

RAPPORT GÉNÉRAL SUR LE VPN (OPENVPN)

Présenté par :

Berenger Benam

Encadrant :

Mr Bessan

Table des matières

Introduction :	1
Prérequis :	2
Étape 1 - Mettez à jour votre système	2
Client Linux	10
Installation	10
Coté Serveur :	12
Client Windows :	13
Installation	13
iOS (iPhone)	23
Installation	24
Conclusion :	29

Introduction :

Un réseau privé virtuel (VPN) vous permet de traverser des réseaux non fiables comme si vous étiez sur un réseau privé. Il vous donne la liberté d'accéder à l'internet en toute sécurité depuis votre smartphone ou votre ordinateur portable lorsque vous êtes connecté à un réseau non sécurisé, comme le Wifi d'un hôtel ou d'un café.

OpenVPN est un VPN SSL (réseau privé virtuel) complet. Il implémente l'extension de réseau sécurisé de couche 2 ou 3 OSI à l'aide du protocole SSL/TLS. C'est un logiciel open source et distribué sous licence GNU GPL. Un VPN vous permet de vous connecter en toute sécurité à un réseau public non sécurisé tel qu'un réseau wifi à l'aéroport ou à l'hôtel. Un VPN est également requis pour accéder aux ressources de votre serveur d'entreprise ou d'entreprise ou domestique. Vous pouvez contourner le site géo bloqué et augmenter votre confidentialité ou votre sécurité en ligne. Ce didacticiel fournit des instructions étape par étape pour configurer un serveur OpenVPN sur le serveur Ubuntu Linux 18.04 LTS , puis le configurer pour qu'il soit accessible depuis une machine cliente.

Prérequis :

Installer les paquets suivants et assurez-vous de mettre à jour le noyau de votre distribution Linux pour avoir une version au moins supérieure ou égale à 5.3.0.x

Vérifier la version du noyau de votre distribution Linux par la commande :

```
root@berenger:~# uname -rm
5.4.0-113-generic x86_64
root@berenger:~#
```

Étape 1 - Mettez à jour votre système

Exécutez la commande :

```
root@berenger:~# apt-get update && upgrade
Atteint :2 http://sn.archive.ubuntu.com/ubuntu bionic InRelease
Atteint :3 http://ppa.launchpad.net/ansible/ansible/ubuntu bionic InRelease
Réception de :1 http://security.ubuntu.com/ubuntu bionic-security InRelease [88,7 kB]
Réception de :4 http://sn.archive.ubuntu.com/ubuntu bionic-updates InRelease [88,7 kB]
Réception de :5 http://sn.archive.ubuntu.com/ubuntu bionic-backports InRelease [74,6 kB]
Réception de :6 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [2279 kB]
45% [6 Packages 1733 kB/2279 kB 76%]

root@berenger:~# apt-get install --install-recommends linux-generic-hwe-18.04 xserver-xorg-hwe-18.04
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
xserver-xorg-hwe-18.04 est déjà la version la plus récente (1:7.7+19ubuntu8~18.04.3).
xserver-xorg-hwe-18.04 passé en « installé manuellement ».
Les paquets supplémentaires suivants seront installés :
  linux-headers-5.4.0-117-generic linux-headers-generic-hwe-18.04 linux-hwe-5.4-headers-5.4.0-117
  linux-image-5.4.0-117-generic linux-image-generic-hwe-18.04 linux-modules-5.4.0-117-generic
  linux-modules-extra-5.4.0-117-generic
Paquets suggérés :
  fdutils linux-hwe-5.4-doc-5.4.0 | linux-hwe-5.4-source-5.4.0 linux-hwe-5.4-tools
Les NOUVEAUX paquets suivants seront installés :
  linux-headers-5.4.0-117-generic linux-hwe-5.4-headers-5.4.0-117 linux-image-5.4.0-117-generic
  linux-modules-5.4.0-117-generic linux-modules-extra-5.4.0-117-generic
Les paquets suivants seront mis à jour :
  linux-generic-hwe-18.04 linux-headers-generic-hwe-18.04 linux-image-generic-hwe-18.04
3 mis à jour, 5 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 76,4 Mo dans les archives.
Après cette opération, 368 Mo d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [0/n]
```

On installe tous les dépendances.

```
root@berenger:~# apt-get install linux-headers-$(uname -r)
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
linux-headers-5.4.0-113-generic est déjà la version la plus récente (5.4.0-113.127~18.04.1).
linux-headers-5.4.0-113-generic passé en « installé manuellement ».
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  linux-headers-5.4.0-110-generic linux-hwe-5.4-headers-5.4.0-110 linux-image-5.4.0-110-generic
  linux-modules-5.4.0-110-generic linux-modules-extra-5.4.0-110-generic
Veuillez utiliser « apt autoremove » pour les supprimer.
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@berenger:~#
```

On confirme la version du noyau

```

root@berenger:~# apt-get install resolvconf
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  linux-headers-5.4.0-110-generic linux-hwe-5.4-headers-5.4.0-110 linux-image-5.4.0-110-generic
  linux-modules-5.4.0-110-generic linux-modules-extra-5.4.0-110-generic
Veuillez utiliser « apt autoremove » pour les supprimer.
Les NOUVEAUX paquets suivants seront installés :
  resolvconf
0 mis à jour, 1 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 48,0 ko dans les archives.
Après cette opération, 187 ko d'espace disque supplémentaires seront utilisés.
Réception de :1 http://sn.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 resolvconf all 1.79ubuntu
10.18.04.3 [48,0 kB]
48,0 ko réceptionnés en 2s (25,6 ko/s)
Préconfiguration des paquets...

```

On installe le paquet **resolvconf** pour gérer l'IP du serveur via le réseau.

```

root@berenger:~# git clone https://github.com/pivpn/pivpn.git
Clonage dans 'pivpn'...
remote: Enumerating objects: 3677, done.
remote: Counting objects: 100% (500/500), done.
remote: Compressing objects: 100% (188/188), done.
remote: Total 3677 (delta 297), reused 417 (delta 268), pack-reused 3177
Réception d'objets: 100% (3677/3677), 1.93 MiB | 1.16 MiB/s, fait.
Résolution des deltas: 100% (2157/2157), fait.
root@berenger:~#

```

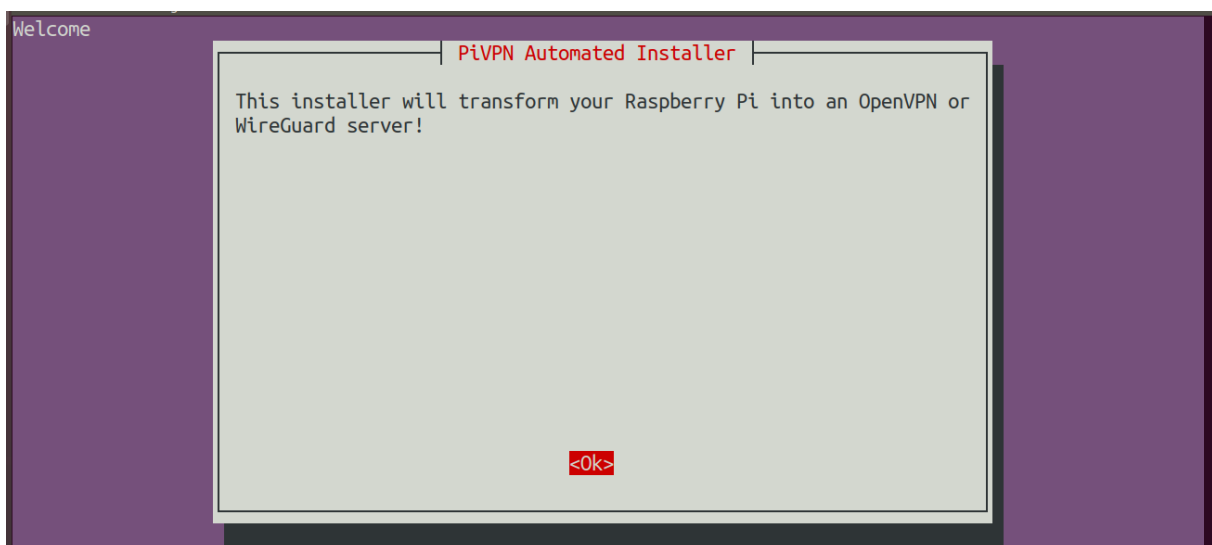
Téléchargez le script d'installation automatique :

```

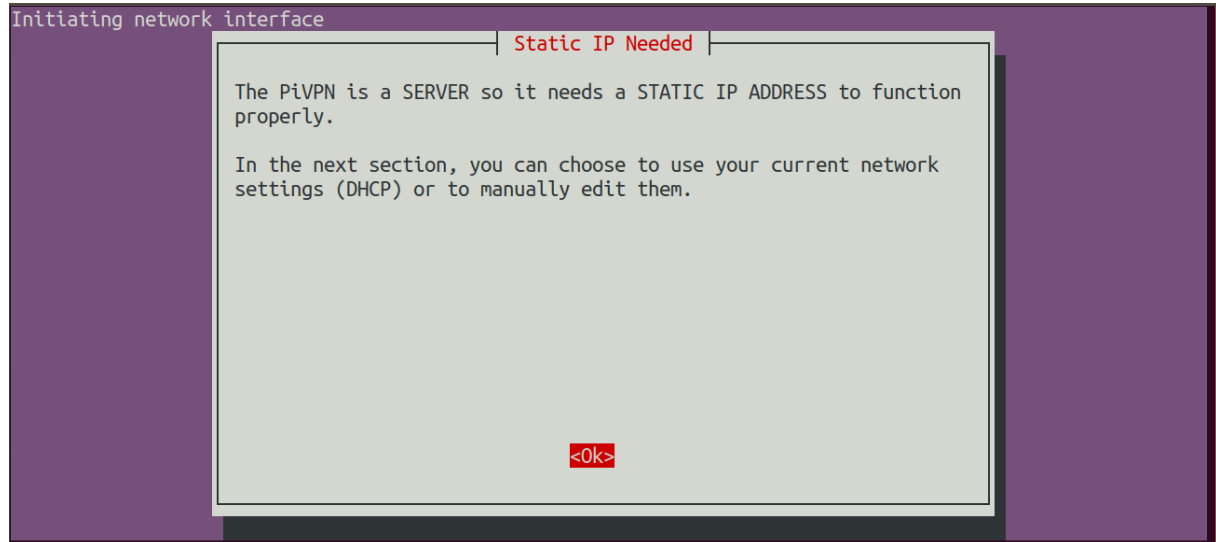
root@berenger:~# ./pivpn/auto_install/install.sh
:::
::: You are root.
::: Hostname length OK
::: Verifying free disk space...
:::
::: Package Cache update is needed, running apt-get update -y ...
[ \ ]

```

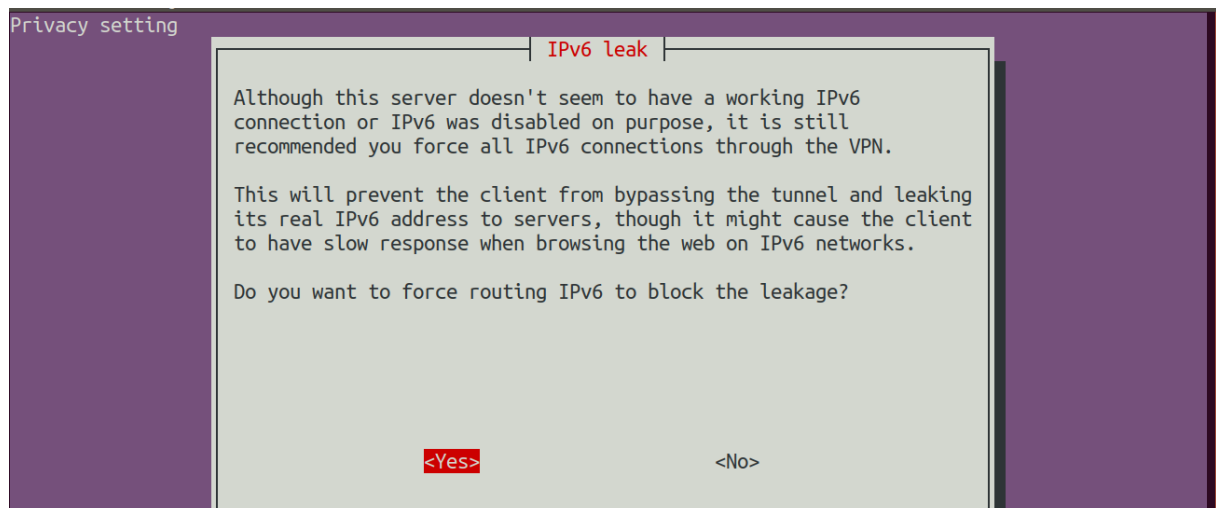
On installe le script et paramétrer OpenVPN



On clique sur Ok



On valide



On force IPv6 toujours l'option **Yes**

Parsing User List

Local Users

Choose a local user that will hold your ovpn configurations.

<Ok>

IP Information

IP Information

Since we think you are not using Raspberry Pi OS, we will not configure a static IP for you.
If you are in Amazon then you can not configure a static IP anyway. Just ensure before this installer started you had set an elastic IP on your instance.

<Ok>

Fichier Édition Affichage Recherche Terminal Aide

Choose A User

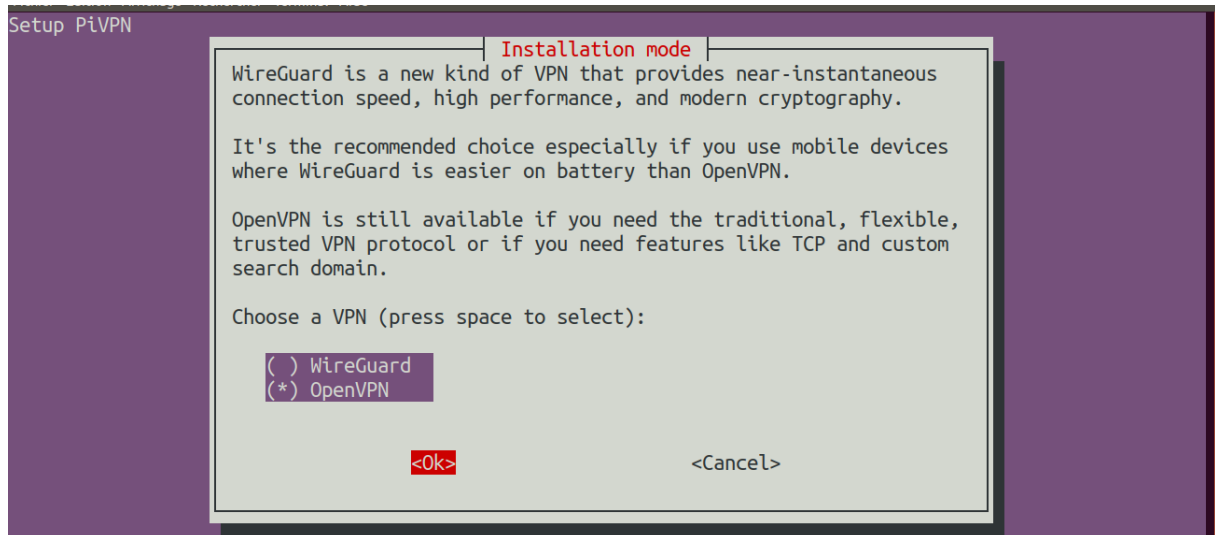
Choose (press space to select):

(*) berenger
() jenkins

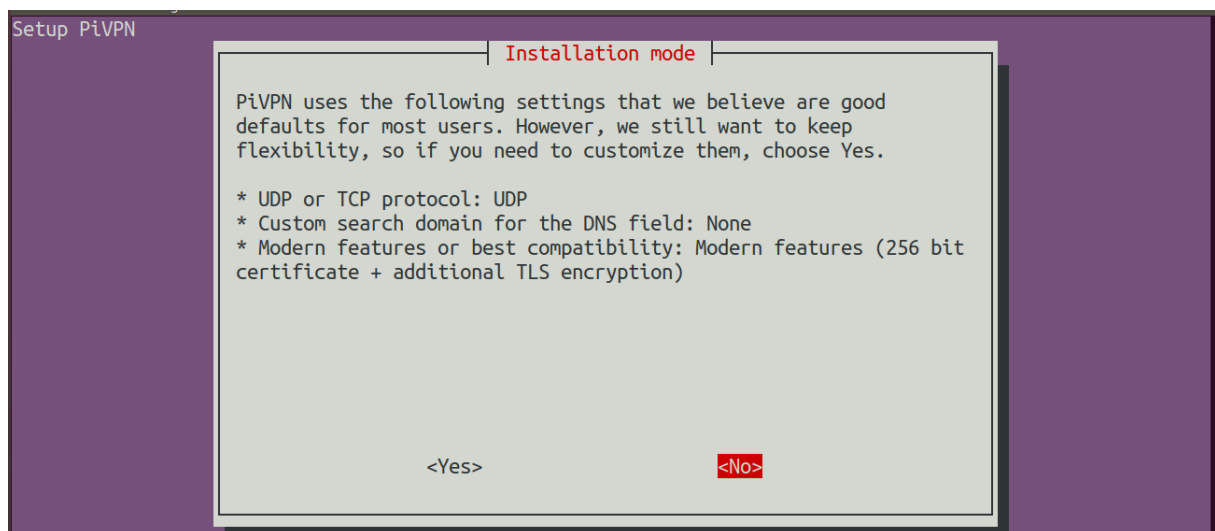
<Ok>

<Cancel>

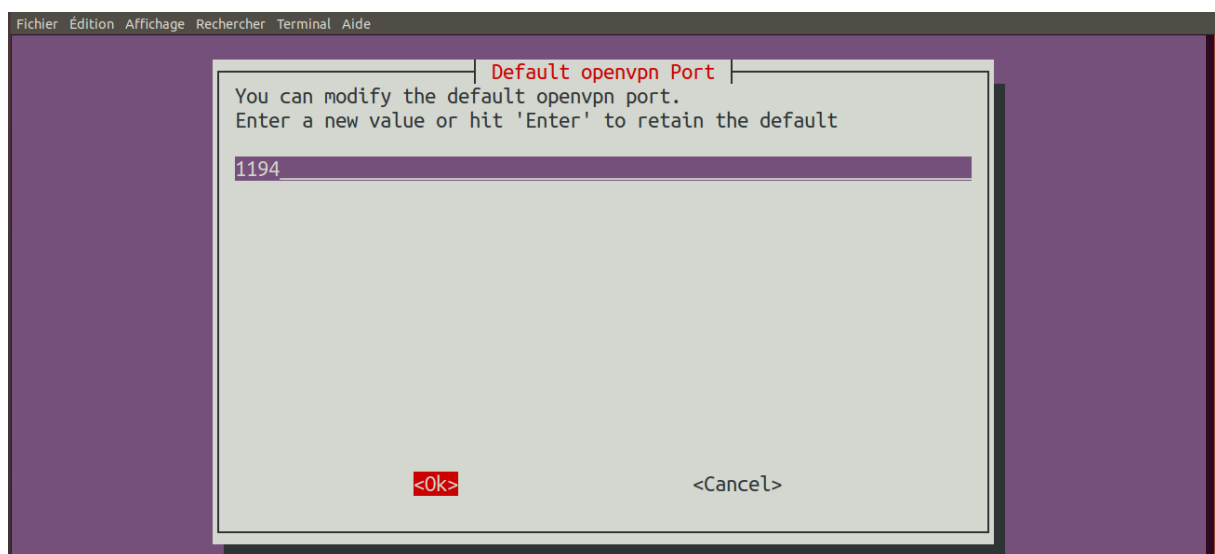
On choisit l'utilisateur qui va administrer l'OpenVPN.



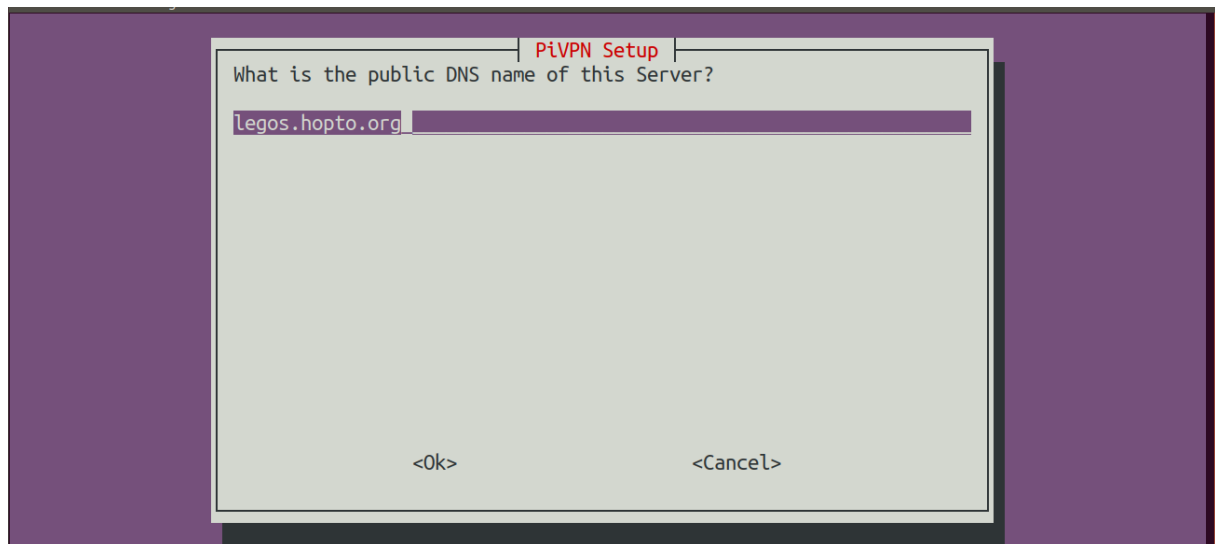
Choisir le type de VPN à mettre en place moi je veux paramétrer **OpenVPN**.



Il faut choisir **<No>** pour qu'il prenne en compte les deux Protocoles **UDP /TLS**



Par défaut **OpenVPN** utilise le port **1194**



On donne un nom de domaine.



Ok



Yes pour activer le serveur



Si vous arrivez a cette dernière capture cela veut-dire le paramétrage marche Nickel et Il faut choisir **Yes** pour qu'il redémarre la machine.

On lance la commande **pivpn -a**


```

root@berenger:~# pivpn -a
Enter a Name for the Client: berenger
How many days should the certificate last? 1080
Enter the password for the client:
Enter the password again to verify:
spawn ./easyrsa build-client-full berenger
* Notice:
Using Easy-RSA configuration from: /etc/openvpn/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 1.1.1 11 Sep 2018

Generating an EC private key
writing new private key to '/etc/openvpn/easy-rsa/pki/8846c3be/temp.a7ef74de'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
* Notice:

Keypair and certificate request completed. Your files are:
req: /etc/openvpn/easy-rsa/pki/reqs/berenger.req
key: /etc/openvpn/easy-rsa/pki/private/berenger.key

Using configuration from /etc/openvpn/easy-rsa/pki/safesl-easyrsa.cnf.init-tmp
Check that the request matches the signature

```

```

Signature ok
The Subject's Distinguished Name is as follows
commonName      :ASN.1 12:'berenger'
Certificate is to be certified until May 23 13:02:53 2025 GMT (1080 days)

Write out database with 1 new entries
Data Base Updated

* Notice:
Certificate created at: /etc/openvpn/easy-rsa/pki/issued/berenger.crt

Client's cert found: berenger.crt
Client's Private Key found: berenger.key
CA public Key found: ca.crt
tls Private Key found: ta.key

=====
Done! berenger.ovpn successfully created!
berenger.ovpn was copied to:
/home/berenger/ovpns
for easy transfer. Please use this profile only on one
device and create additional profiles for other devices.
=====

```

Cette commande nous permet de créer un utilisateur **berenger** et préciser son mot de passe. et à la fin de création de user **berenger** cette commande crée en même temps un dossier **ovpn** qui contient la clé générée par l'utilisateur **berenger**.

```

root@berenger:~# ls
Bureau      Documents  lien.txt   ovpns      Public      Vidéos
ca-cert.pem examples.desktop Modèles    pivpn      'Rapport_Ansible&Jenkins_Berenger.pdf'
ca.pem      Images     Musique    pki         Téléchargements
root@berenger:~# ls ovpns/
berenger.ovpn
root@berenger:~#

```

Sur cette capture on remarque que la commande génère la clé qui porte le nom de l'utilisateur par l'extension **.ovpn**

On se place dans le dossier **/etc/openvpn/** et lancer la commande **grep -r logos.hopto.org :**

```

root@berenger:/etc/openvpn# grep -r logos.hopto.org
easy-rsa/pki/Default.txt:remote logos.hopto.org 1194
easy-rsa/pki/berenger.ovpn:remote logos.hopto.org 1194
root@berenger:/etc/openvpn#

```

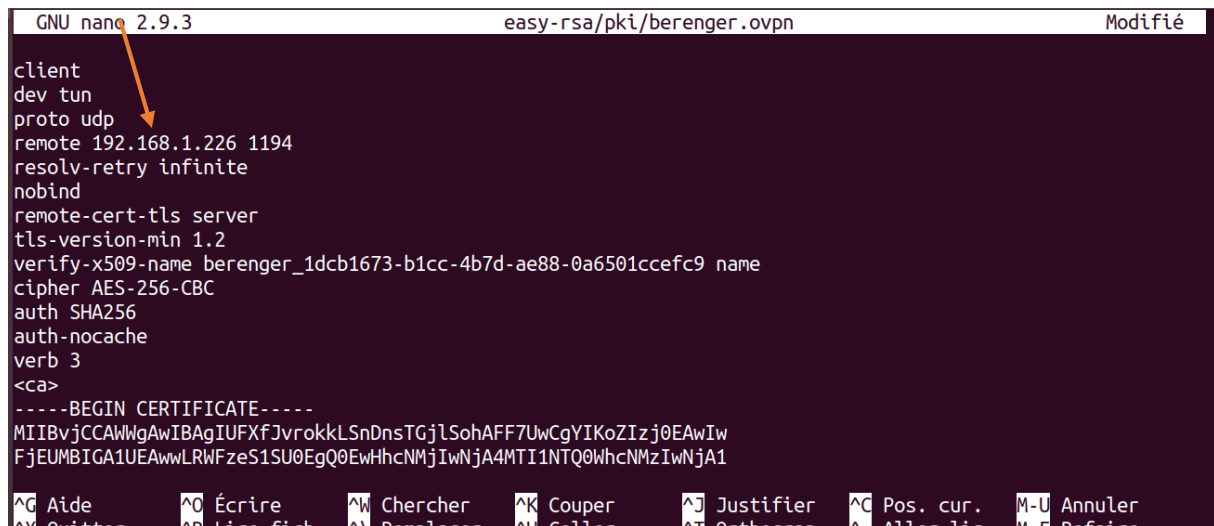
logos.hopto.org c'est le nom de domaine que j'ai choisi lors de l'installation d'OpenVPN.

```

root@berenger:/etc/ovpn# grep -r logos.hopto.org
easy-rsa/pki/Default.txt:remote logos.hopto.org 1194
easy-rsa/pki/berenger.ovpn:remote logos.hopto.org 1194
root@berenger:/etc/ovpn# ls
ccd client crt.pem easy-rsa server server.conf update-resolv-conf
root@berenger:/etc/ovpn# nano easy-rsa/pki/berenger.ovpn
root@berenger:/etc/ovpn#

```

On édite le fichier **easy-rsa/pki/berenger.ovpn** pour préciser l'IP du serveur.



```

GNU nano 2.9.3 easy-rsa/pki/berenger.ovpn Modifié
client
dev tun
proto udp
remote 192.168.1.226 1194
resolv-retry infinite
nobind
remote-cert-tls server
tls-version-min 1.2
verify-x509-name berenger_1dcb1673-b1cc-4b7d-ae88-0a6501ccef9 name
cipher AES-256-CBC
auth SHA256
auth-nocache
verb 3
<ca>
-----BEGIN CERTIFICATE-----
MIIBvjCCAWWgAwIBAgIUFXFJvrokKLSnDnsTGjLSohAFF7UwCgYIKoZIZj0EAWIw
FjEUMBIGA1UEAwLRWFzeS1SU0EgQ0EwHhcNMjIwMTI1NTQ0WWhcNMzIwNjA1

```

On précise juste l'IP du serveur.

```

root@berenger:/etc/ovpn# service openvpn start
root@berenger:/etc/ovpn#

```

On redémarre le serveur.

```

root@berenger:~/ovpn# scp berenger.ovpn berenger@192.168.2.76:~/
The authenticity of host '192.168.2.76 (192.168.2.76)' can't be established.
ECDSA key fingerprint is SHA256:HDf59u8mzkVbQzLWKtmtpGMD7gZf8/ZGBD9sMdykw60.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.2.76' (ECDSA) to the list of known hosts.
berenger@192.168.2.76's password:
berenger.ovpn                                100% 2675   111.0KB/s   00:00
root@berenger:~/ovpn#

```

Il faut envoyer le certificat à un client Linux pour le Test.

Client Linux

Installation

Si vous utilisez Linux, vous pouvez utiliser plusieurs outils en fonction de votre distribution. Votre environnement de bureau ou votre gestionnaire de fenêtres peut également inclure des utilitaires de connexion.

La façon la plus universelle de se connecter, cependant, est d'utiliser simplement le logiciel OpenVPN.

Sur Ubuntu, vous pouvez l'installer comme vous l'avez fait sur le serveur en tapant :

```

root@berenger:~# apt-get install openvpn
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  gir1.2-geocodeglib-1.0 libegl1-mesa libpython-stdlib libwayland-egl1-mesa
  linux-hwe-5.4-headers-5.4.0-107 python python-minimal python2.7 python2.7-minimal
  ubuntu-web-launchers
Veuillez utiliser « apt autoremove » pour les supprimer.
Les paquets supplémentaires suivants seront installés :
  libpkcs11-helper1
Paquets suggérés :
  easy-rsa resolvconf
Les NOUVEAUX paquets suivants seront installés :
  libpkcs11-helper1 openvpn
0 mis à jour, 2 nouvellement installés, 0 à enlever et 54 non mis à jour.
Il est nécessaire de prendre 513 ko dans les archives.
Après cette opération, 1 275 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]

```

Une fois l'installation est terminée il faut éditer le certificat que le serveur a envoyé et préciser son IP .

```

root@berenger:~# ls
berenger.ovpn  Documents      Images  Musique  Téléchargements
Bureau         exemples.desktop  Modèles  Public   Vidéos
root@berenger:~#

```

```

GNU nano 2.9.3                                berenger.ovpn                                Modifié
client
dev tun
proto udp
remote 192.168.1.226 1194
resolv-retry infinite
nobind
remote-cert-tls server
tls-version-min 1.2
verify-x509-name berenger_1dcb1673-b1cc-4b7d-ae88-0a6501ccefc9 name
cipher AES-256-CBC
auth SHA256
auth-nocache
verb 3
<ca>
-----BEGIN CERTIFICATE-----
MIIBvjCCAWWgAwIBAgIUFXfJvrokKLSnDnsTGjLSohAFF7UwCgYIKoZIZj0EAwIw
FjEUMBIGA1UEAwLRWFzeS1SU0EgQ0EwHhcNMjIwNjA4MTI1NTQ0WWhcNMzIwNjA1
MTI1NTQ0WjAwMRQwEgYDVQDDAtFYXN5LVJTSBDQTBZMBMGByqGSM49AgEGCCqG
SM49AwEHA0IABI75aJ8LIX73PlDIardcv7D6jk30Sm10FSDx+1+7Vc0C0VNzKuqa
NyGktnH9bIT18kcBK+EBA170HgcfiNFyo9qjqZAwgY0wDAYDVR0TBAAUwAwEB/zAd

```

Voici l'adresse IP du serveur.

Pour se connecter le client doit taper la commande **openvpn --config berenger.ovpn** :

```

root@berenger:~# openvpn --config berenger.ovpn
Wed Jun  8 16:03:39 2022 OpenVPN 2.4.4 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKC
S11] [MH/PKTINFO] [AEAD] built on Mar 22 2022
Wed Jun  8 16:03:39 2022 library versions: OpenSSL 1.1.1 11 Sep 2018, LZO 2.08
Enter Private Key Password: *****

```

S'il lance cette commande on demande son mot de passe puis entrer le **mdp** du client.

```

Wed Jun  8 16:04:15 2022 Options error: Unrecognized option or missing or extra parameter(s) in [PU
SH-OPTIONS]:2: block-outside-dns (2.4.4)
Wed Jun  8 16:04:15 2022 OPTIONS IMPORT: timers and/or timeouts modified
Wed Jun  8 16:04:15 2022 OPTIONS IMPORT: --ifconfig/up options modified
Wed Jun  8 16:04:15 2022 OPTIONS IMPORT: route options modified
Wed Jun  8 16:04:15 2022 OPTIONS IMPORT: route-related options modified
Wed Jun  8 16:04:15 2022 OPTIONS IMPORT: --ip-win32 and/or --dhcp-option options modified
Wed Jun  8 16:04:15 2022 OPTIONS IMPORT: peer-id set
Wed Jun  8 16:04:15 2022 OPTIONS IMPORT: adjusting link_mtu to 1624
Wed Jun  8 16:04:15 2022 OPTIONS IMPORT: data channel crypto options modified
Wed Jun  8 16:04:15 2022 Data Channel: using negotiated cipher 'AES-256-GCM'
Wed Jun  8 16:04:15 2022 Outgoing Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
Wed Jun  8 16:04:15 2022 Incoming Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
Wed Jun  8 16:04:15 2022 ROUTE_GATEWAY 192.168.1.1/255.255.0.0 IFACE=enp0s3 HWADDR=08:00:27:34:db:4
5
Wed Jun  8 16:04:15 2022 TUN/TAP device tun0 opened
Wed Jun  8 16:04:15 2022 TUN/TAP TX queue length set to 100
Wed Jun  8 16:04:15 2022 do_ifconfig, tt->did_ifconfig_ipv6_setup=0
Wed Jun  8 16:04:15 2022 /sbin/ip link set dev tun0 up mtu 1500
Wed Jun  8 16:04:15 2022 /sbin/ip addr add dev tun0 10.195.230.2/24 broadcast 10.195.230.255
Wed Jun  8 16:04:15 2022 /sbin/ip route add 192.168.1.226/32 dev enp0s3
Wed Jun  8 16:04:15 2022 /sbin/ip route add 0.0.0.0/1 via 10.195.230.1
Wed Jun  8 16:04:15 2022 /sbin/ip route add 128.0.0.0/1 via 10.195.230.1
Wed Jun  8 16:04:15 2022 Initialization Sequence Completed

```

Sur cette capture le client arrive à se connecter au serveur super !

Côté Serveur :

```

TX packets 2746 bytes 509600 (509.6 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Boucle locale)
    RX packets 632 bytes 54290 (54.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 632 bytes 54290 (54.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
    inet 10.195.230.1 netmask 255.255.255.0 destination 10.195.230.1
    inet6 fe80::47ec:9c61:d3bd:c2e6 prefixlen 64 scopeid 0x20<link>
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 100 (UNSPEC)
    RX packets 12 bytes 678 (678.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 181 bytes 30482 (30.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@berenger:~/ovpn#

```

Si on fait **ifconfig** sur la machine serveur on remarque un tunnel s'est établi avec une adresse IP **10.195. 230.1**

Le serveur va faire un ping sur l'IP 10.195.230.1

```

root@berenger:~/ovpn# ping 10.195.230.1
PING 10.195.230.1 (10.195.230.1) 56(84) bytes of data.
64 bytes from 10.195.230.1: icmp_seq=1 ttl=64 time=0.026 ms
64 bytes from 10.195.230.1: icmp_seq=2 ttl=64 time=0.072 ms
64 bytes from 10.195.230.1: icmp_seq=3 ttl=64 time=0.056 ms
64 bytes from 10.195.230.1: icmp_seq=4 ttl=64 time=0.341 ms
64 bytes from 10.195.230.1: icmp_seq=5 ttl=64 time=0.059 ms
64 bytes from 10.195.230.1: icmp_seq=6 ttl=64 time=0.033 ms

```

On voit une réponse cool !

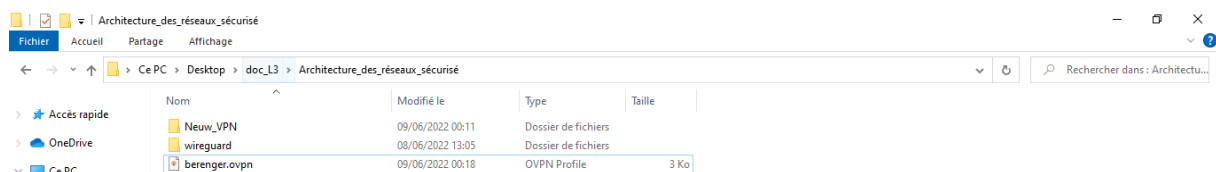
Donc le client Linux arrive à joindre la machine serveur OpenVPN.

Client Windows :

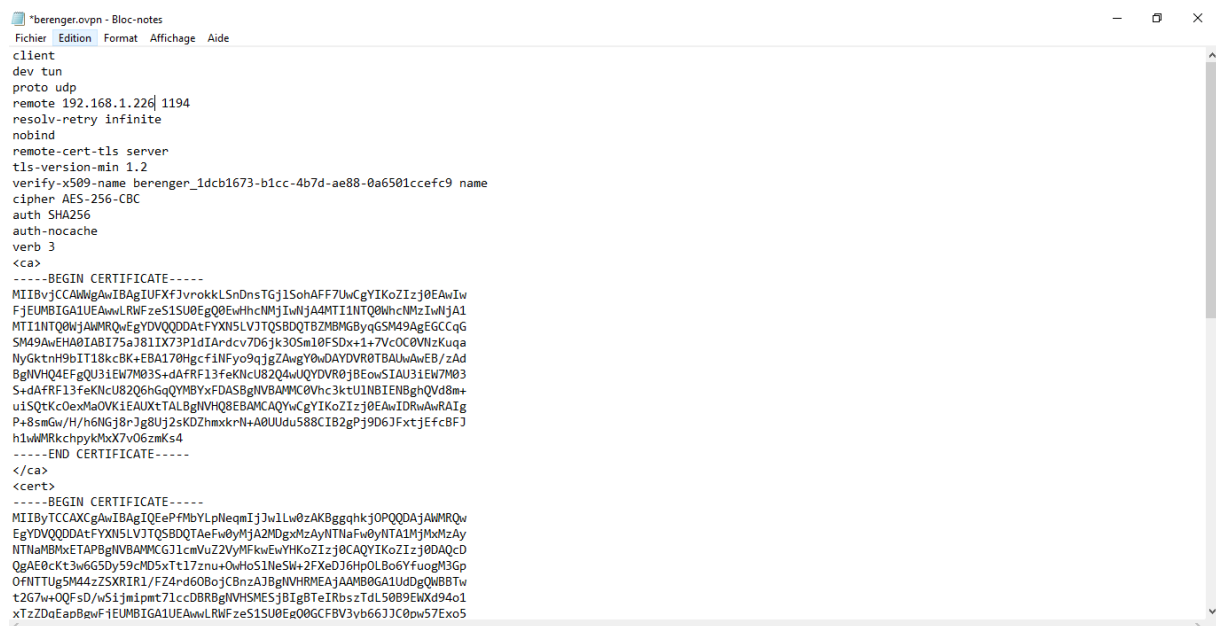
Cette section explique comment installer un profil VPN client sur Windows. Aucune de ces instructions du client ne dépend d'une autre, alors n'hésitez pas à passer à celle qui s'applique à votre appareil.

La connexion OpenVPN portera le même nom que celui du fichier **.ovpn**. Pour ce tutoriel, cela signifie que la connexion est nommée **berenger.ovpn**, s'alignant sur le premier fichier client que vous avez généré.

Il faut envoyer le certificat au niveau de la machine Client **Windows**.



Voici le certificat.

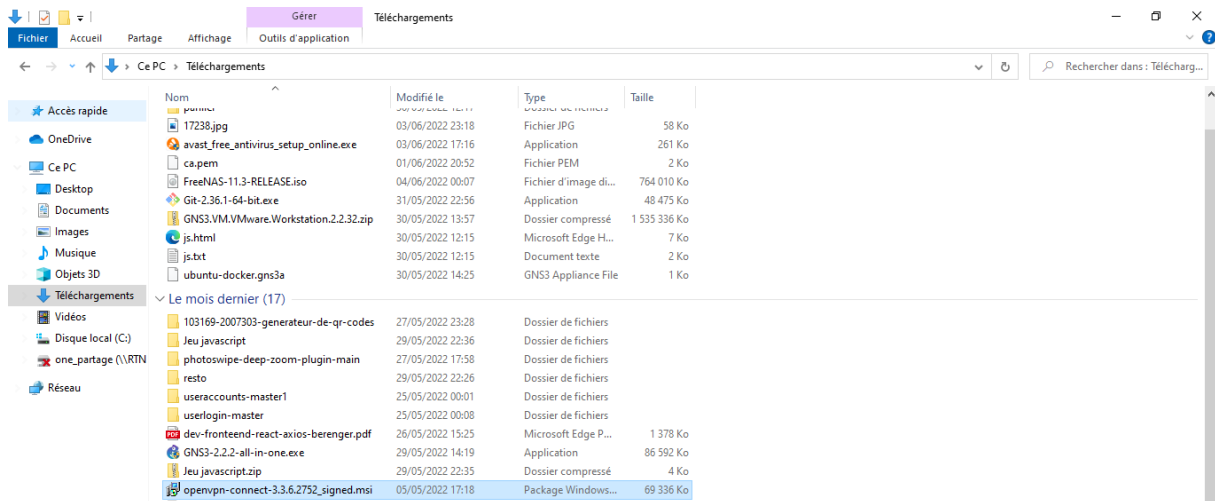


On renseigne l'adresse IP du serveur.

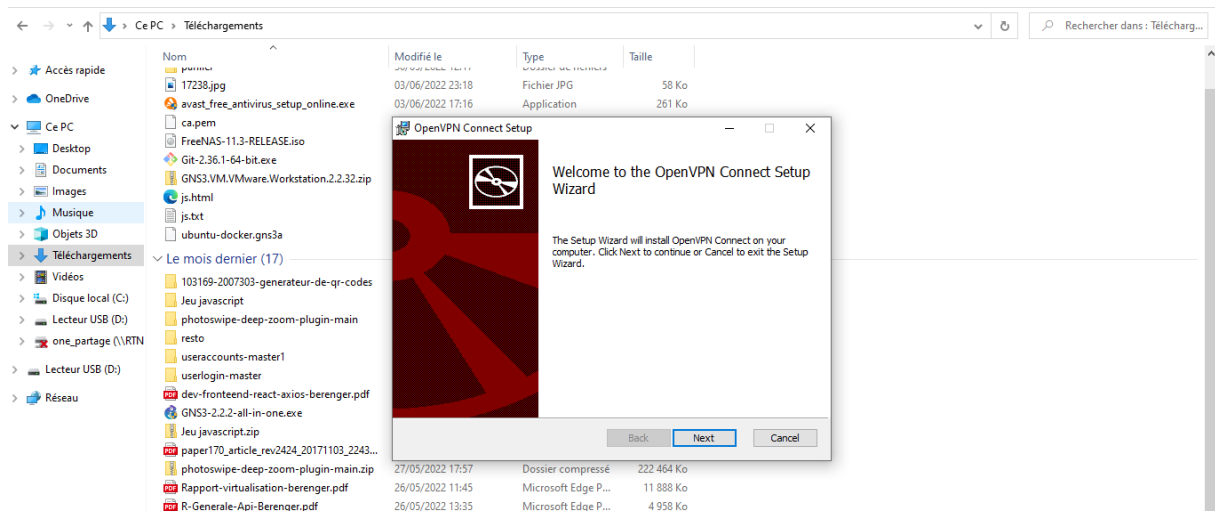
Installation

Téléchargez l'application client OpenVPN pour Windows depuis la [page de téléchargement d'OpenVPN](#). Choisissez la version d'installation appropriée pour votre version de Windows.

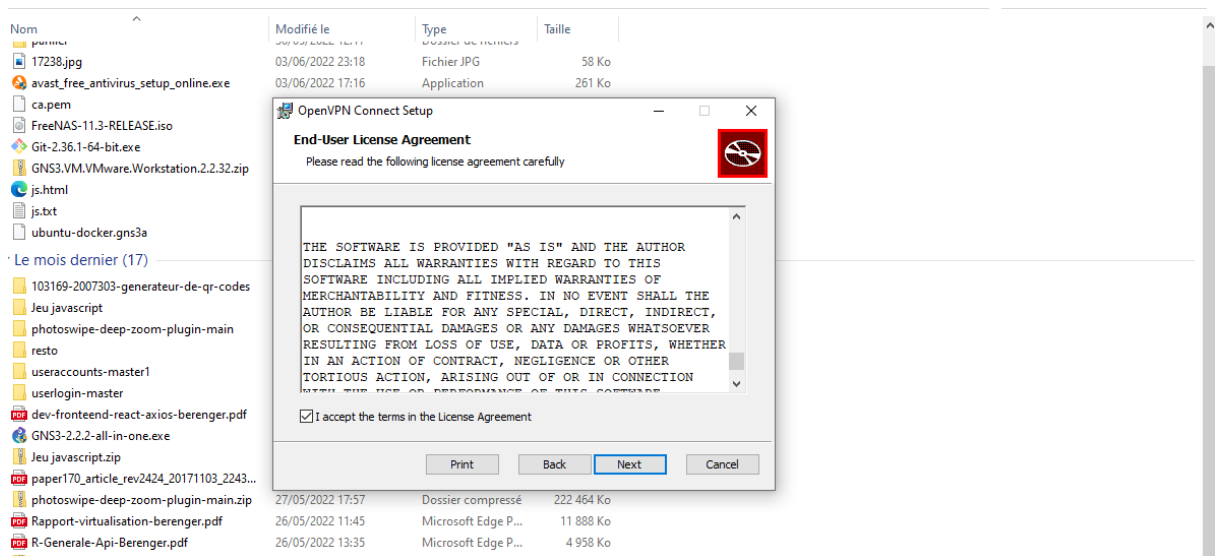
Voici le lien du téléchargement : <https://openvpn.net/community-downloads/>



Il faut faire un clic droit et installer.

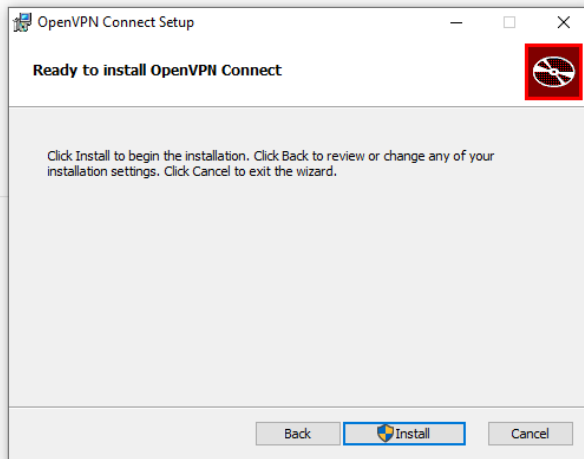


On clique sur **Next**

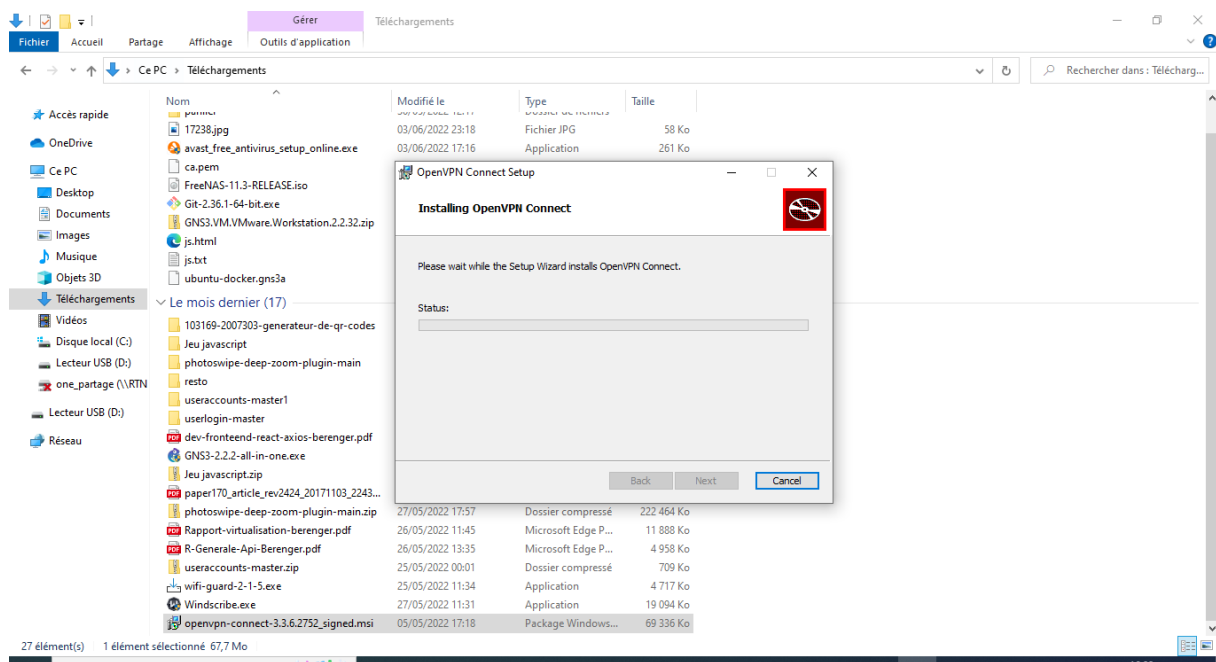


On coche la case puis **Next**

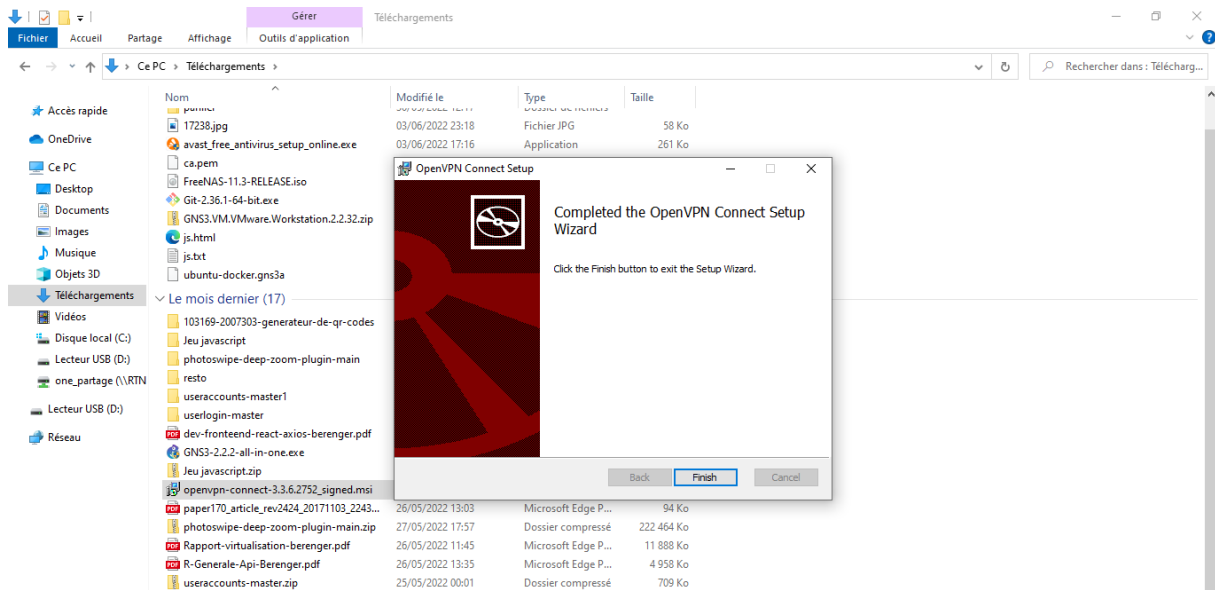
	Modifié le	Type	Taille
	03/06/2022 23:18	Fichier JPG	58 Ko
stup_online.exe	03/06/2022 17:16	Application	261 Ko
E.iso			
rkstation.2.2.32.zip			
rateur-de-qr-codes			
m-plugin-main			
ixios-berenger.pdf			
.exe			
l24_20171103_2243...			
m-plugin-main.zip	27/05/2022 17:57	Dossier compressé	222 464 Ko
-berenger.pdf	26/05/2022 11:45	Microsoft Edge P...	11 888 Ko
ger.pdf	26/05/2022 13:35	Microsoft Edge P...	4 958 Ko
zip	25/05/2022 00:01	Dossier compressé	709 Ko
	25/05/2022 11:34	Application	4 717 Ko
	27/05/2022 11:31	Application	19 094 Ko
6.2752_signed.msi	05/05/2022 17:18	Package Windows...	69 336 Ko



Cliquez sur Installer



Puis Terminé



OpenVPN Connect

Import Profile

URL

FILE

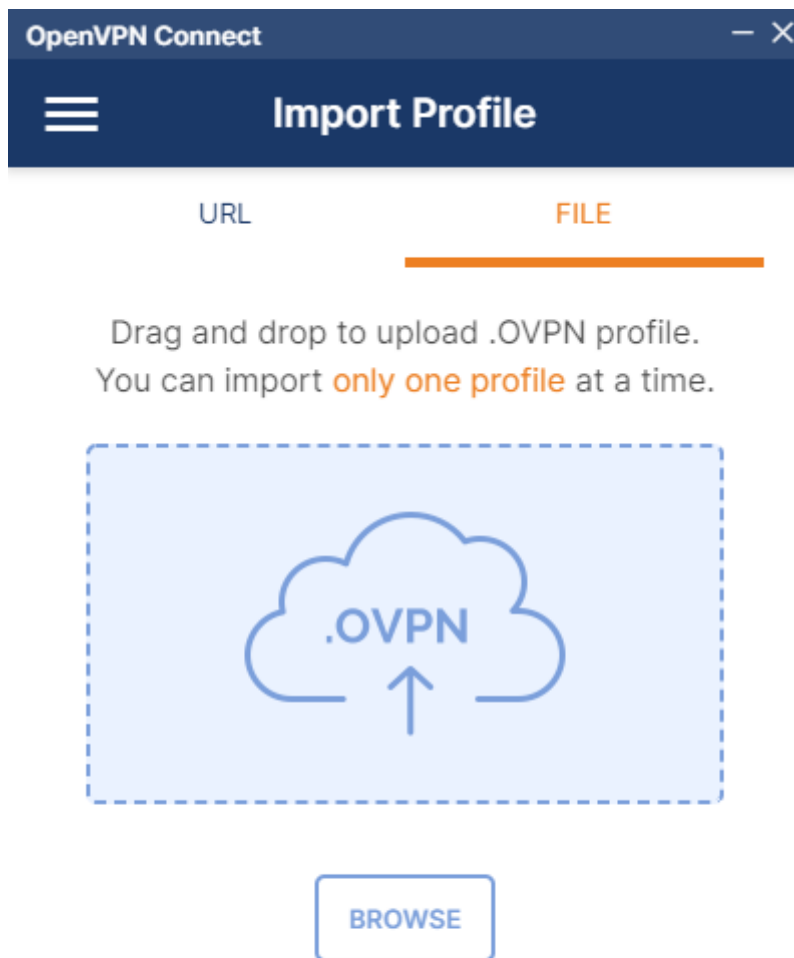
URL

https://192.168.1.226|

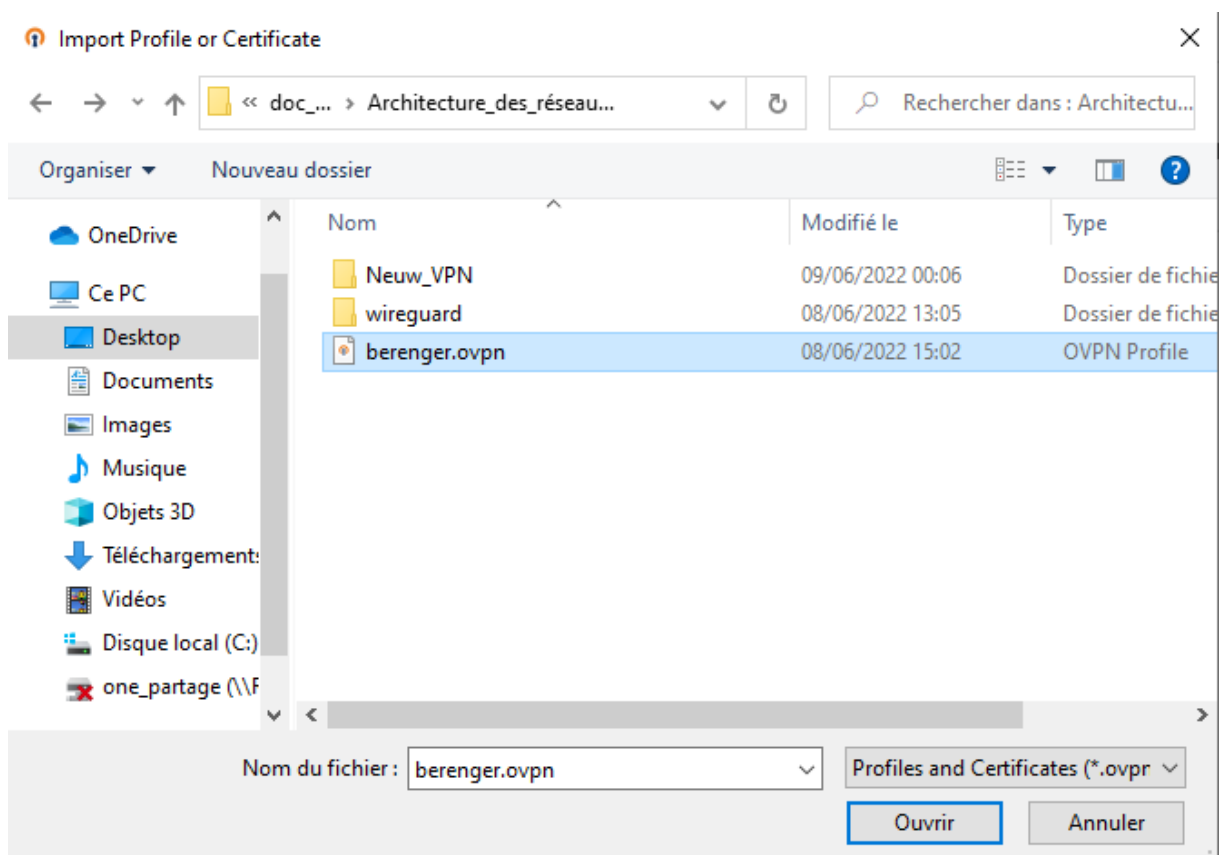
Please note that you can only import profile using URL if it is supported by your VPN provider

NEXT

On précise Ip du serveur



On clique BROWSE pour choisir le certificat.



Puis cliquer sur **Ouvrir**



Imported Profile

Profile Name

192.168.1.226 [berenger]

Server Hostname (locked)

192.168.1.226

☐ Save Private Key Password

PROFILES

CONNECT

Automatiquement il détecte l'adresse IP du serveur puis coche la case.

OpenVPN Connect

<

Imported Profile

Profile Name

192.168.1.226 [berenger]


Server Hostname (locked)

192.168.1.226

☒ Save Private Key Password

Private Key Password

.....



PROFILES

CONNECT

On renseigne le mot de passe du client et Cliquez sur **Connect**.

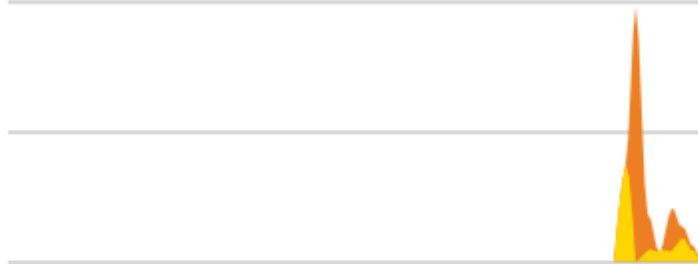
**CONNECTED**

OpenVPN Profile

192.168.1.226 [berenger]

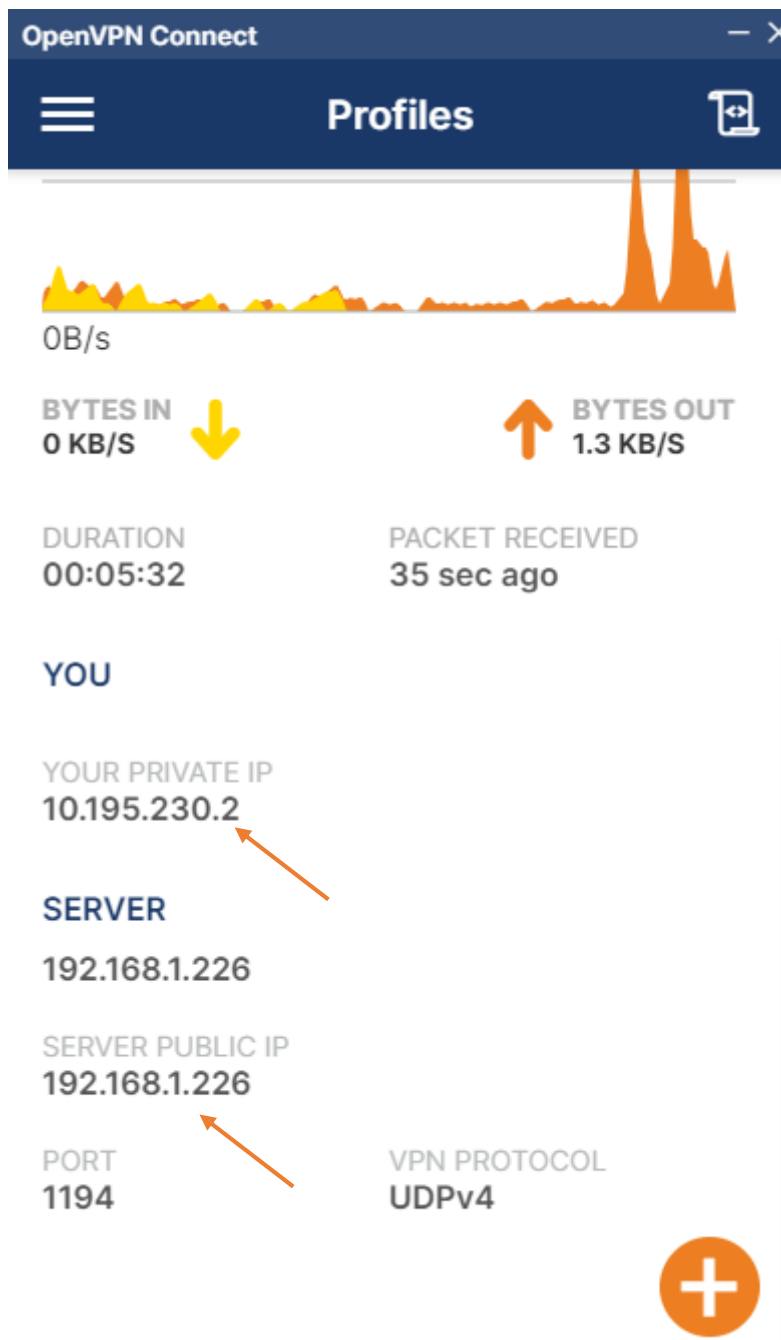
DISCONNECTED**CONNECTION STATS**

7.2KB/s



0B/s

BYTES IN
222 B/S**BYTES OUT**
341 B/S**DURATION**
00:00:08**PACKET RECEIVED**
0 sec ago



On voit l'adresse **10.195.230.2** que le tunnel lui donne

iOS (iPhone)

Côté serveur on va créer un utilisateur iPhone.

```

root@berenger:~# pivpn -a
Enter a Name for the Client: iphone
How many days should the certificate last? 1080
Enter the password for the client:
Enter the password again to verify:
Passwords do not match! Please try again.
Enter the password for the client:
Enter the password again to verify:
spawn ./easyrsa build-client-full iphone
* Notice:
Using Easy-RSA configuration from: /etc/openssl/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 1.1.1 11 Sep 2018

Generating an EC private key
writing new private key to '/etc/openssl/easy-rsa/pki/c6860479/temp.6c53dc5e'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
* Notice:

```

```

Write out database with 1 new entries
Data Base Updated

* Notice:
Certificate created at: /etc/openssl/easy-rsa/pki/issued/iphone.crt

Client's cert found: iphone.crt
Client's Private Key found: iphone.key
CA public Key found: ca.crt
tls Private Key found: ta.key

```

```

=====
Done! iphone.ovpn successfully created!
iphone.ovpn was copied to:
/home/berenger/ovpns
for easy transfer. Please use this profile only on one
device and create additional profiles for other devices.
=====

```

On se place dans le dossier **ovpns**

```

root@berenger:~# cd ovpns/
root@berenger:~/ovpns# ls
berenger.ovpn  iphone.ovpn
root@berenger:~/ovpns# chmod +x iphone.ovpn
root@berenger:~/ovpns#

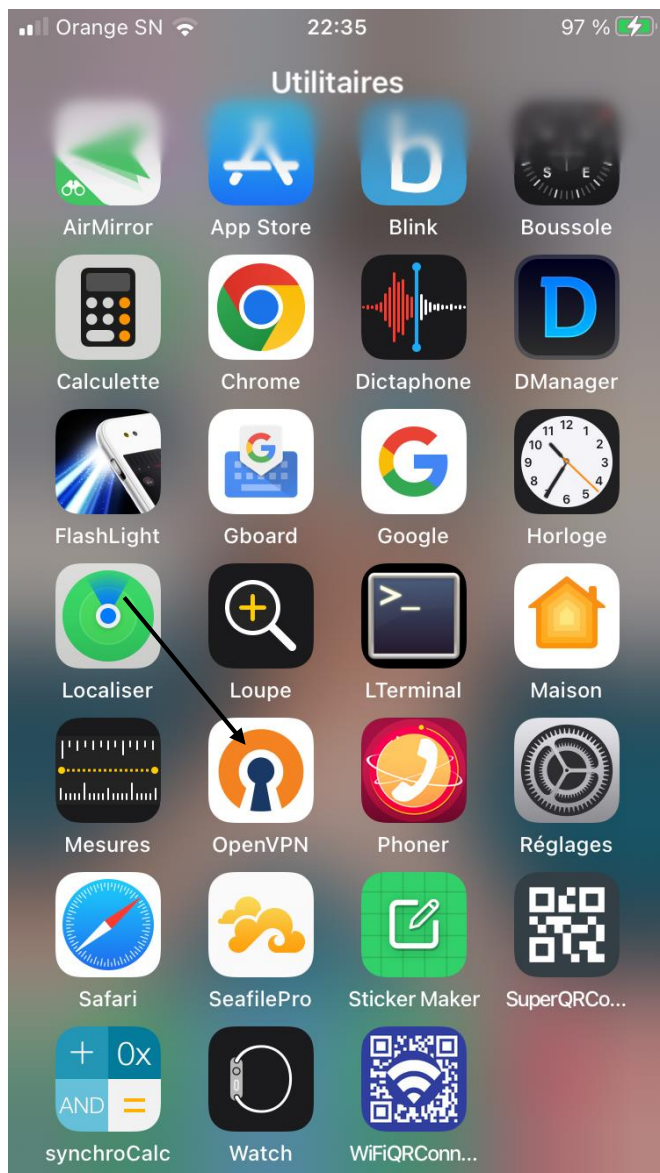
```

On voit le fichier qui contient le certificat du client iPhone puis on lui donne le droit X ensuite faut transférer le certificat sur le phone iPhone.

Installation

Depuis l'App Store d'iTunes, recherchez et installez **OpenVPN Connect**, l'application client OpenVPN officielle pour iOS. Pour transférer la configuration de votre client iOS sur l'appareil, connectez-le directement à un ordinateur.

La procédure à suivre pour effectuer le transfert avec iTunes est décrite ici. Ouvrez iTunes sur l'ordinateur et cliquez sur **iPhone > apps**. Faites défiler vers le bas jusqu'à la section **Partage de fichiers** et cliquez sur l'application OpenVPN. La fenêtre vide à droite, **Documents OpenVPN**, est destinée au partage de fichiers. Faites glisser le fichier **.ovpn** vers la fenêtre OpenVPN Documents (Documents OpenVPN).



On met IP du serveur

Orange SN22:3597 %

Import Profile

URL

FILE

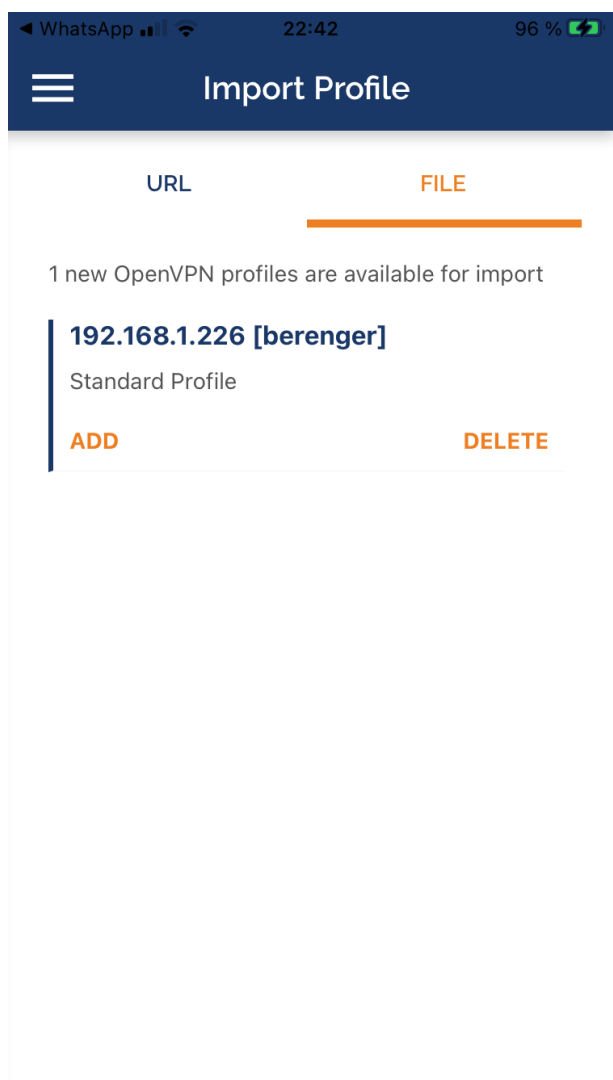
URL

https://192.168.1.226

Please note that you can only import profile using URL if it is supported by your VPN provider

NEXT

On ajoute le certificat



Cliquez sur ADD pour ajoute le certificat

 Profile successfully imported

Profile Name

192.168.1.226 [berenger]

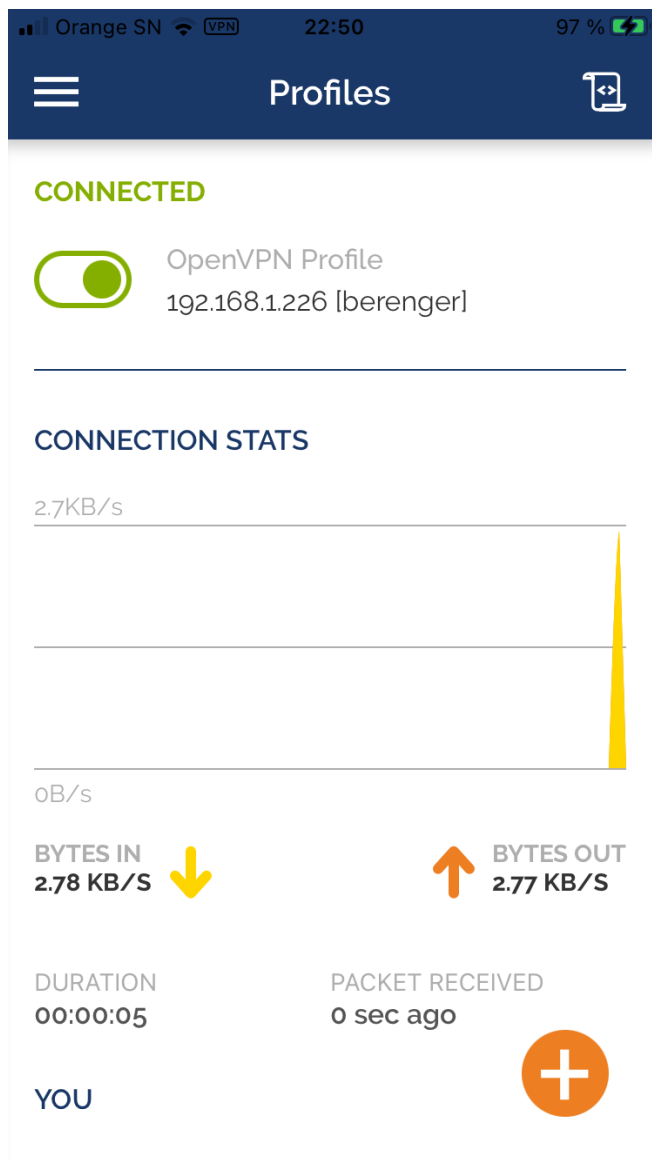
☒ Save Private Key Password

Private Key Password



☒ Connect after import

On renseigne le mot de passe de client



Le client s'est connecté.

Conclusion :

Le protocole OpenVPN est responsable de la gestion des communications client-serveur. Fondamentalement, il aide à établir un "tunnel" sécurisé entre le client VPN et le serveur VPN. Quand OpenVPN gère le cryptage et l'authentification, il utilise la bibliothèque OpenSSL de manière assez extensive.

