveurvpn:~# cp /usr/share/easy-rsa/* /etc/openvpn/easy-rsa/
root@serveurvpn:~# mkdir /etc/openvpn/easy-rsa/keys
```

On va se situer dans le répertoire où il y a les scripts :

```
root@serveurvpn:~# cd /etc/openvpn/easy-rsa/
root@serveurvpn:/etc/openvpn/easy-rsa# ls
build-ca           build-key-server  list-crl           sign-req
build-dh           build-req         openssl-0.9.6.cnf  vars
build-inter        build-req-pass    openssl-0.9.8.cnf  whichopensslcnf
build-key          clean-all         openssl-1.0.0.cnf
build-key-pass     inherit-inter     pkitool
build-key-pkcs12   keys              revoke-full
```

On va éditer le fichier vars et modifier ces valeurs :

```
  GNU nano 2.2.6                Fichier : vars                              Modifié

export CA_EXPIRE=3650

# In how many days should certificates expire?
export KEY_EXPIRE=3650

# 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="FR"
export KEY_PROVINCE="France"
export KEY_CITY="Caen"
export KEY_ORG="BTSSIO"
export KEY_EMAIL="root@sio.local"
export KEY_OU="MyOrganizationalUnit"
```

On initialise les variables :

```
root@serveurvpn:/etc/openvpn/easy-rsa# source ./vars
```

On tape cette commande :

```
root@serveurvpn:/etc/openvpn/easy-rsa# ./build-ca
```

Maintenant, on va créer le certificat du serveur, 3 commandes à faire :

- `root@serveurvpn:/etc/openvpn/easy-rsa# touch keys/index.txt`
- `root@serveurvpn:/etc/openvpn/easy-rsa# echo 01 > keys/serial`
- `root@serveurvpn:/etc/openvpn/easy-rsa# chmod -R 0700 keys`

On peut lancer la commande maintenant :

```
root@serveurvpn:/etc/openvpn/easy-rsa# ./build-key-server serveurvpn
```

Important, répondre yes (y) aux deux questions qui suivent :

```
Certificate is to be certified until Nov 14 14:37:35 2025 GMT (3650 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

On fait le certificat du client maintenant :

```
root@serveurvpn:/etc/openvpn/easy-rsa# ./build-key client1
```

NB : sur le client même, il lui faut le ca.cert.

Enfin, on va générer les paramètres Diffie Hellman. Cela dure un assez long moment et le fichier crée est nommé dh2048.pem dans le sous répertoire keys :

```
root@serveurvpn:/etc/openvpn/easy-rsa# ./build-dh
```

III. Configuration du serveur

On va créer un utilisateur spécial openvpn et son groupe sans répertoire ni shell :

```
root@serveurvpn:/# groupadd openvpn
root@serveurvpn:/# useradd -d /dev/null -g openvpn -s /bin/false openvpn
```

Puis, on va récupérer le fichier de conf du serveur :

```
root@serveurvpn:/etc/openvpn# cp /usr/share/doc/openvpn/examples/sample-config-f
iles/server.conf.gz /etc/openvpn/
root@serveurvpn:/etc/openvpn# gunzip server.conf.gz
```

Éditez-le et modifiez-le :

```
# 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 ca.crt
cert serveurvpn.crt
key servervpn.key  # This file should be kept secret

# 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 dh2048.pem
```

```
# Push routes to the client to allow it
# to reach other private subnets behind
# the server.  Remember that these
# private subnets will also need
# to know to route the OpenVPN client
# address pool (10.8.0.0/255.255.255.0)
# back to the OpenVPN server.
push "route 192.168.1.0 255.255.255.0"
```

```
# Select a cryptographic cipher.
# This config item must be copied to
# the client config file as well.
cipher BF-CBC        # Blowfish (default)
;cipher AES-128-CBC   # AES
```

```
# You can uncomment this out on
# non-Windows systems.
user openvpn
group openvpn
```

```
# Silence repeating messages.  At most 20
# sequential messages of the same message
# category will be output to the log.
mute 20
```

IV. Configuration des clients Linux et Windows.

Linux :

Sur un client Linux, on récupére le client.conf :

```
root@serveurvpn:/etc/openvpn# cp /usr/share/doc/openvpn/examples/sample-config-f
iles/client.conf /etc/openvpn/ _
```

Puis on le modifie suivant notre configuration :

```
# The hostname/IP and port of the server.
# You can have multiple remote entries
# to load balance between the servers.
remote 192.168.1.3 1194
;remote my-server-2 1194
```

```
# SSL/TLS parms.
# See the server config file for more
# description.  It's best to use
# a separate .crt/.key file pair
# for each client.  A single ca
# file can be used for all clients.
ca ca.crt
cert client1.crt
key client1.key
```

```
# Select a cryptographic cipher.
# If the cipher option is used on the server
# then you must also specify it here.
cipher BF-CBC
```

```
# Silence repeating messages
mute 20
```

Ainsi, nous devons avoir trois documents :

```
root@debian8:/etc/openvpn# ls
ca.crt  client1.key  client.conf  update-resolv-conf
```

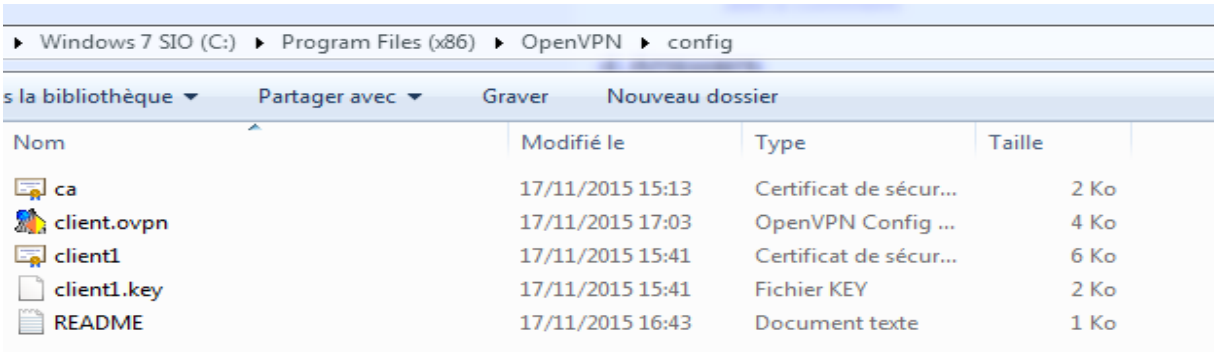
Windows :

Installer le client VPN windows via le site : openvpn-2.0.9-gui-1.0.3-install.exe

De plus, on copie le fichier de conf par défaut présent dans :

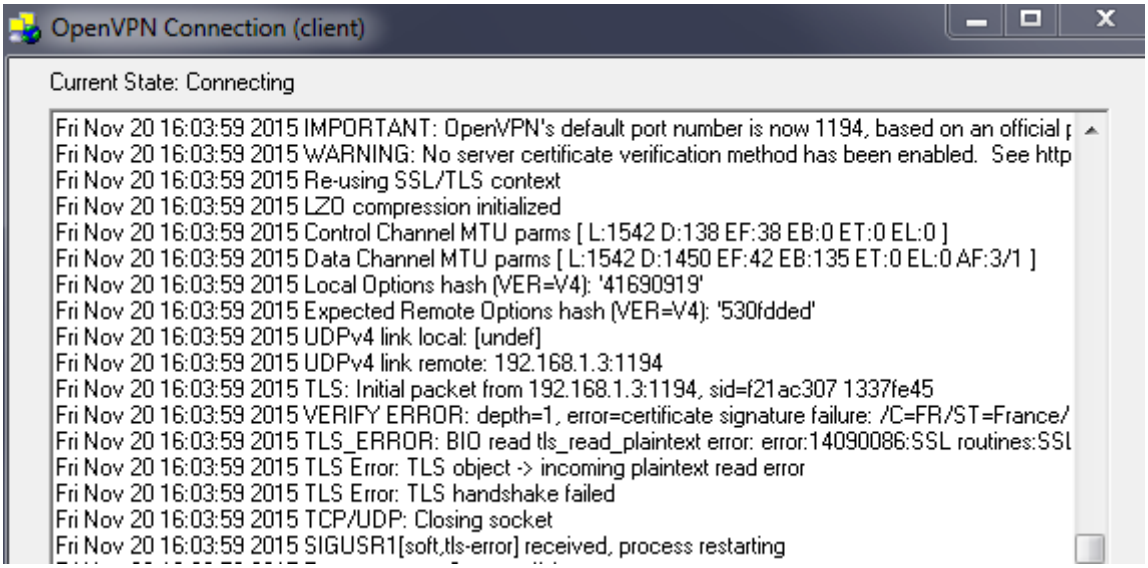
C:\Program Files\OpenVPN\Sample-config\clientopvn dans le sous repertoire config

Mettez dans le répertoire config le ca.crt, le client1.crt et le client1.key.



On démarre le serveur openvpn : service openvpn start

Sur le windows, on remarque avec un clique droit en bas à droit puis Connect :



Sur le serveur, le tun s'est créé :

Avec la commande ifconfig :

```
tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
-00
          inet adr:10.8.0.1  P-t-P:10.8.0.2  Masque:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:100
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Et le port d'écoute :

```
root@serveurvpn:~# lsof -i:1194
COMMAND  PID    USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
openvpn 1057 openvpn    5u  IPv4  13024      0t0  UDP *:openvpn
root@serveurvpn:~# _
```

Puis sur le client linux :

```
tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
-00
          inet adr:10.8.0.6  P-t-P:10.8.0.5  Masque:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:100
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```