2017-2018

Mise en place d'un VPN avec OpenVpn

Epreuve E6

Raphaël Andrieu
ARCONIC

Table des matières

Raphael Andrieu

Cahier des charges

L'entreprise Xinthya souhaiterait mettre à disposition des utilisateurs nomades un système leur permettant de se connecter à distance au réseau de l'entreprise, depuis leur domicile ou d'un hôtel quand ils sont en déplacement.

Les clients connectés au VPN doivent pouvoir accéder à internet et au réseau local de l'entreprise pour avoir accès au serveur de fichiers Windows.

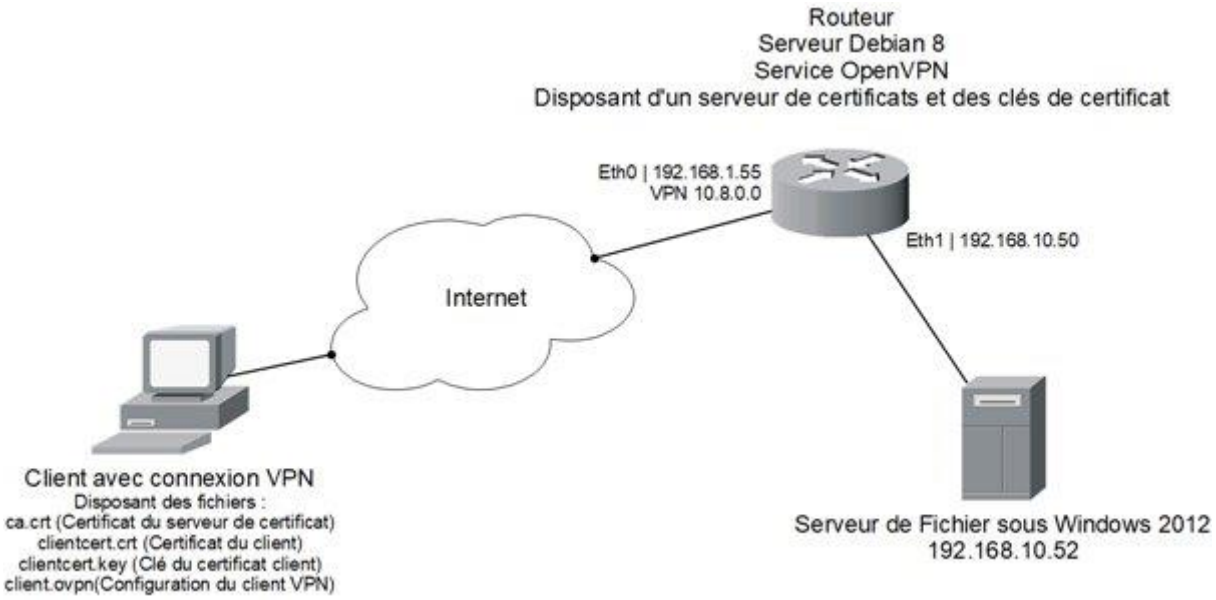
Solution proposée :

Pour répondre à ce besoin, nous proposons de mettre en place une solution reliant de façon sécurisée le poste de l'utilisateur au réseau de l'entreprise. Durant la connexion établie depuis l'extérieur, tout le trafic avec le réseau local sera chiffré et l'utilisateur aura accès aux ressources. Une telle solution est appelée VPN.

Matériels et logiciels utilisés

- Un serveur sous debian 8 hébergeant la solution VPN
- Un serveur sous windows 2012 hébergeant les ressources du réseau local
- Un client sous Windows 10
- OpenVPN : Service VPN sur le serveur Debian 8.
- FileZilla : Logiciel qui nous permettra de récupérer certains fichiers depuis le poste Client Windows sur le Serveur Debian 8
- PROFTPD : Service sur le Serveur Debian 8 qui nous permettra de récupérer certains fichiers depuis le poste Client Windows sur le Serveur Debian 8

Schéma du résultat attendu



Raphael Andrieu

Présentation d'OpenVPN

OpenVPN est un logiciel libre permettant de créer un réseau privé virtuel (VPN), l'une de ses principales fonctions est l'accès à des ordinateurs distants comme si l'on était connecté au réseau local. Il permet donc d'avoir un accès au réseau interne (réseau d'entreprise, par exemple) ou de créer un réseau de pairs.

Mise en place des cartes réseaux

Avant de mettre en le service OpenVPN, il faut paramétrer les 2 cartes réseaux du serveur Debian 8.

Une carte eth0 faisant le lien avec l'exterieur et une carte eth1 reliée avec le réseau local

Ouvrez un Terminal et connectez-vous en root avec « su root » et ouvrez le fichier d'interfaces en écrivant « nano /etc/network/interfaces »

```
  GNU nano 2.2.6        Fichier : /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback
```

Pour mettre en place Eth0 et Eth1, il vous faut ajouter les lignes :

```
  GNU nano 2.2.6        Fichier : /etc/network/interfaces          Modifié

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto eth0
iface eth0 inet static
        address 192.168.1.55
        netmask 255.255.255.0
        gateway 192.168.1.1

auto eth1
iface eth1 inet static
        address 192.168.10.50
        netmask 255.255.255.0
```

Nous avons donc paramétré Eth0 avec comme adresse 192.168.1.55, un masque de sous réseau en 255.255.255.0 et une passerelle en 192.168.1.1 qui nous permettra de nous connecter à internet.

Eth1 a comme adresse 192.168.10.50 et un masque de sous réseaux en 255.255.255.0.

Sauvegardez les paramètres et fermez le fichier, nous allons maintenant mettre un DNS, pour cela modifier le fichier resolv.conf en faisant « nano /etc/resolv.conf » et ajoutez la ligne « nameserver 8.8.8.8 » Cette adresse correspond au DNS de google.

```
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
  GNU nano 2.2.6          Fichier : /etc/resolv.conf                Modifié

# Generated by NetworkManager
nameserver 8.8.8.8
```

Après avoir modifié ces 2 fichiers, vous devez redémarrer le service réseau en écrivant « /etc/init.d/networking restart », vous pouvez vérifier les paramètres des cartes réseaux en écrivant « ifconfig »

```
rtt min/avg/max/mdev = 37.644/58.978/80.313/21.317 ms
root@srv-deb-ar:/home/andri# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:b8:83:86
          inet adr:192.168.1.55  Bcast:192.168.1.255  Masque:255.255.255.0
          adr inet6: fe80::a00:27ff:feb8:8386/64 Scope:Lien
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5151 errors:0 dropped:0 overruns:0 frame:0
          TX packets:512 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:774470 (756.3 KiB)  TX bytes:55001 (53.7 KiB)

eth1      Link encap:Ethernet  HWaddr 08:00:27:64:96:1f
          inet adr:192.168.10.50  Bcast:192.168.10.255  Masque:255.255.255.0
          adr inet6: fe80::a00:27ff:fe64:961f/64 Scope:Lien
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3948 errors:0 dropped:0 overruns:0 frame:0
          TX packets:102 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:394728 (385.4 KiB)  TX bytes:11698 (11.4 KiB)
```

Vous pouvez maintenant essayer de pinger l'IP 8.8.8.8 et www.google.fr pour vérifier que vous avez bien accès à internet

```
mot de passe :
root@srv-deb-ar:/home/andri# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=5 ttl=251 time=985 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=251 time=985 ms (DUP!)
64 bytes from 8.8.8.8: icmp_seq=6 ttl=251 time=47.0 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=251 time=47.1 ms (DUP!)
64 bytes from 8.8.8.8: icmp_seq=7 ttl=251 time=37.0 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=251 time=37.1 ms (DUP!)
64 bytes from 8.8.8.8: icmp_seq=8 ttl=251 time=38.9 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=251 time=38.9 ms (DUP!)
^C
```

```
root@srv-deb-ar:/home/andri# ping www.google.fr
PING www.google.fr (216.58.206.227) 56(84) bytes of data.
64 bytes from par10s34-in-f3.1e100.net (216.58.206.227): icmp_seq=1 tt
=37.6 ms
64 bytes from par10s34-in-f3.1e100.net (216.58.206.227): icmp_seq=1 tt
=37.6 ms (DUP!)
64 bytes from par10s34-in-f3.1e100.net (216.58.206.227): icmp_seq=2 tt
=80.2 ms
64 bytes from par10s34-in-f3.1e100.net (216.58.206.227): icmp_seq=2 tt
=80.3 ms (DUP!)
^C
```

Mise en place d'OpenVPN sur Debian

Maintenant que nous avons un accès à internet, nous pouvons commencer l'installation OpenVpn, pour cela il faut taper la commande « apt-get update && apt install -y openvpn »

Cette commande permet de mettre à jour les paquets et installer OpenVPN. Après avoir fini l'installation, nous allons copier le dossier easy-rsa qui contient les scripts permettant la configuration des clés et des certificats, pour cela il faut écrire « cp -a /usr/share/easy-rsa /etc/openvpn/

Puis nous allons nous positionner dans le dossier easy-rsa avec « cd /etc/openvpn/easy-rsa/

Ensuite nous allons lancer une liste de commande qui va nous permettre des créer les clés et les certificats :

Configuration d'un serveur de certificats

1. source vars
2. ./clean-all
3. ./build-ca

```
root@srv-deb-ar:/etc/openvpn/easy-rsa# ./clean-all
root@srv-deb-ar:/etc/openvpn/easy-rsa# ./build-ca
Generating a 2048 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) [US]:
```

```
root@srv-deb-ar:/etc/openvpn/easy-rsa# ./build-ca
Generating a 2048 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) [US]:
State or Province Name (full name) [CA]:
Locality Name (eg, city) [SanFrancisco]:
Organization Name (eg, company) [Fort-Funston]:
Organizational Unit Name (eg, section) [MyOrganizationalUnit]:
Common Name (eg, your name or your server's hostname) [Fort-Funston CA]:
Name [EasyRSA]:
Email Address [me@myhost.mydomain]:
root@srv-deb-ar:/etc/openvpn/easy-rsa#
```

Plusieurs informations vous seront demandées, vous pouvez appuyer sur entrée pour chaque proposition ce qui laissera les propositions par défaut. Vous pouvez les indiquer si vous le voulez. Ces propositions par défaut sont modifiables dans le fichier « /etc/openvpn/easy-rsa/vars », pensez a retapez les commandes « source vars » et « clean-all » après la modification.

Vous pouvez vérifier que les fichiers ont bien été créés en faisant « ls -la keys/ », cette commande va vous afficher la liste des fichiers du dossier keys, vous pouvez donc vérifier si les certificats qui caractérisent le serveur de certificat sont bien présent : ca.crt et ca.key

```
Email Address [me@myhost.mydomain]:
root@srv-deb-ar:/etc/openvpn/easy-rsa# ls -la keys/
total 20
drwx------ 2 root root 4096 oct.  17 18:09 .
drwxr-xr-x 3 root root 4096 oct.  17 18:07 ..
-rw-r--r-- 1 root root 1818 oct.  17 18:09 ca.crt
-rw------- 1 root root 1704 oct.  17 18:09 ca.key
-rw-r--r-- 1 root root    0 oct.  17 18:07 index.txt
-rw-r--r-- 1 root root    3 oct.  17 18:07 serial
root@srv-deb-ar:/etc/openvpn/easy-rsa#
```

Génération de la clé Diffie-Hellman

Maintenant on génère la clé Diffie-Hellman qui va sécuriser les échanges du VPN, pour cela il faut écrire « ./build-dh ». Cette opération peut prendre un peu de temps.

```
root@srv-deb-ar:/etc/openvpn/easy-rsa# ./build-dh
Generating DH parameters, 2048 bit long safe prime, generator 2
This is going to take a long time
.....................+.............................+.....................
................................+..................
```

Puis on génère les certificats du serveur en faisant « ./build-key-server srvcert »

Le srvcert est le nom des certificats serveur, vous pouvez mettre ce que vous voulez mais ce nom est signifcatif : srv = serveur et cert = certificat, ce qui donne srvcert.

Pareil que pour la commande « ./build-ca » , plusieurs informations vous seront demandées, indiqué le nom de votre serveur Debian a la ligne « Common name »

```
root@srv-deb-ar:/etc/openvpn/easy-rsa# ./build-key-server srvcert
Generating a 2048 bit RSA private key
.................................+++
.........+++
writing new private key to 'srvcert.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) [US]:
```

Configuration d'OpenVPN sur le serveur

Maintenant qu'OpenVPN est installé, et que les clés et les certificats sont générés, nous allons copier la configuration par défaut de Openvpn

1. gunzip -c /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz > /etc/openvpn/server.conf

Puis nous allons éditer le fichier de configuration avec « nano /etc/openvpn/server.conf »

Décommenter les lignes :

- user nobody
- group nogroup
- push "redirect-gateway def1 bypass-dhcp"

Les lignes « user nobody » et « group nogroup » permettent d'augmenter la sécurité. La ligne « push « redirect-gateway def1 bypass-dhcp » permet d'ajouter une route directe au serveur DHCP.

Puis ajouter la ligne « push "dhcp-option DNS 8.8.8.8" » et « push "route 192.168.10.0 255.255.255.0" »

Modifier les chemins des certificats et des clés :

- ca /etc/openvpn/easy-rsa/keys/ca.crt
- cert /etc/openvpn/easy-rsa/keys/srvcert.crt
- key /etc/openvpn/easy-rsa/keys/srvcert.key # This file should be kept secret
- dh /etc/openvpn/easy-rsa/keys/dh2048.pem

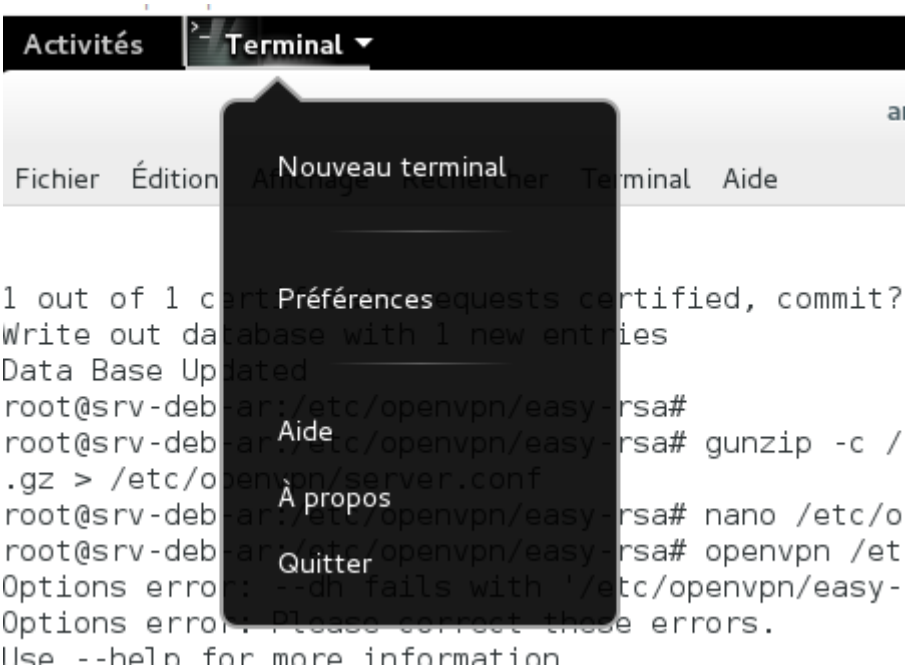
Sauvegardez et quittez le fichier, vous pouvez maintenant lancer OpenVPN avec la commande

« openvpn /etc/openvpn/server.conf »

Si tout se passe bien, il devrait vous mettre « initialization sequence completed »

```
root@srv-deb-ar:/etc/openvpn/easy-rsa# openvpn /etc/openvpn/server.conf
Tue Oct 17 18:43:55 2017 OpenVPN 2.3.4 i586-pc-linux-gnu [SSL (OpenSSL)] [LZO] [EPOLL] [PKCS11] [MH] [IPv6] buil
t on Jun 26 2017
Tue Oct 17 18:43:55 2017 library versions: OpenSSL 1.0.1t  3 May 2016, LZO 2.08
Tue Oct 17 18:43:55 2017 NOTE: your local LAN uses the extremely common subnet address 192.168.0.x or 192.168.1.
x.  Be aware that this might create routing conflicts if you connect to the VPN server from public locations suc
h as internet cafes that use the same subnet.
Tue Oct 17 18:43:55 2017 Diffie-Hellman initialized with 2048 bit key
Tue Oct 17 18:43:55 2017 Socket Buffers: R=[163840->131072] S=[163840->131072]
Tue Oct 17 18:43:55 2017 ROUTE_GATEWAY 192.168.1.1/255.255.255.0 IFACE=eth0 HWADDR=08:00:27:b8:83:86
Tue Oct 17 18:43:55 2017 TUN/TAP device tun0 opened
Tue Oct 17 18:43:55 2017 TUN/TAP TX queue length set to 100
Tue Oct 17 18:43:55 2017 do_ifconfig, tt->ipv6=0, tt->did_ifconfig_ipv6_setup=0
Tue Oct 17 18:43:55 2017 /sbin/ip link set dev tun0 up mtu 1500
Tue Oct 17 18:43:55 2017 /sbin/ip addr add dev tun0 local 10.8.0.1 peer 10.8.0.2
Tue Oct 17 18:43:55 2017 /sbin/ip route add 10.8.0.0/24 via 10.8.0.2
Tue Oct 17 18:43:55 2017 UDPv4 link local (bound): [undef]
Tue Oct 17 18:43:55 2017 UDPv4 link remote: [undef]
Tue Oct 17 18:43:55 2017 MULTI: multi_init called, r=256 v=256
Tue Oct 17 18:43:55 2017 IFCONFIG POOL: base=10.8.0.4 size=62, ipv6=0
Tue Oct 17 18:43:55 2017 IFCONFIG POOL LIST
Tue Oct 17 18:43:55 2017 Initialization Sequence Completed
```

Ouvrez un nouveau terminal, connectez-vous en root avec « su root » et vérifiez que le serveur a bien ouvert un nouveau point d'accès en écrivant « ifconfig tun0 »




```
root@srv-deb-ar:/home/andri# ifconfig tun0
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)

root@srv-deb-ar:/home/andri#
```

Vous pouvez voir qu'une nouvelle carte réseau est apparue sous le nom de tun0-00, elle dispose d'une adresse en 10.8.0.1 et un masque en 255.255.255.255. Cette adresse nous permettra d'accéder au réseau du VPN.

Nous allons maintenant créer les certificats pour l'ordinateur qui va se connecter au VPN.

Nous nous positionnons dans le dossier easy-rsa « cd /etc/openvpn/easy-rsa/ puis on tape la commande « source vars » et nous créons la clé avec « ./build-key clientcert »

Pareil que pour le certificat du serveur, le nom « clientcert » peut être changé par le nom de votre choix. Après avoir saisie la commande, plusieurs indications vous sont demandé, indiquez le nom de votre pc dans « common name »

Puis nous allons copier les certificats et les clés du client dans le dossier OpenVPN, pour les récupérer plus tard en FTP.

- Cd /etc/openvpn/easy-rsa/
- Cp keys/ca.crt keys/clientcrt.crt keys/clientcrt.key /etc/openvpn

Il faut donner l'accès en lecture au fichier clientcert.key, on va donc se placer dans le dossier openvpn et changer les autorisations avec chmod

- Cd /etc/openvpn
- Chmod 744 clientcert.key

Le 744 signifie que le root a accès en écriture/lecture/Execution(7) et les autres en lecture(4)

Maintenant que le serveur OpenVPN est installé, nous allons nous occuper des routes qui permettront le trafic vers d'autres machines et vers internet.

Configuration du routage sur le serveur OpenVPN

Editez le fichier «sysctl.conf » avec « nano /etc/sysctl.conf » et décommenter la ligne :

« net.ipv4.ip_forward=1 »

Puis recharger la configuration avec « sysctl -p /etc/sysctl.conf »

Mise en place du NAT

Pour utiliser notre routeur Linux comme passerelle internet pour tous les sous-réseaux, nous devons y implémenter la translation d'adresses, le NAT. Cela permettra aux adresses privées des réseaux locaux d'accéder à Internet.

Nous allons installer iptables-persistent qui nous permettra de garder le NAT activer après un redémarrage du serveur.

« apt install iptables-persistent »

Répondre « oui » aux 2 choses.

Puis nous allons taper les lignes

- iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
- iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o eth0 -j MASQUERADE

Vous pouvez vérifier que les règles ont bien été créés avec « iptables -t nat -L »

```
root@srv-deb-ar:/home/andri# iptables -t nat -A POSTROUTING -o eth0 -j MASQUE
E
root@srv-deb-ar:/home/andri# iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o
h0 -j MASQUERADE
root@srv-deb-ar:/home/andri# iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
MASQUERADE  all  --  anywhere             anywhere
MASQUERADE  all  --  10.8.0.0/24          anywhere
```

Nous sauvegardons les règles pour le prochain redémarrage avec la commande « /etc/init.d/netfilter-persistent save »

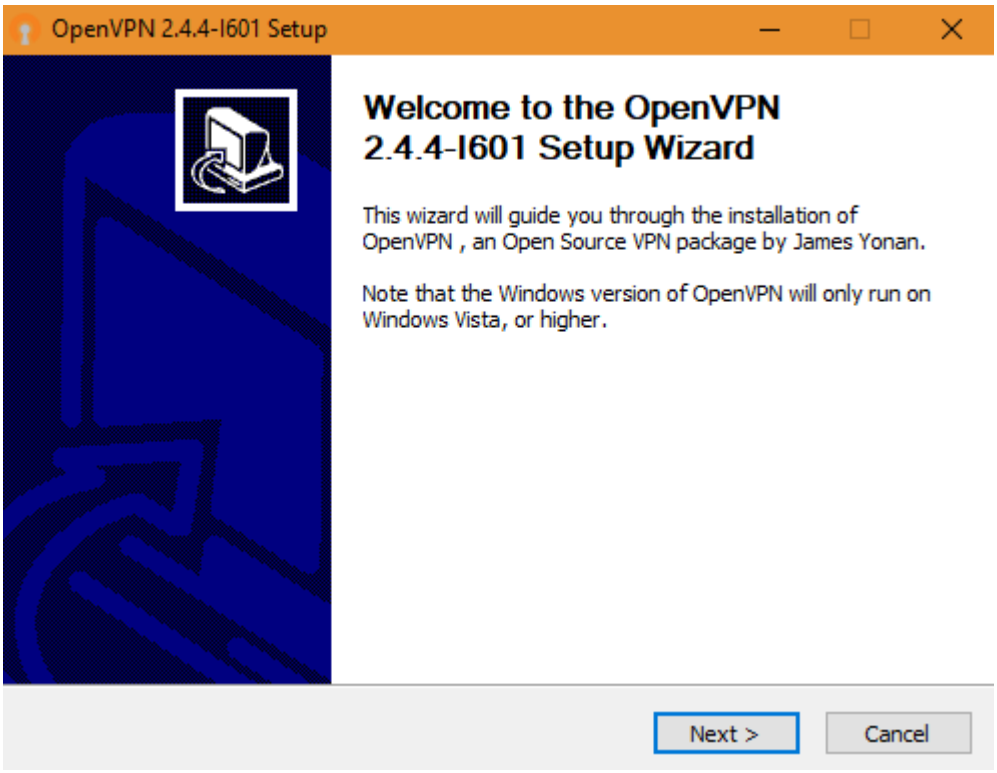
```
root@srv-deb-ar:/home/andri# /etc/init.d/netfilter-persistent save
[....] Saving netfilter rules...run-parts: executing /usr/share/netfilter-persis
tent/plugins.d/15-ip4tables save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables save
done.
root@srv-deb-ar:/home/andri#
```

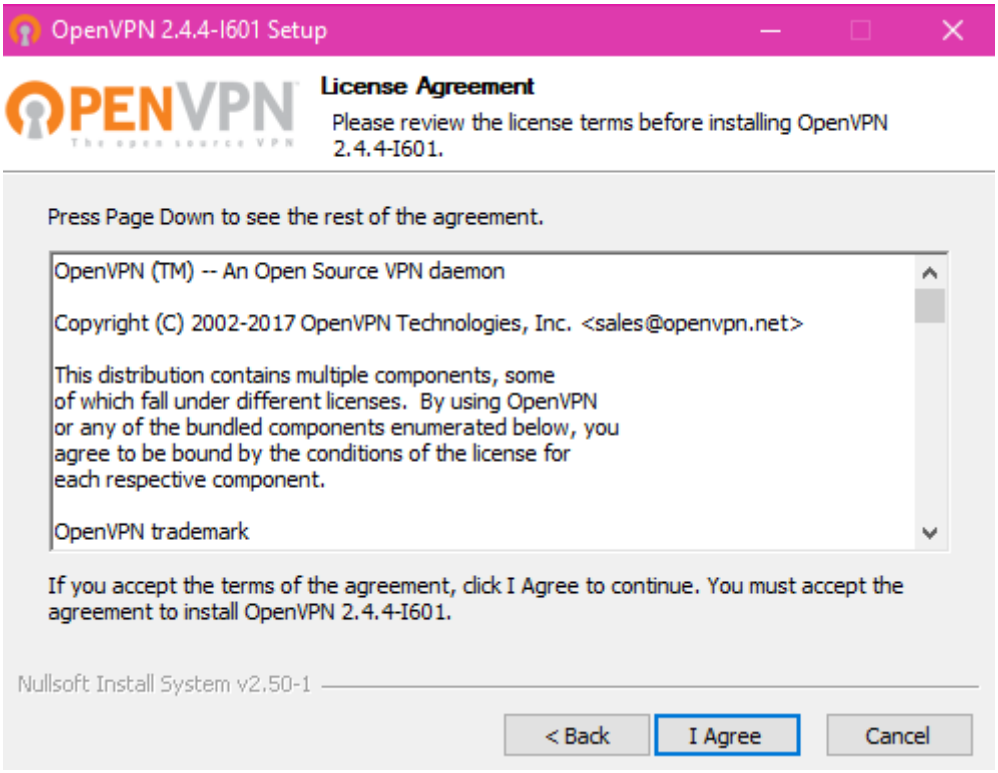
Vous pouvez redémarrer le serveur pour vérifier que tout est bien rechargé avec « reboot » puis dès que le serveur est redémarré avec « iptables -t nat -L »

Installation d'OpenVPN sur un client Windows

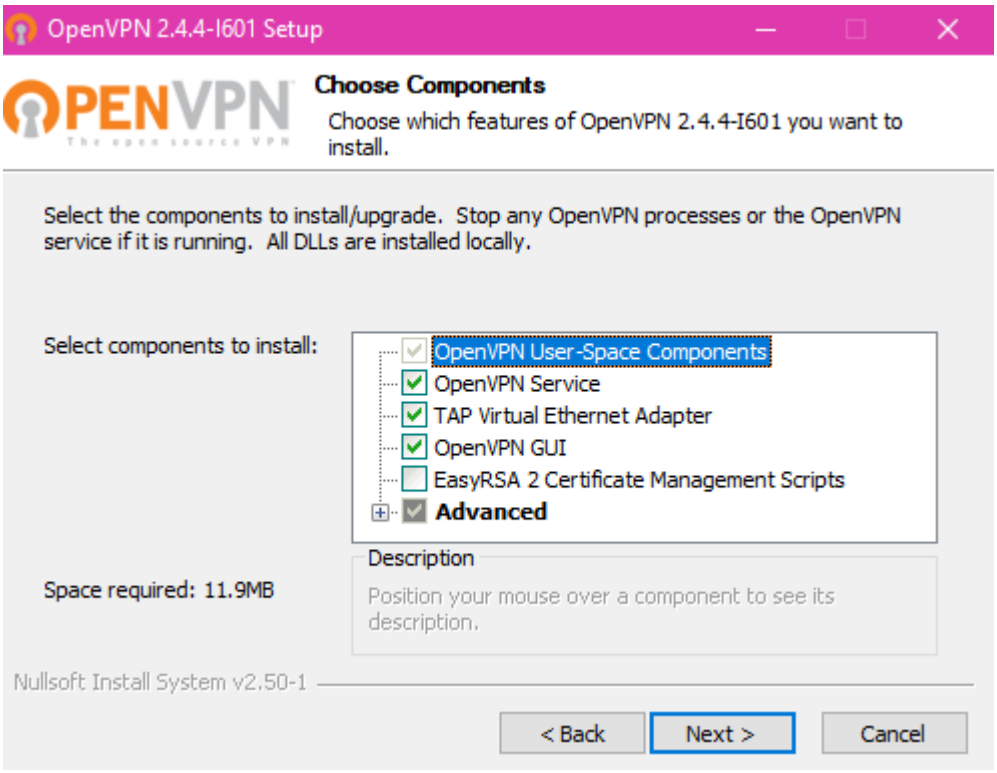
Téléchargez l'installateur OpenVPN sur le site Openvpn.net.

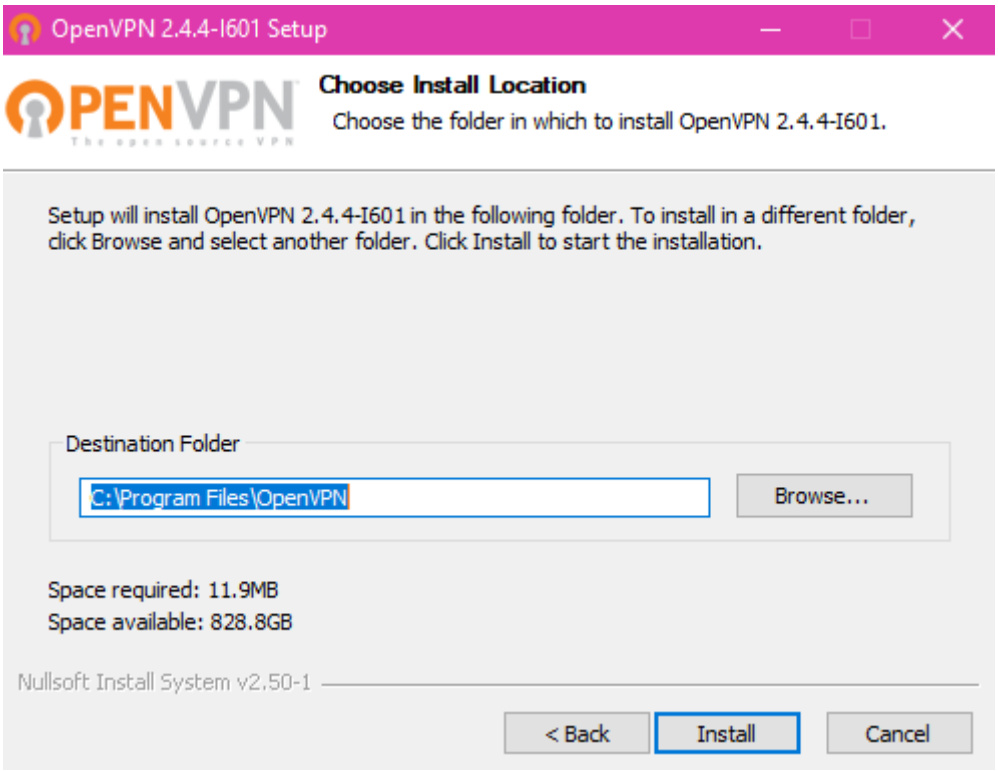
Raphael Andrieu

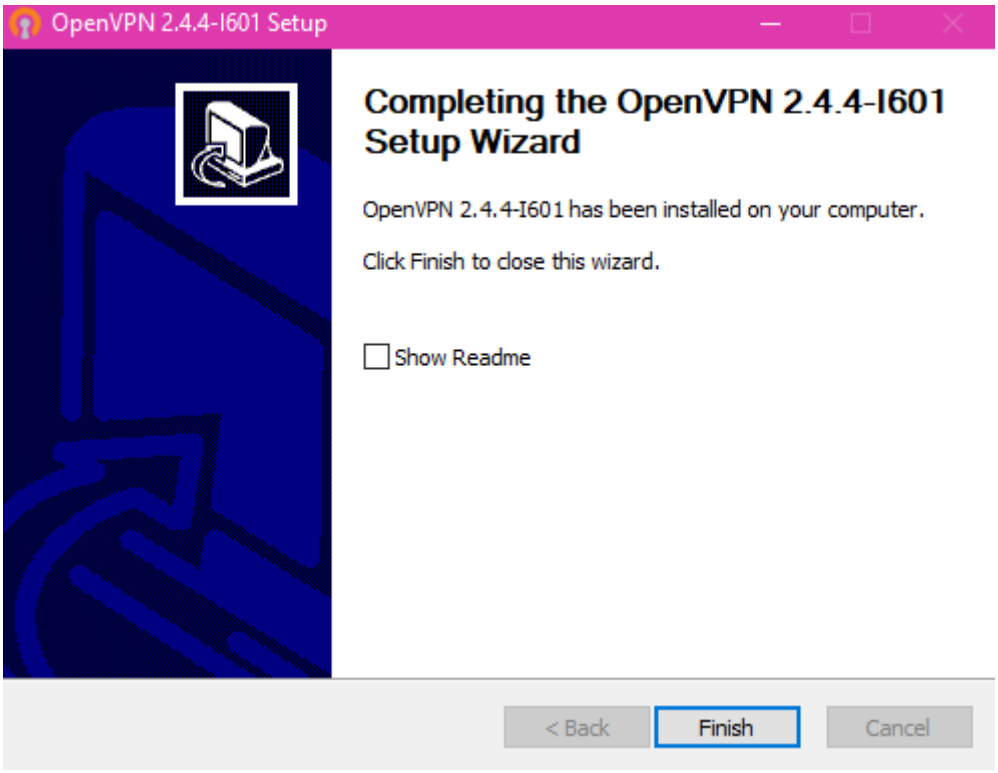




Raphael Andrieu





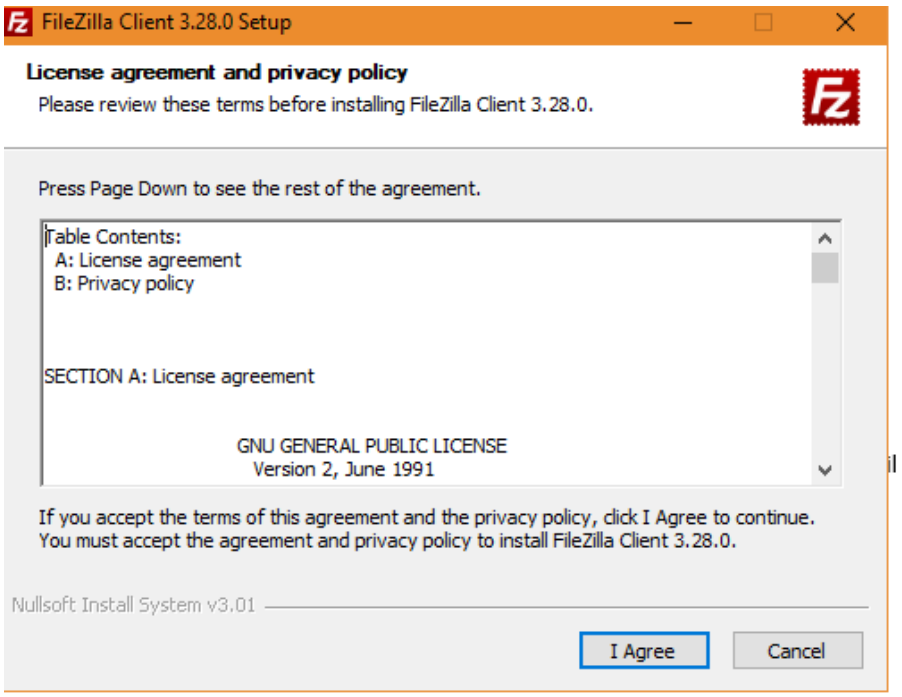


Récupération des certificats sur le client Windows

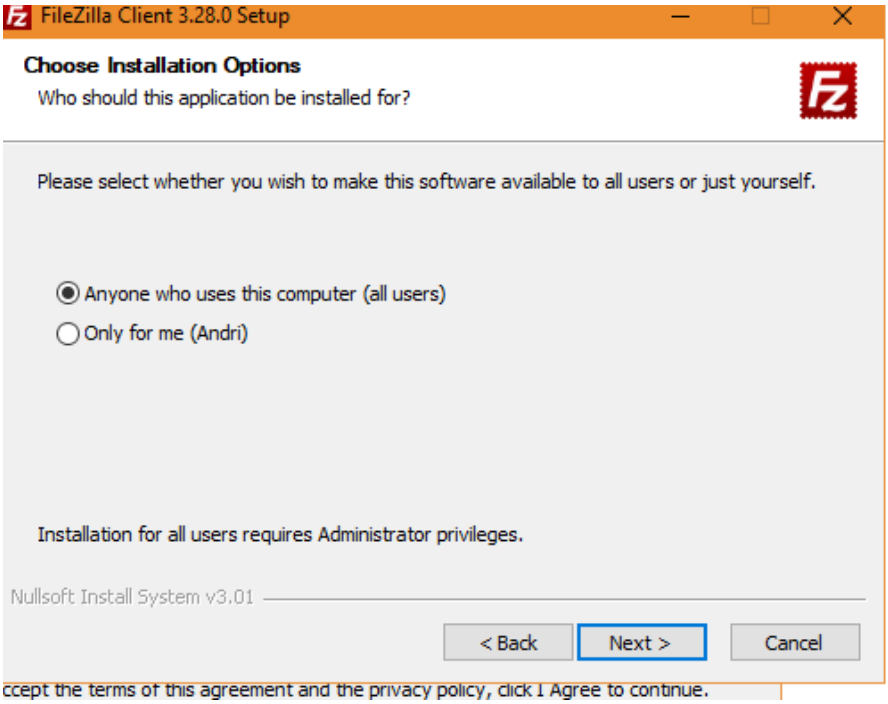
Pour récupérer les certificats sur le client Windows, j'utilise le service PROFTPD sur Debian et le logiciel FileZilla sur le client Windows qui est une possibilité parmi tant d'autre.

Pour installer le service PROFTPD, il faut taper la commande « apt install proftpd »

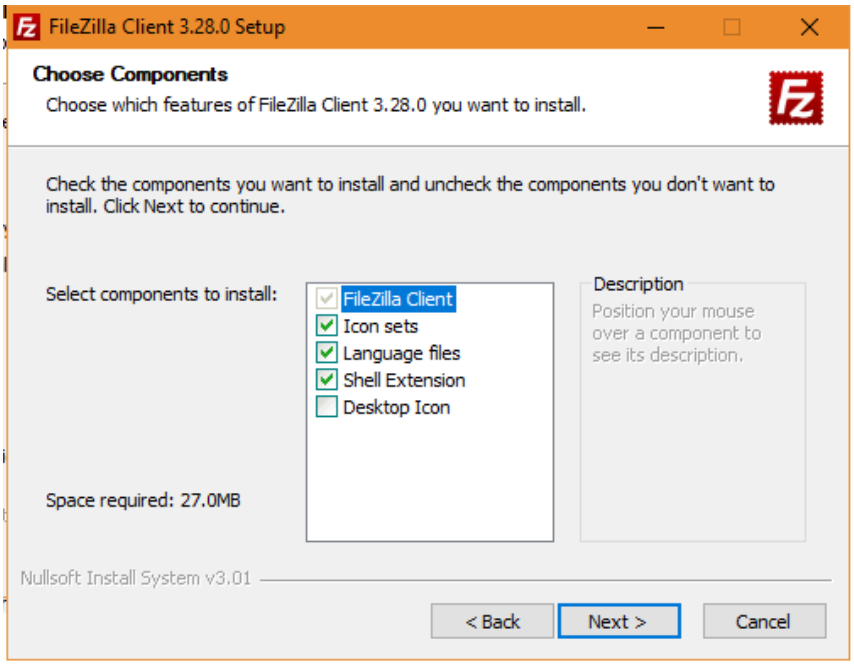
Pour installer Filezilla, il vous faut accéder au site internet : https://filezilla-project.org/ et télécharger l'exécutable de la version Client, après l'avoir installé, lancez-le et suivez les instructions



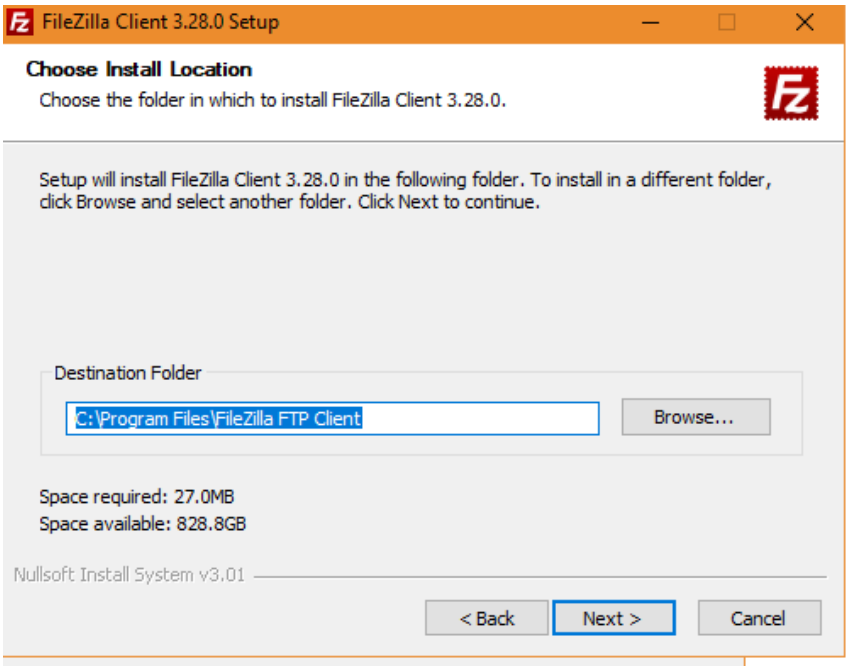
Raphael Andrieu

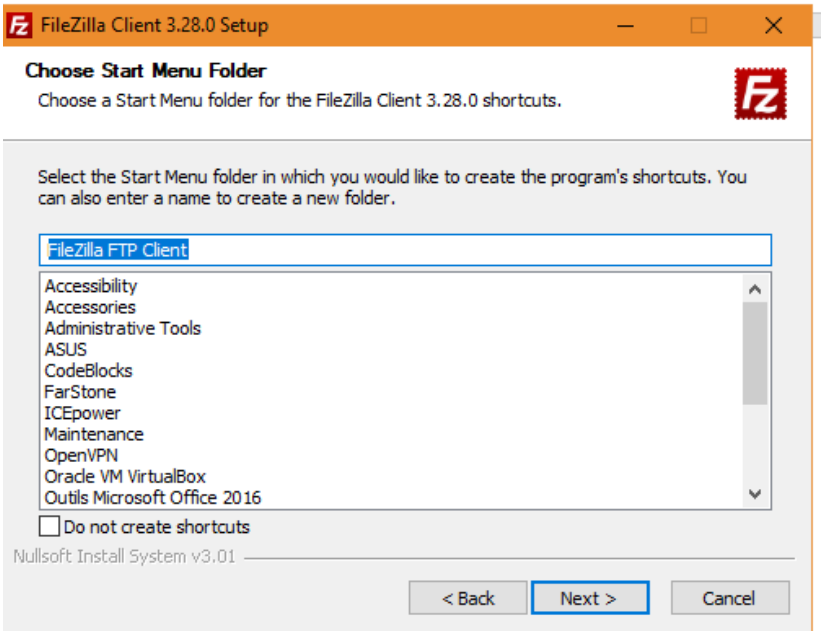


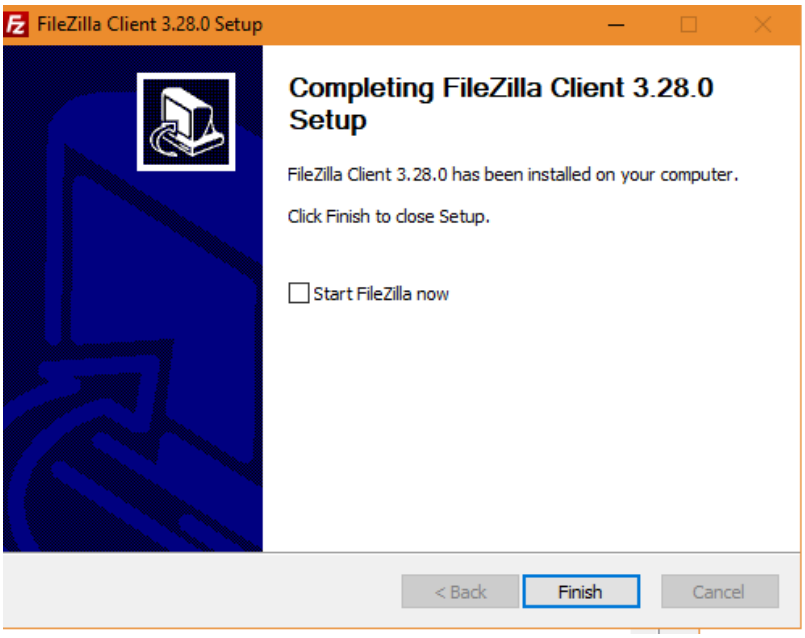
ccept the terms of this agreement and the privacy policy, click I Agree to continue.



Raphael Andrieu

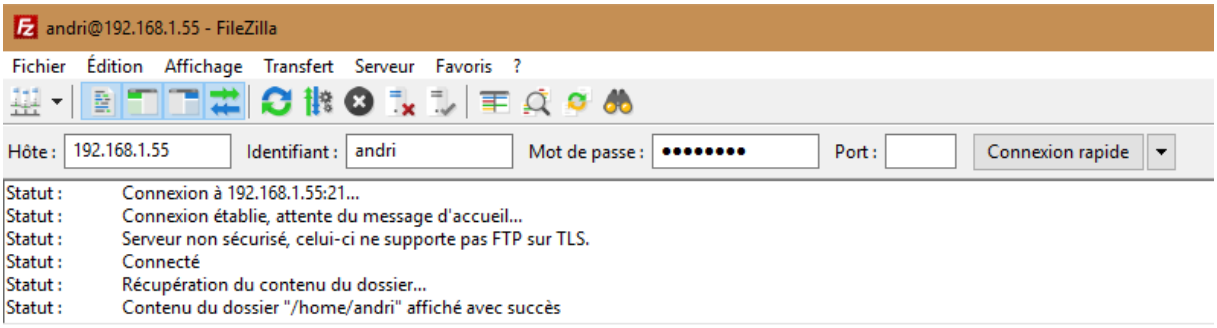




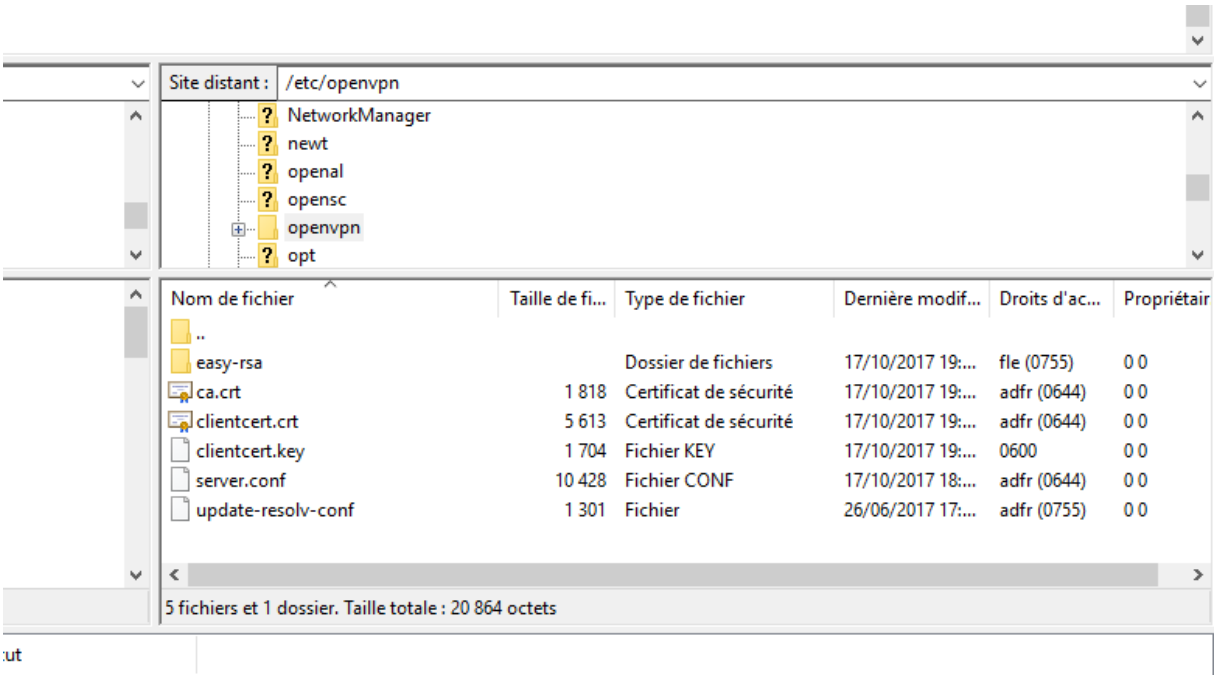


Récupération des certificats par FTP

Après avoir bien installé FileZilla, lancez-le et connectez-vous avec l'adresse IP du Serveur Debian et vos identifiants. (Le root ne fonctionne pas en FTP)



Puis accédez au dossier /etc/openvpn



Récupérez les fichiers ca.crt, clientcert.crt et clientcert.key que nous avons copiés tout a l'heure dans ce dossier.

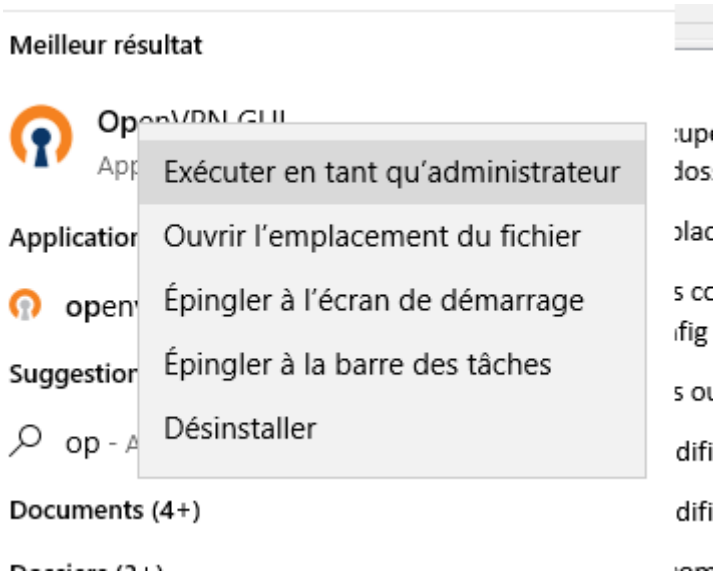
Déplacez ces fichiers dans le dossier config de OpenVPN. « C:\Program Files\OpenVPN\config »

Puis copier le fichier client.ovpn du dossier sample-config « C:\Program Files\OpenVPN\sample-config » dans le dossier de config.

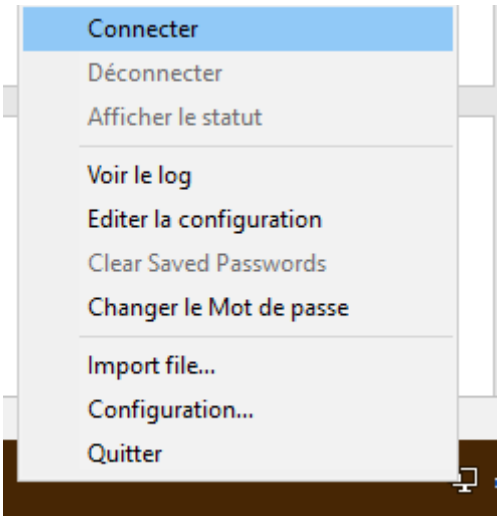
Puis avec un éditeur de texte modifiez la ligne « remote my-server-1 1194 » par « remote 192.168.1.55 1194 », ensuite modifiez le nom des certificats client.key et client.crt par clientcert.key et clientcert.crt et commentez la ligne « tls-auth ta.key 1 » et « cipher AES-256-CBC » si ces paramètres ne sont pas activés sur le serveur dans le fichier server.conf

Lancement du VPN sur le client

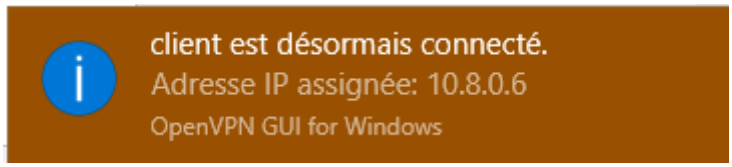
Il ne vous reste plus qu'à lancer OpenVPN en administrateur



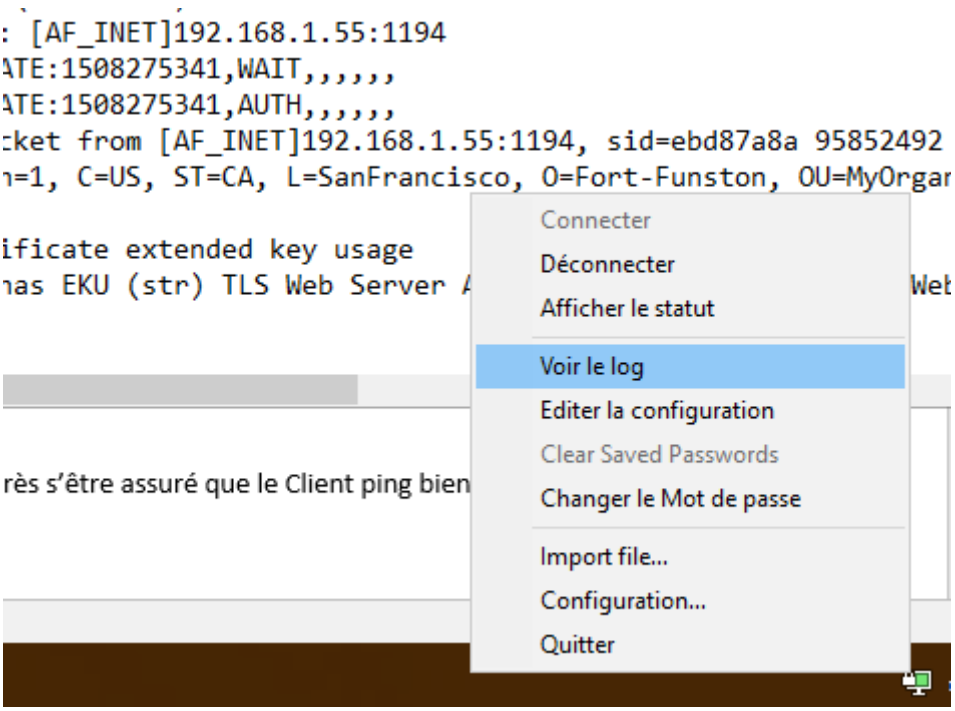
Et à lancer la connexion depuis la barre des taches



Si tout se passe bien, le client devrait se lancer correctement

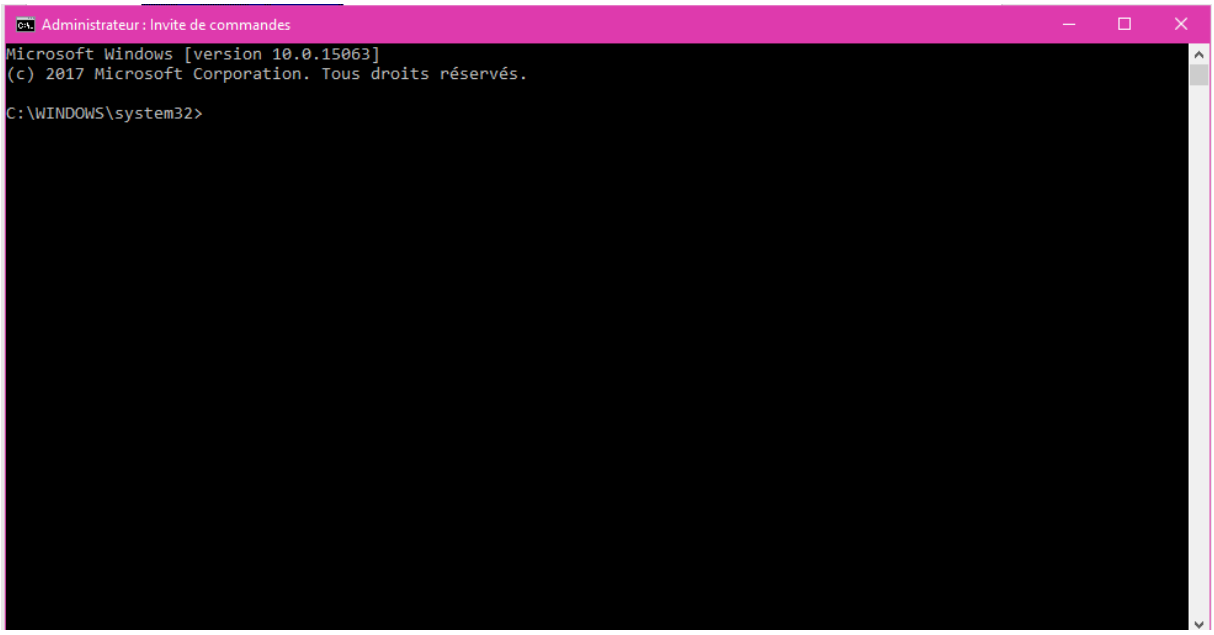


Sinon, vous pouvez vérifier les logs pour corriger votre problème en faisant clic droit et « voir le log »



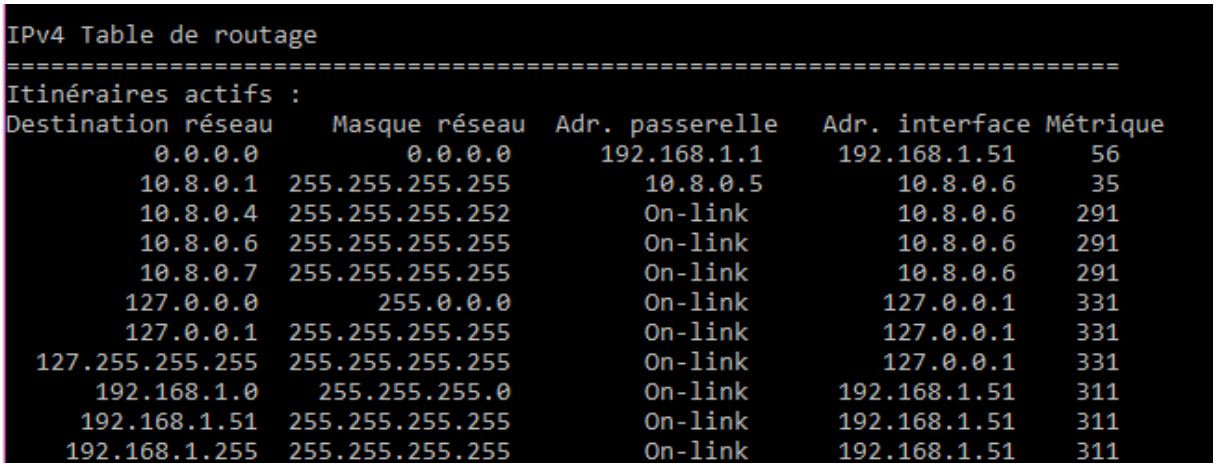
Accéder à internet depuis le Client Windows

Après avoir lancé le serveur VPN et après s'être assuré que le Client ping bien le serveur VPN, il faut mettre en place la passerelle du VPN, ouvrez donc un CMD en administrateur



Et tapez la commande :

- route print



La table de routage apparait, tout en haut nous pouvons voir une route vers le réseau 0.0.0.0 en utilisant la passerelle 192.168.1.1 par l'interface 192.168.1.51. cette route utilise le routeur du LAN pour aller sur Internet et non le serveur OPENVPN. Nous allons enlever cette route et mettre en place une route utilisant la passerelle du VPN.

Pour cela il faut taper la commande :

Route delete 0.0.0.0.0

Si vous refaites « route print » vous pouvez voir que la route a été supprimée.

```
IPv4 Table de routage
===========================================================================
Itinéraires actifs :
Destination réseau    Masque réseau  Adr. passerelle   Adr. interface Métrique
        127.0.0.0        255.0.0.0         On-link         127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link         127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link         127.0.0.1    331
      192.168.1.0    255.255.255.0         On-link       192.168.1.51    311
     192.168.1.51  255.255.255.255         On-link       192.168.1.51    311
    192.168.1.255  255.255.255.255         On-link       192.168.1.51    311
        224.0.0.0        240.0.0.0         On-link         127.0.0.1    331
        224.0.0.0        240.0.0.0         On-link       192.168.1.51    311
  255.255.255.255  255.255.255.255         On-link         127.0.0.1    331
  255.255.255.255  255.255.255.255         On-link       192.168.1.51    311
===========================================================================
```

Nous allons maintenant ajouter une autre route en écrivant :

Route add 0.0.0.0.0 mask 255.255.255.0 10.8.0.5

Cette ligne ajoute une route vers le réseau 0.0.0.0 avec comme masque 255.255.255.0 et en utilisant comme passerelle 10.8.0.5 (passerelle du VPN)

Vous pouvez vérifier que cette ligne a bien été ajouté en retapant la commande « route print »

```
IPv4 Table de routage
===========================================================================
Itinéraires actifs :
Destination réseau    Masque réseau  Adr. passerelle   Adr. interface Métrique
          0.0.0.0          0.0.0.0        10.8.0.5        10.8.0.6     36
         10.8.0.1  255.255.255.255        10.8.0.5        10.8.0.6     35
         10.8.0.4  255.255.255.252         On-link        10.8.0.6    291
         10.8.0.6  255.255.255.255         On-link        10.8.0.6    291
         10.8.0.7  255.255.255.255         On-link        10.8.0.6    291
        127.0.0.0        255.0.0.0         On-link       127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link       127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link       127.0.0.1    331
      192.168.1.0    255.255.255.0         On-link     192.168.1.51    316
     192.168.1.51  255.255.255.255         On-link     192.168.1.51    316
    192.168.1.255  255.255.255.255         On-link     192.168.1.51    316
```

Vous pouvez maintenant faire un « ipconfig » et voir que la passerelle a bien été ajouté dans la passerelle par défaut de la carte réseau de Openvpn

```
Carte Ethernet Ethernet 3 :

   Suffixe DNS propre à la connexion. . . :
   Adresse IPv6 de liaison locale. . . . .: fe80::2027:7eb2:132c:21e5%41
   Adresse IPv4. . . . . . . . . . . . . .: 10.8.0.6
   Masque de sous-réseau. . . . . . . . . : 255.255.255.252
   Passerelle par défaut. . . . . . . . . : 10.8.0.5
```

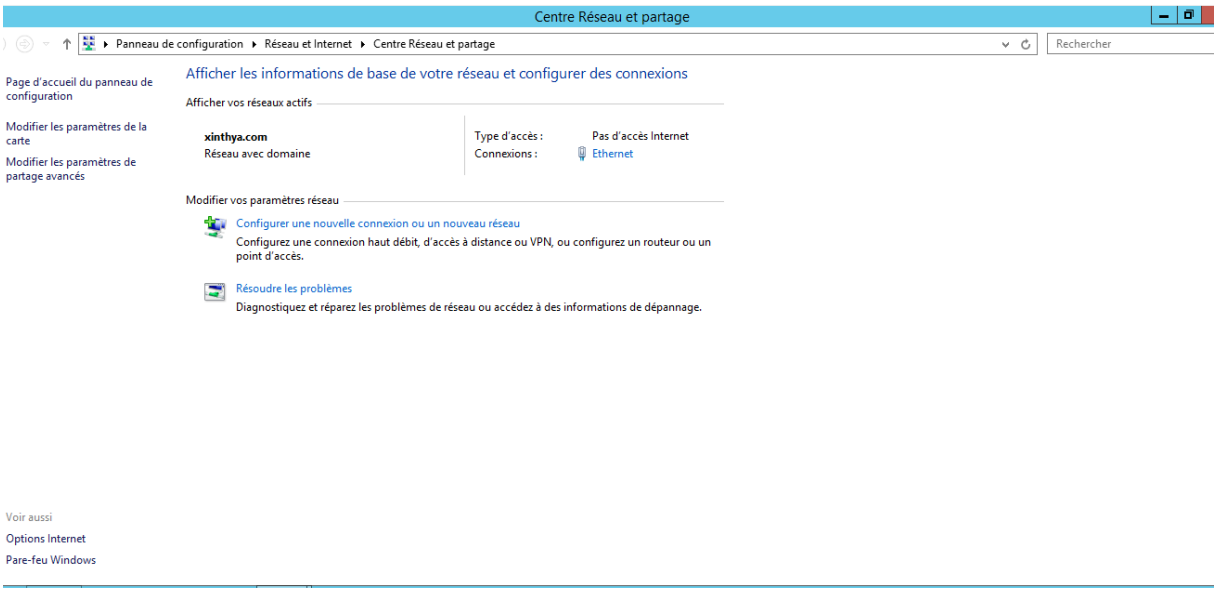
Et vous pouvez tester la connexion vers internet avec « ping 8.8.8.8 »

```
C:\WINDOWS\system32>ping 8.8.8.8

Envoi d'une requête 'Ping'  8.8.8.8 avec 32 octets de données :
Réponse de 8.8.8.8 : octets=32 temps=142 ms TTL=250
Réponse de 8.8.8.8 : octets=32 temps=216 ms TTL=250
Réponse de 8.8.8.8 : octets=32 temps=191 ms TTL=250
Réponse de 8.8.8.8 : octets=32 temps=133 ms TTL=250

Statistiques Ping pour 8.8.8.8:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 133ms, Maximum = 216ms, Moyenne = 170ms

C:\WINDOWS\system32>
```

Accéder avec le poste client au Serveur Windows du réseau distant

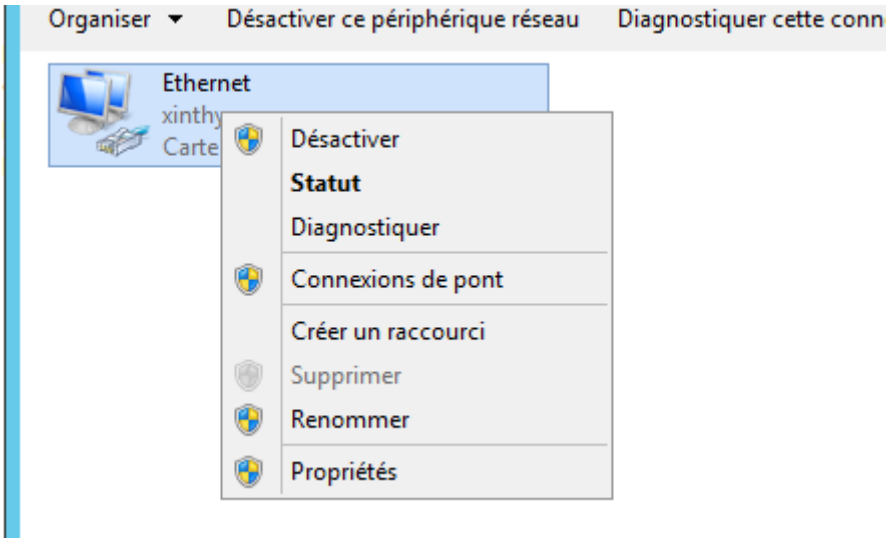
Pour accéder au serveur Windows du Lan avec le poste Client OpenVpn, il vous faut ajouter une passerelle au serveur Windows.

Pour cela il vous faut aller sur « centre réseau et partage »

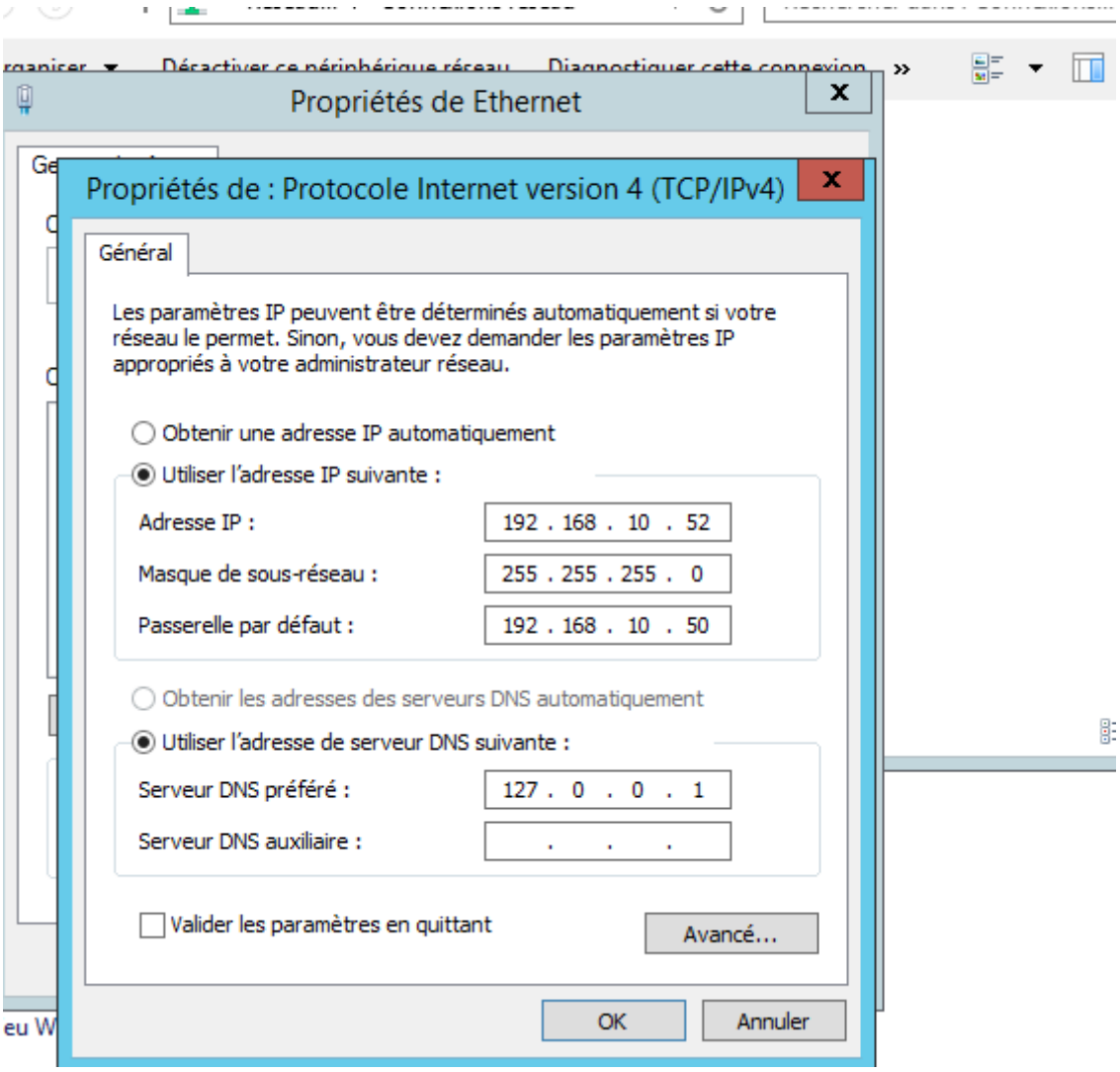


Puis cliquer sur « Modifier les paramètres de la carte »

Raphael Andrieu



Clic droit sur la carte réseau utilisée puis sur propriétés. En arrivant sur la fenêtre de propriétés, modifier les paramètres IPV4 en mettant une adresse fixe et en passerelle : l'adresse IP de Eth1 du serveur Debian.

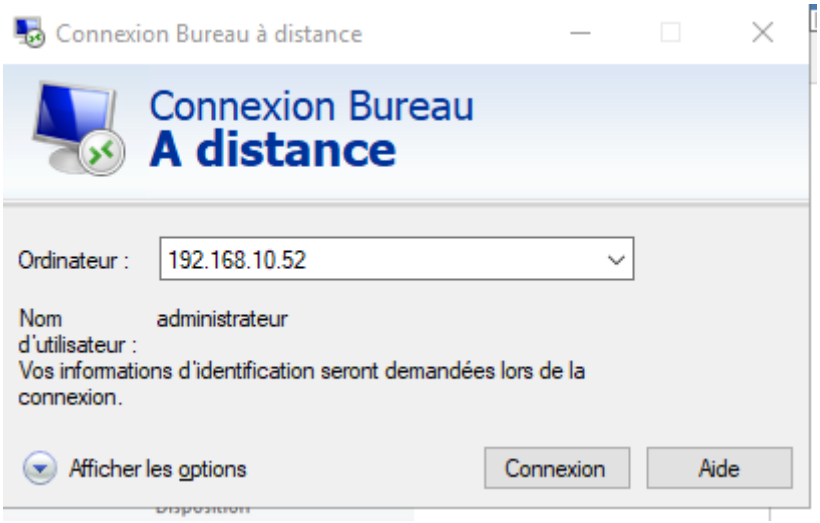


Vous pouvez maintenant essayer de Ping le Serveur Windows depuis le Client :

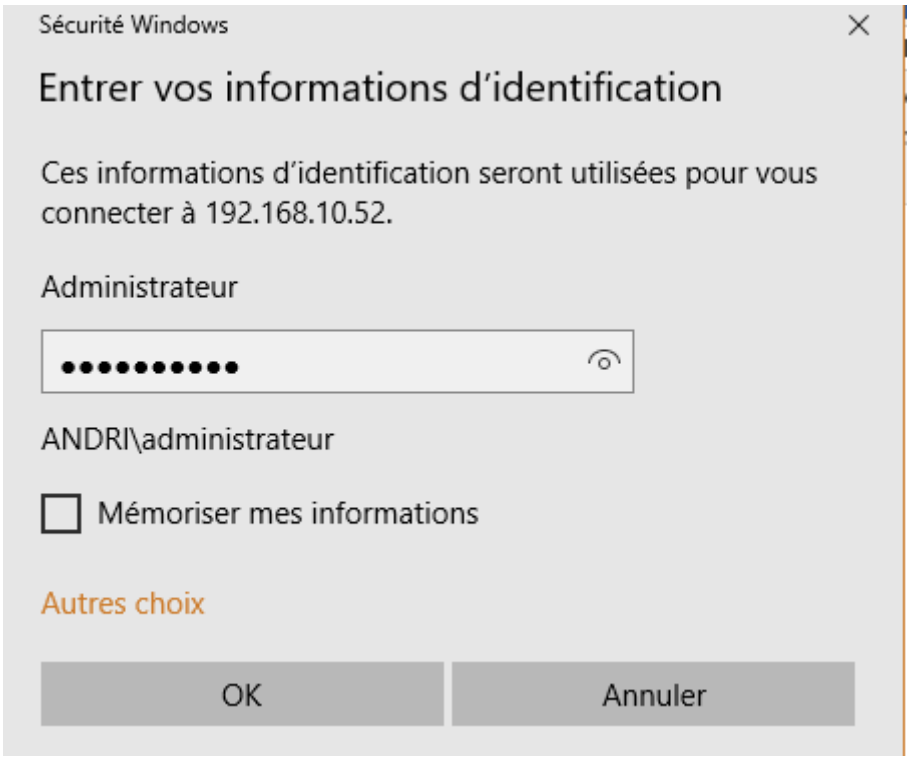
```
C:\WINDOWS\system32>ping 192.168.10.52

Envoi d'une requête 'Ping'  192.168.10.52 avec 32 octets de données :
Réponse de 192.168.10.52 : octets=32 temps=60 ms TTL=127
Réponse de 192.168.10.52 : octets=32 temps=65 ms TTL=127
Réponse de 192.168.10.52 : octets=32 temps=9 ms TTL=127
Réponse de 192.168.10.52 : octets=32 temps=6 ms TTL=127

Statistiques Ping pour 192.168.10.52:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 6ms, Maximum = 65ms, Moyenne = 35ms
```

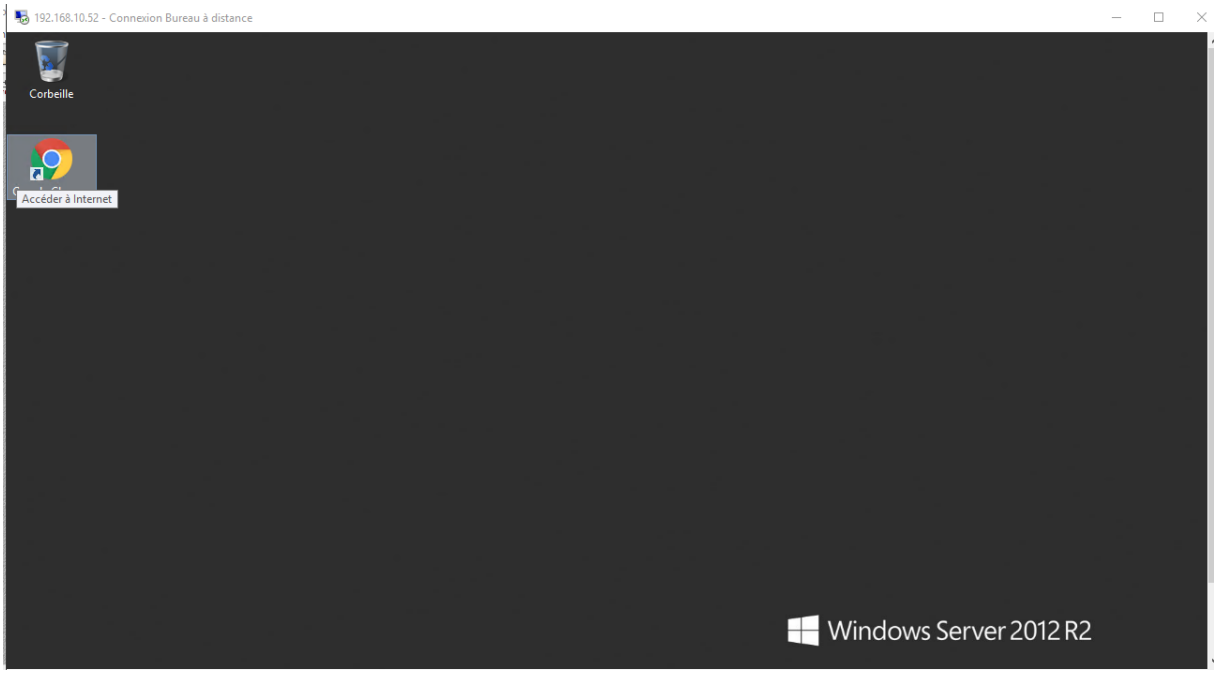
Maintenant que le ping est possible, vous pouvez vous connecter sur le serveur Windows avec « connexion bureau a distance » et accéder au dossier de l'entreprise « \\192.168.10.52\xinthya »

Connexion bureau à distance sur le serveur Windows 2012



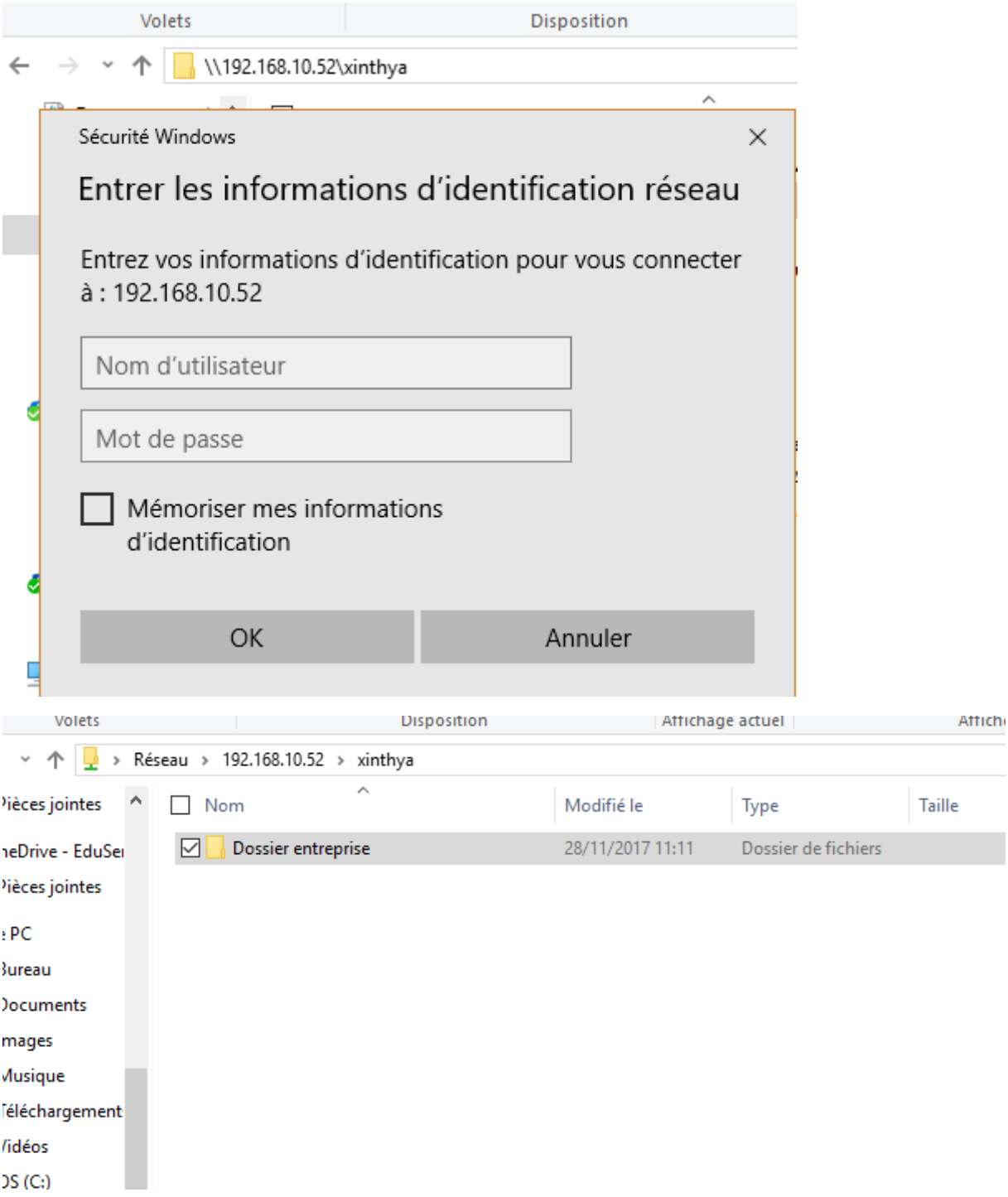
Raphael Andrieu





Raphael Andrieu

Accès au dossier de l'entreprise sur le serveur de fichier Windows 2012



Et voilà, vous accédez bien au serveur de fichier Windows !