

**MISE EN PLACE
D'UN VPN AVEC
STUNNEL**

Résumé

Cette documentation est une aide à l'installation d'un VPN sous Debian à l'aide de l'utilitaire stunnel. Toutes les critiques sont les bienvenues.

La dernière version de cette documentation est disponible en ligne :
http://www.pileouface.org/linux/documentation/vpn_stunnel.pdf

Copyright

Auteur : Loïc Brayat, loack@pileouface.org

[Ce document peut être utilisé selon les termes de la Licence Publique Générale de GNU version 2 ou suivante.](#)

Il est permis de produire et distribuer des copies conformes de ce document à condition que la présente notice de copyright et la présente notice de permission soient préservées sur toutes les copies.

Il est permis de copier et distribuer des versions modifiées de ce document selon les conditions d'une copie conforme, à condition que le travail dérivé résultant soit entièrement distribué selon les termes d'une notice de permission identique à celle-ci.

Table des matières

Résumé.....	2
Copyright.....	2
Exemple de configuration.....	4
I \ Coté serveur.....	4
II \ Coté client.....	4
III \ Divers.....	4
Mise en place du serveur.....	5
I \ Création des certificats.....	5
1.Installation d'Openssl.....	5
2.Génération de la clef privé.....	5
II \ Permettre l'accès au réseau de la passerelle distante.....	5
1.Fichiers /etc/hosts.*	5
2.Configuration du firewall.....	6
III \ Lancement du serveur VPN.....	6
1.Installation de stunnel.....	6
2.Lancement du serveur.....	6
4.Ajouter la route du réseau local vers le distant.....	6
Mise en place du client.....	7
I \ Permettre l'accès du réseau local a la passerelle distante.....	7
1. configuration du firewall.....	7
II \ Lancement du client VPN.....	7
1. Installation de stunnel.....	7
2. Récupération des certificats.....	7
3. Lancement du client.....	7
4. Ajouter la route du réseau distant vers le local.....	7

EXEMPLE DE CONFIGURATION

I \ Coté serveur

Réseau interne : 192.168.0.0/24

Adresse privée de la passerelle : 192.168.0.1

Adresse publique de la passerelle : ip.public.serveur (62.212.x.x)

Adresse vpn de la passerelle : ip.vpn.serveur (10.99.99.1, par exemple)

II \ Coté client

Réseau interne : 192.168.1.0/24

Adresse privée de la passerelle : 192.168.1.1

Adresse publique de la passerelle : ip.public.client (62.212.y.y)

Adresse vpn de la passerelle : ip.vpn.client (10.99.99.2, par exemple)

III \ Divers

Port utilisé pour le VPN : 5555

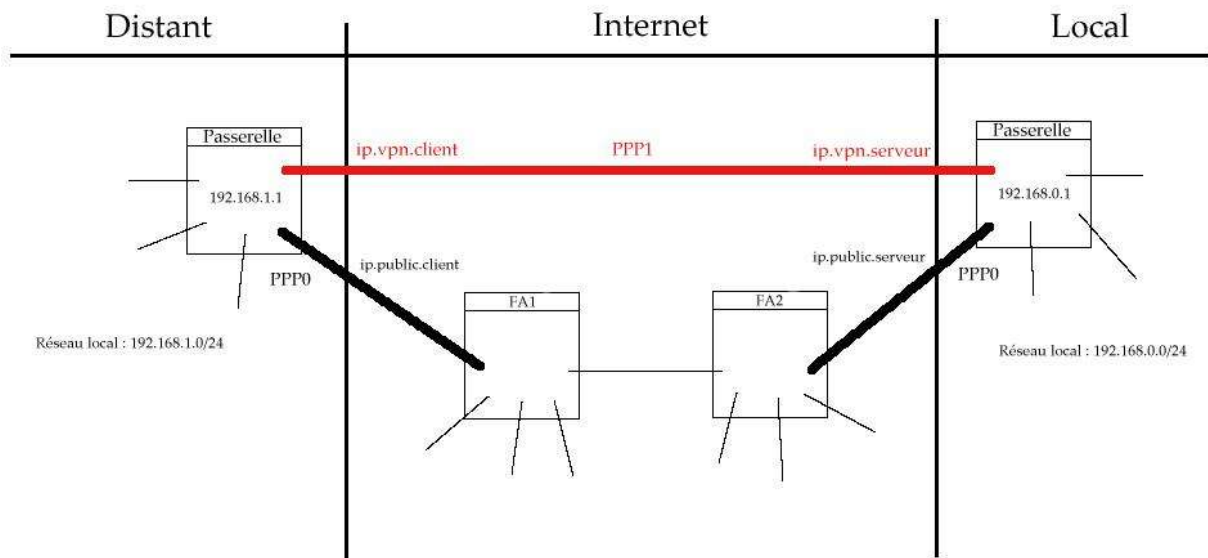
Durée de validité des certificats : 1an

Format des certificats : x509

Longueur de la clef : 512

Les connexions au FA seront établie par ppp : ppp0

La connexion VPN sera établie par ppp1



MISE EN PLACE DU SERVEUR

I \ Création des certificats

1. Installation d'Openssl

A partir des paquets debian : Apt-get install openssl

A partir des sources : <http://www.openssl.org/source/>

2. Génération de la clef privé

- Taper les lignes suivantes dans /etc/ssl/certs/ :
openssl req -new -x509 -nodes -days 365 -out stunnel.pem -keyout stunnel.pem
chmod 600 stunnel.pem
dd if=/dev/urandom of=temp_file count=2
openssl dhparam -rand temp_file 512 >> stunnel.pem
ln -sf stunnel.pem `openssl x509 -noout -hash < stunnel.pem`.0
- La clef est contenue dans le fichier stunnel.pem

II \ Permettre l'accès au réseau de la passerelle distante

1. Fichiers /etc/hosts.*

Configurer le fichier /etc/hosts.deny afin d'être sécurisé :

```
ALL: ALL
```

Configurer le fichier /etc/hosts.allow afin de permettre un accès :

```
stunnel: ip.public.client  
pppd: ip.public.client
```

2. Configuration du firewall

Permettre à la passerelle distante (cliente) d'envoyer et de recevoir des informations au réseau local.

```
$ADR_VPN = ip.public.client
$PORT_VPN = 5555
```

```
#On accepte la connexion de la passerelle distante VPN sur le PORT_VPN
$IPTABLES -A INPUT -i ppp0 -s $ADR_VPN -m state --state NEW,ESTABLISHED -p tcp
--dport $PORT_VPN -j LOG_ACCEPT
$IPTABLES -A OUTPUT -o ppp0 -d $ADR_VPN -m state --state ESTABLISHED -p tcp
--sport $PORT_VPN -j LOG_ACCEPT
```

```
#On accepte la connexion sur le ppp1 cad sur le VPN sur la passerelle
$IPTABLES -A INPUT -i ppp1 -m state --state NEW,ESTABLISHED -j LOG_ACCEPT
$IPTABLES -A OUTPUT -o ppp1 -m state --state ESTABLISHED -j LOG_ACCEPT
```

```
#On donne acces au VPN au reseau local
$IPTABLES -A FORWARD -i ppp1 -o eth1 -j ACCEPT
$IPTABLES -A FORWARD -o ppp1 -i eth1 -j ACCEPT
```

Pour plus d'informations sur le firewall, lire ma documentation sur Iptables :
http://www.pileouface.org/linux/documentation/firewall_iptables.pdf

III \ Lancement du serveur VPN

1. Installation de stunnel

A partir des paquets debian : apt-get install stunnel
A partir des sources : <http://www.stunnel.org/download/>

2. Lancement du serveur

```
stunnel -d 5555 [-f] -v3 -D7 -p /etc/ssl/certs/stunnel.pem -L /usr/sbin/pppd -- pppd local noauth
```

L'option "-f" force le démon à rester en avant plan. C'est plus pratique pour déboguer mais inutile en exploitation.

L'option "-v3" force la vérification des certificats. Par défaut, n'importe qui peut se connecter.

4. Ajouter la route du réseau local vers le distant

```
route add -net 192.168.1.0/24 gw ip.vpn.client
```

MISE EN PLACE DU CLIENT

I \ Permettre l'accès du réseau local a la passerelle distante

1. configuration du firewall

```
#On accepte la connexion sur le ppp1 cad sur le VPN sur la passerelle
$IPTABLES -A INPUT -i ppp1 -m state --state NEW,ESTABLISHED -j LOG_ACCEPT
$IPTABLES -A OUTPUT -o ppp1 -m state --state ESTABLISHED -j LOG_ACCEPT

#On donne acces au VPN au reseau local
$IPTABLES -A FORWARD -i ppp1 -o eth1 -j ACCEPT
$IPTABLES -A FORWARD -o ppp1 -i eth1 -j ACCEPT
```

II \ Lancement du client VPN

1. Installation de stunnel

A partir des paquets debian : apt-get install stunnel
A partir des sources : <http://www.stunnel.org/download/>

2. Récupération des certificats

Copier le fichier /etc/ssl/certs/stunnel.pem depuis le serveur.

3. Lancement du client

```
stunnel -c -r ip.public.serveur:5555 -D7 -p /etc/ssl/certs/stunnel.pem -L /usr/sbin/pppd -- pppd
local noauth lcp-echo-interval 50 lcp-echo-failure 3 ip.vpn.client:ip.vpn.serveur
```

4. Ajouter la route du réseau distant vers le local

```
route add -net 192.168.0.0/24 gw ip.vpn.serveur
```

Remarque :

* Si les deux réseaux ont le même masque, il faut ajouter des routes vers les machines. (route add -host ip.locale.machine.distante gw ip.vpn.serveur)