

# Configuration du client openvpn sur wheezy

Utiliser le serveur DHCP et les DNS du réseau distant

dimanche 10 novembre 2013, par [Pierrox](#)

## 1) Configuration du serveur openvpn

Pour info, voici le fichier de configuration du serveur en mode bridge afin de fournir les infos DHCP et DNS du réseau :

### server.conf

```
1. port 1194
2. proto tcp
3. dev tap0
4.
5. ca ca.crt
6. cert server.crt
7. key server.key
8. dh dh1024.pem
9.
10. server-bridge
11.
12. ### Envoyer les infos DHCP au clients ###
13. push "dhcp-option DNS 192.168.122.1"
14. push "dhcp-option DOMAIN domain.loc"
15.
16. keepalive 10 120
17. comp-lzo
18.
19. max-clients 8
20.
21. persist-key
22. persist-tun
23.
24. status openvpn-status.log
25. log-append /var/log/openvpn.log
26.
27. user openvpn
28. group openvpn
29.
30. persist-key
31. persist-tun
32. status openvpn-status.log
33. verb 3
34.
```

35. username-as-common-name

[Télécharger](#)

## 2) Configuration du client openvpn

Ci-dessous la configuration client nommer **client.conf**. J'insiste sur L'extension, car le script de démarrage d'openvpn va lire les fichiers aux extension \*.conf et demarrer autant de tunnel que de fichier .conf

### **client.conf**

1. client
2. dev tap0
3. proto tcp
4. remote xxxxxxxx.com 1194 #IP PORT
5. resolv-retry infinite
6. nobind
7. persist-key
8. persist-tun
9. ca ca.crt
10. cert server.crt
11. key server.key
12. comp-lzo
13. auth-user-pass
14. verb 3
15. auth-user-pass login.txt
16. log-append /var/log/openvpn.log
- 17.
18. #IMPORTANT pour obtenir une adresse IP du serveur réseau dhcp distant
19. #Installer le paquet resolvconf pour debian
- 20.
21. script-security 2
22. up update-resolv-conf
23. down update-resolv-conf

[Télécharger](#)

On peut tester la config et suivre les logs :

```
service openvpn start & tail -f /var/log/openvpn.log
```

## 3) Obtenir l'adresse IP et le DNS via le DHCP du vpn

Lorsque le service openvpn se connecte il démarre pas l'interface tap0 je devais taper les commandes suivantes :

```
ifconfig tap0 up  
dhclient tap0
```

## 4) Automatiser l'obtention de l'IP et du DNS

Pour utiliser le script ci-dessous il est important d'installer le paquet resolvconf

1. **apt-get install** resolvconf

Pour automatiser toute la procédure de connexion j'ai modifié le script update-resolv-conf dans /etc/openvpn

**/etc/openvpn/update-resolv-conf : L 31 -> 36**

```
1. ##### RESOLVCONF DOIT ETRE INSTALLE #####  
2. [ -x /sbin/resolvconf ] || exit 0  
3.  
4. case $script_type in  
5.  
6. up)  
7.     for optionname in ${!foreign_option_*} ; do  
8.         option="${!optionname}"  
9.         echo $option  
10.        part1=$(echo "$option" | cut -d " " -f 1)  
11.        if [ "$part1" == "dhcp-option" ] ; then  
12.            part2=$(echo "$option" | cut -d " " -f 2)  
13.            part3=$(echo "$option" | cut -d " " -f 3)  
14.            if [ "$part2" == "DNS" ] ; then  
15.                IF_DNS_NAMESERVERS="$IF_DNS_NAMESERVERS $part3"  
16.            fi  
17.            if [ "$part2" == "DOMAIN" ] ; then  
18.                IF_DNS_SEARCH="$IF_DNS_SEARCH $part3"  
19.            fi  
20.        fi  
21.    done  
22.    R=""  
23.    for SS in $IF_DNS_SEARCH ; do  
24.        R+="${R}search $SS  
25.    "  
26.    done  
27.    for NS in $IF_DNS_NAMESERVERS ; do  
28.        R+="${R}nameserver $NS  
29.    "  
30.    done  
31.  
32.    # Recupere les DNS du VPN  
33.    echo -n "$R" | /sbin/resolvconf -a "${dev}.inet"
```

```
35. # Démarrer l'interface passé en paramètre du script (voir dev tapxx ds cfg)
36. /sbin/ifconfig ${dev} up
37. sleep 3
38.
39. # Demander une IP au Serveur DHCP au reseau PV
40. (
41.     /sbin/dhclient ${dev} &
42. )
43. ;;
44. down)
45.     /sbin/resolvconf -d "${dev}.inet"
46.     ;;
47. esac
```

[Télécharger](#)

Et voilà, désormais un simple `service openvpn start` connecte directement la machine au réseau privé et récupère une ip en dhcp.

A noter que le service démarre automatiquement au boot de la machine.