



Installation OpenVPN 2.01 sur une Debian Sarge

But de ce document

Ce document me sert de **mémo** pour mettre en place le système **OpenVPN** sur un serveur Debian Sarge et sur des clients Windows ou Linux Debian.

Rubrique : [Installation](#)
Le : jeudi 23 mars 2006
Par : Tony
Visites : 3167

OpenVPN est un système permettant de relier des postes distants sur un réseau informatique en passant par Internet mais de manière sécurisée (Tunnel VPN).

Pré-requis

Avoir installé une Debian Sarge de base :

► http://www.coagul.org/article.php3?id_article=337

ATTENTION : Il est fortement conseillé de savoir ce qu'est un certificat et une clé de cryptage pour la configuration d'OpenSSL. Il est conseillé également de connaître le fonctionnement des tables de routage pour la configuration d'OpenVPN.

Installation d'OpenVPN 2.01 sur une Debian Sarge

Paquet un installer :

```
# apt-get install openvpn
```

Questions posées après l'installation du paquet :

Écran (Titre en rouge)	Question	Réponse
Configuration de openvpn	Faut-il créer un périphérique TUN/TAP ?	Oui
Configuration de openvpn	Faut-il arrêter openvpn avant de le mettre à jour ?	Non

Avec **OpenVPN**, il est possible de compresser les données passant dans le tunnel VPN pour limiter l'utilisation de la bande passante. Pour cela, il faut installer le paquet « liblz1 » :

```
# apt-get install liblz1
```

Installation OpenSSL

Pour sécuriser le tunnel VPN avec des clés et des certificats, OpenVPN, utilise le programme OpenSSL :

```
# apt-get install openssl
```

Création du certificat de l'autorité de certification (CA)

La partie la plus compliquée et la plus fastidieuse dans la mise en place d'un serveur OpenVPN concerne la génération des clés et des certificats. **OpenVPN** est livré avec plusieurs scripts permettant de générer plus facilement les clés et les certificats pour OpenSSL. Ces scripts sont enregistrés dans le dossier « **easy-rsa** » :

```
# cd /usr/share/doc/openvpn/examples/easy-rsa/
```

Avant d'utiliser les scripts, il faut éditer le fichier « **vars** » pour initialiser les variables par défaut indiquées à la fin de ce fichier. Par exemple :

- ▶ export KEY_COUNTRY=**FR**
- ▶ export KEY_PROVINCE=**France**
- ▶ export KEY_CITY=**Dijon**
- ▶ export KEY_ORG="**MonEntreprise**"
- ▶ export KEY_EMAIL="**contact@monentreprise.fr**"

Une fois le fichier modifié, la ligne suivante permet d'initialiser les variables pour les scripts :

```
# . ./vars
```

Le script suivant, permet de créer ou de réinitialiser le sous-dossier « **keys** » :

```
# ./clean-all
```

Le dossier « **easy-rsa** » contient un fichier de configuration par défaut d'OpenSSL, qu'il faut décompresser, pour qu'il soit utilisable pour faire fonctionner la génération des clés et des certificats :

```
# gunzip openssl.cnf.gz
```

Le script suivant permet de créer dans « **keys** » le certificat principal du serveur « **ca.crt** » et la clé correspondante « **ca.key** » :

```
# ./build-ca
```

Ce script doit afficher à l'écran quelque chose qui ressemble à ça :

```
# ./build-ca
```

```
Generating a 1024 bit RSA private key
.....++++++
.....++++++
writing new private key to 'ca.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.

What you are about to enter is what is called a Distinguished Name or a
DN.

There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

-----
Country Name (2 letter code) [FR]:
State or Province Name (full name) [France]:
Locality Name (eg, city) [Dijon]:
Organization Name (eg, company) [MonEntreprise]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:MonServeur
Email Address
[[contact@monentreprise.fr->mailto:contact@monentreprise.fr]]:
```

ATTENTION : Pour les questions, la plupart des champs sont renseignés par défaut sauf le « **Common Name** » qu'il faut renseigner manuellement. Exemple « **MonServeur** ».

Création du certificat et de la clé pour le serveur OpenVPN

Le script suivant permet de créer dans « **keys** » le certificat « **LeServeurVPN.crt** » et la clé « **LeServeurVPN.key** » pour le serveur VPN nommé par exemple « **LeServeurVPN** » :

```
# ./build-key-server LeServeurVPN
```

Ce script doit afficher à l'écran quelque chose qui ressemble à ça :

```
# ./build-key-server MonServeur
```

```
Generating a 1024 bit RSA private key
.....
..+++++
...+++++
writing new private key to 'MonServeur.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a
DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [FR]:
State or Province Name (full name) [France]:
Locality Name (eg, city) [Dijon]:
Organization Name (eg, company) [MonEntreprise]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:LeServeurVPN
Email Address
[[tony.galmiche@plastigray.com->mailto:tony.galmiche@plastigray.com]]:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Using configuration from
/usr/share/doc/openvpn/examples/easy-rsa/openssl.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
```

```

countryName          :PRINTABLE:'FR'
stateOrProvinceName :PRINTABLE:'France'
localityName         :PRINTABLE:'Dijon'
organizationName     :PRINTABLE:'MonEntreprise'
commonName           :PRINTABLE:'MonServeurVPN'
emailAddress

:IA5STRING:['contact@monentreprise.fr->mailto:'contact@monentreprise.fr]
Certificate is to be certified until Dec  7 13:41:02 2015 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

```

ATTENTION : Pour les questions, tous les champs sont renseignés par défaut sauf le « **Common Name** » qu'il faut renseigner manuellement. Exemple « **MonServeurVPN** ». Personnellement, je n'ai pas renseigné le champ « password »

Création du certificat et de la clé pour un client OpenVPN

Le script suivant permet de créer dans « **keys** » le certificat « **Client01.crt** » et la clé « **Client01.key** » pour le client VPN nommé par exemple « **Client01** » :

```
# ./build-key Client01
```

Ce script doit afficher à l'écran quelque chose qui ressemble à ça :

```

pgdebian:/usr/share/doc/openvpn/examples/easy-rsa# ./build-key Client01
Generating a 1024 bit RSA private key
..+++++
.....+++++
writing new private key to 'Client01.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a
DN.

```

```
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [FR]:
State or Province Name (full name) [France]:
Locality Name (eg, city) [Dijon]:
Organization Name (eg, company) [MonEntreprise]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:Client01
Email Address [contact@monentreprise.fr]:
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Using configuration from
/usr/share/doc/openvpn/examples/easy-sa/openssl.cnf
DEBUG[load_index]: unique_subject = "yes"
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName             :PRINTABLE:'FR'
stateOrProvinceName     :PRINTABLE:'France'
localityName            :PRINTABLE:'Dijon'
organizationName        :PRINTABLE:'MonEntreprise'
commonName               :PRINTABLE:'Client01'
emailAddress
:IA5STRING:['contact@monentreprise.fr->mailto:'contact@monentreprise.fr']
Certificate is to be certified until Mar  4 09:19:09 2016 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
```

ATTENTION : Il faudra renouveler cette opération pour chaque client. Pour les questions, tous les champs sont renseignés par défaut sauf le « **Common Name** » qu'il faut renseigner manuellement. Exemple « **Client01** ». Chaque « **Common Name** » de chaque client doit être différent. Personnellement, je n'ai pas renseigné le champ « **password** »

Création du paramètre Diffie Hellman

Le script suivant permet de créer dans « **keys** » le fichier « **dh1024.pem** » :

```
# ./build-dh
```

Ce script doit afficher à l'écran quelque chose qui ressemble à ça :

```
# ./build-dh
```

```
Generating DH parameters, 1024 bit long safe prime, generator 2
```

```
This is going to take a long time
.....
.....
.....+.
.+++++
```

Résumé des certificats et clés créés précédemment dans « **keys** »

Emplacement du fichier	Type	Nom	Secret
Serveur de certification (CA) OpenSSL	Certificat	ca.crt	Non
Serveur de certification (CA) OpenSSL	Clé	ca.key	Oui
Serveur OpenVPN	Certificat	LeServeurVPN.crt	Non
Serveur OpenVPN	Clé	LeServeurVPN.key	Oui
paramètre Diffie Hellman		dh1024.pem	Non
Client OpenVPN N°1	Certificat	Client01.crt	Non
Client OpenVPN N°1	Clé	Client01.key	Oui

Mise en place des certificats et des clés

concernant le serveur OpenVPN/ OpenSSL, le plus simple est de copier les 4 fichiers

dans le dossier `/etc/openvpn`

```
# cp ./keys/ca.crt /etc/openvpn/  
# cp ./keys/ca.key /etc/openvpn/  
# cp ./keys/LeServeurVPN.crt /etc/openvpn/  
# cp ./keys/LeServeurVPN.key /etc/openvpn/  
# cp ./keys/dh1024.pem /etc/openvpn/
```

Pour le client, il faudra copier ses deux fichiers une fois que celui-ci sera installé.

Création d'un utilisateur avec des droits limités pour OpenVPN

Pour limiter les risques d'attaques sur OpenVPN, il est important que le processus d'OpenVPN fonctionne sur un utilisateur n'ayant aucun droit sur le système.

Souvent, l'utilisateur « **nobody** » est utilisé par défaut, mais il est encore plus sécurisant de faire tourner chaque processus avec un utilisateur différent. Donc, pour le processus OpenVPN, nous allons créer l'utilisateur « **openvpn** » :

```
# groupadd openvpn  
# useradd -d /dev/null -g openvpn -s /bin/false openvpn
```

Configuration d'OpenVPN

Par défaut OpenVPN est fourni avec plusieurs fichiers d'exemples enregistrés dans le dossier :

► `/usr/share/doc/openvpn/examples/sample-config-files/`

Pour configurer le serveur, je suis parti du fichier d'exemple « **server.conf.gz** », qu'il faut donc décompresser et mettre en place dans « `/etc/openvpn` » :

```
# cd /usr/share/doc/openvpn/examples/sample-config-files/  
# gunzip server.conf.gz  
# cp server.conf /etc/openvpn/
```

Il suffit ensuite d'adapter ce fichier en fonction des besoins. Voici par exemple le fichier de configuration que j'utilise :

```
;Port en écoute utilisé pour la connexion VPN  
;port 1194
```

```
;Protocole utilisé (Le protocole udp est plus sécurisé que le tcp)
proto udp

;Type d'interface réseau virtuelle créée
dev tun

;Nom des fichiers servant à l'authentification des clients via OpenSSL
ca ca.crt
cert LeServeurVPN.crt
key LeServeurVPN.key
dh dh1024.pem

;Adresse du réseau virtuel (Le serveur aura l'adresse 10.8.0.1)
server 10.8.0.0 255.255.255.0

;Cette ligne ajoute sur le client la route du réseau du serveur
push "route 192.168.0.0 255.255.255.0"

;Ces lignes indiquent aux clients l'adresse des serveur DNS et WINS
push "dhcp-option DNS 192.168.0.2"
push "dhcp-option DOMAIN MonDomaine.com"
push "dhcp-option WINS 192.168.0.3"

# Cette ligne permet aux clients de voir les autres clients
;client-to-client

keepalive 10 120

;Cette ligne active la compression
comp-lzo

;Ces lignes indiquent un user et un group particulier pour le processus
```

```
user openvpn
group openvpn

;Ces lignes permettent de rendre persistante la connexion
persist-key
persist-tun

status openvpn-status.log

;Cette ligne permet d'indiquer le niveau de log souhaité (de 1 à 9)
verb 1
```

Démarrage du serveur OpenVPN

La commande suivante permet de démarrer ou redémarrer le serveur :

```
# /etc/init.d/openvpn restart
```

Ne pas hésiter à regarder dans les logs que tout c'est bien passé :

```
# tail -100 /var/log/syslog
```

Bien vérifier également que le processus tourne sous l'utilisateur « openvpn »

```
# ps aux | grep openvpn
```

Pour finir, si tout c'est bien passé l'interface « tun0 » doit apparaître dans la configuration du réseau :

```
# ifconfig
...
tun0      Lien encap:UNSPEC  HWaddr
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 il doit être possible de la pinguer :

```
# ping 10.8.0.1
```

Installation du Client Windows

Avec OpenVPN, la notion de client et de serveur n'existe pas étant donné que c'est le même logiciel qui peut faire office de client ou de serveur. Dans mon cas, le serveur est installé sur une Debian Sarge et les clients sont sous Debian Testing ou Windows 2000/XP.

Pour Windows, il existe une version d'OpenVPN avec une installation simplifiée téléchargeable à cette adresse :

- ▶ <http://openvpn.se/download.html>
- ▶ http://openvpn.se/files/install_packages/openvpn-2.0.5-gui-1.0.3-install.exe

Lors de l'installation de ce programme, la carte réseau virtuelle « TAP-Win32 Adapter V8 » est installée. Une fois le programme installé, il est lancé automatiquement et à chaque démarrage de Windows. Un icône pour le configurer est disponible à côté de l'heure.

Ensuite, il est possible de franciser OpenVPN en remplaçant le binaire enregistré dans « C:\Program Files\OpenVPN\bin\openvpn-gui.exe » par le binaire téléchargeable à l'adresse ci-dessous (Attention : Il faut le renommer après l'avoir téléchargé) :

- ▶ <http://openvpn.se/files/localized/binary/1.0.3/openvpn-gui-1.0.3-fr.exe>

Configuration du client Windows

La première chose à faire est de copier dans le dossier « C:\Program Files\OpenVPN\config » les fichiers servant à l'authentification du client via OpenSSL créés dans les chapitres précédents :

- ▶ ca.crt, Client01.crt et Client01.key

Ensuite, il faut modifier le fichier de configuration pour l'adapter à votre cas. Pour éditer le fichier, il est possible de faire un clic droit sur l'icône « OpenVPN » situé à gauche de l'heure et de choisir l'option « Éditer la configuration ».

Voici un exemple de fichier que j'utilise pour mes clients :

```
client
dev tun
proto udp

remote 192.0.1.2 1194
```

```
resolv-retry infinite
nobind

persist-key
persist-tun

ca ca.crt
cert Client01.crt
key Client01.key

comp-lzo

verb 1
```

Normalement, si vous utilisiez la même configuration que moi, vous n'aurez qu'à changer :

- ▶ L'adresse IP du serveur OpenVPN sur la ligne « remote »
- ▶ Le nom des fichiers : ca.crt, Client01.crt et Client01.key

Lancement du client Windows

Pour lancer la connexion, il suffit de faire un clic droit sur l'icône « OpenVPN » situé à gauche de l'heure et de choisir l'option « Connecter ».

Si tout se passe bien, une fenêtre affichant les logs doit s'afficher et une fois la connexion effectuée, le réseau est opérationnel.

En cas de problème, et pour trouver l'origine de celui-ci il faut augmenter le niveau des logs en changeant le paramètre « **verb** » du fichier de configuration :

- ▶ verb 3 -> Suffisamment de logs dans la plupart des cas.
- ▶ verb 9 -> Énormément de logs.

Une fois la connexion établie, il doit être possible de pinguer le serveur soit sur son adresse virtuelle (ex : 10.8.0.1 dans notre cas) soit sur son adresse réelle (ex : 192.168.0.1)

Table de routage

Avec la configuration précédente, les clients peuvent accéder au serveur OpenVPN, mais ils ne peuvent pas accéder au reste du réseau sur lequel est connecté le serveur OpenVPN.

Pour permettre aux clients d'accéder au reste du réseau, il faut effectuer deux opérations :

1 - Autoriser le serveur Linux à transmettre les paquets au reste du réseau

Pour cela, il faut activer le forwarding avec la commande suivante :

```
# echo 1 > /proc/sys/net/ipv4/ip_forward
```

La commande suivante, permet de vérifier que le forwarding est bien activé :

```
# cat /proc/sys/net/ipv4/ip_forward
```

2 - Indiquer aux autres postes du réseau la route vers le serveur OpenVPN

Si le serveur VPN est sur l'adresse 192.168.0.1, il faut ajouter une route manuellement sur chaque poste du réseau avec la commande suivante sous Windows :

```
C:\>route add 10.8.0.0 mask 255.255.255.0 192.168.0.1
```

La commande suivante, permet d'avoir la liste des routes :

```
C:\>route print
```

La commande suivante permet de supprimer une route :

```
C:\>route delete 10.8.0.0 mask 255.255.255.0 192.168.0.1
```

Pour ajouter une route sous Linux, il faut utiliser la commande suivante :

```
# route add -net 10.8.0.0 netmask 255.255.255.0 gw 192.168.0.1
```

La commande suivante, permet d'avoir la liste des routes sous Linux :

```
# route
```

La commande suivante permet de supprimer une route sous Linux :

```
# route delete -net 10.8.0.0 netmask 255.255.255.0 gw 192.168.0.1
```

Remarque : Pour éviter de devoir ajouter manuellement sur chaque poste du réseau une route, si c'est possible, il faut ajouter une route statique au niveau de la passerelle ou du routeur du réseau.

Installation Client Linux

L'installation du client est identique à celle du serveur, car c'est le même logiciel qui fait office de serveur ou de client en fonction de sa configuration :

```
# apt-get install openvpn liblzol
```

Le fichier de configuration et la gestion des clés est identique à celle du client Windows.

Historique des modifications

Version	Date	Commentaire
0.1	09/12/05	Création par Tony GALMICHE
0.2	23/03/06	Mise à jour pour publication