

OpenVPN Serveur sur Docker

<https://www.grottedubarbu.fr/serveur-openvpn-5-minutes-docker/>

- [Mise en place OpenVPN Serveur avec Docker](#)
- [Utiliser Open VPN Connect](#)
- [Les commandes OVPN Server](#)

Mise en place OpenVPN Serveur avec Docker

Image Docker utilisé : <https://hub.docker.com/r/kylemanna/openvpn/>

Mise en place du serveur

Dans un premier temps, nous allons cloner le dépôt :

```
git clone https://github.com/kylemanna/docker-openvpn.git
```

Dès le téléchargement terminé, rendez-vous dans le dossier :

```
cd docker-openvpn
```

Nous allons pouvoir construire l'image avec les éléments à jour :

```
docker build -t server_vpn .
```

TCP vs UDP ?

On utilise généralement le protocole UDP pour une connexion VPN. Il est malgré tout possible de réaliser cette connexion en TCP.

La configuration TCP offre les avantages suivants :

- Plus stable puisque le protocole TCP garantit l'émission des paquets,
- By-Pass des firewalls : Il est rare de bloquer le port 80 et 443 en TCP,

Par contre de ce fait le protocole TCP est plus lent.

De ce fait le protocole UDP offrira comme gros avantage d'être beaucoup plus rapide. Par contre il peut être déconseillé dans des cas où la connexion n'est pas stable ou si votre infrastructure ne le permet pas.

Création du dossier des configurations

Vous pouvez ensuite créer un dossier afin de stocker les fichiers de configuration et les clés nécessaires au serveur. Si vous ne stockez pas ces informations persistantes, au prochain redémarrage de votre serveur, il sera nécessaire de refaire la configuration et de générer à nouveau les clés.

```
cd ..  
mkdir openvpn-etc
```

Lançons maintenant notre première configuration :

```
docker run -v $PWD/openvpn-etc:/etc/openvpn --rm server_vpn ovpn_genconfig -u  
udp://IP_ADDRESS:1194
```

Génération des certificats

Nous devons maintenant générer les clés PKI du serveur. Le script d'initialisation, en plus de créer la clé privée, va également générer le certificat CA. Un mot de passe vous sera demandé afin de sécuriser votre clé privée.

```
docker run -v $PWD/openvpn-etc:/etc/openvpn --rm -it server_vpn ovpn_initpki
```

Le script exécuté sera cette fois-ci : `ovpn_initpki`

Une fois ces éléments générés, vous pouvez lancer le serveur :

```
docker run -v $PWD/openvpn-etc:/etc/openvpn -d -p 1194:1194/udp --cap-add=NET_ADMIN server_vpn
```

Créer un utilisateur sans mot de passe

Maintenant, nous allons pouvoir créer les comptes utilisateurs pour connecter au VPN. L'image possède encore un outil pour faciliter notre travail :

```
docker run -v $PWD/openvpn-etc:/etc/openvpn --rm -it server_vpn easysrsa build-client-full  
user1 nopass
```

L'outil **easysrsa** vous demandera le mot de passe CA (mot de passe du certificat). C'est le mot de passe que nous avons défini ci-dessus pendant la commande **ovpn_initpki**.

Nous venons d'ajouter un compte utilisateur avec le nom *user1*, et dans le cadre de notre tutoriel, j'ai utilisé l'option `nopass`. Avec cette option, l'utilisateur peut se connecter directement avec le fichier de configuration.

Cela signifie également que si quelqu'un obtient le fichier de configuration que nous venons de générer en quelques secondes, il pourra alors se connecter à notre serveur VPN sans avoir besoin d'une autre authentification.

Créer un utilisateur avec mot de passe

Pour créer un utilisateur avec mot de passe il suffit d'exécuter la même commande sans mettre l'argument `nopass`.

```
docker run -v $PWD/openvpn-etc:/etc/openvpn --rm -it server_vpn easysrsa build-client-full user1
```

L'outil **easysrsa** vous demandera tout d'abord le mot de passe PEM qui correspond à la pass phrase de l'utilisateur. Puis l'outil demandera dans un deuxième temps le mot de passe CA (mot de passe du certificat). C'est le mot de passe que nous avons défini ci-dessus pendant la commande **ovpn_initpki**. Exemple :

```
root@antec:/docker/openvpn# sudo docker run -v $PWD:/etc/openvpn --rm -it server_vpn easysrsa
build-client-full user1
Using SSL: openssl OpenSSL 1.1.1q  5 Jul 2022
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/openvpn/pki/easy-rsa-1.fDdFdh/tmp.HEpcli'
Enter PEM pass phrase: <passphraseForUser1>
Verifying - Enter PEM pass phrase: <passphraseForUser1>
-----
Using configuration from /etc/openvpn/pki/easy-rsa-1.fDdFdh/tmp.dCdGDM
Enter pass phrase for /etc/openvpn/pki/private/ca.key: <passphraseOpenVPN>
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName             :ASN.1 12:'user1'
Certificate is to be certified until Jan 29 09:56:53 2025 GMT (825 days)

Write out database with 1 new entries
Data Base Updated
```

La création d'un utilisateur génère les trois fichiers suivants :

- /openvpn-etc/pki/issued/`user1.crt`
- /openvpn-etc/pki/private/`user1.key`
- /openvpn-etc/pki/reqs/`user1.req`

Exporter le fichier **.ovpn** d'un utilisateur

Dernière étape de notre installation, nous allons exporter le fichier de configuration qui sera envoyé à l'utilisateur pour se connecter :

```
sudo docker exec openvpn ovpn_getclient <user> > monuser.ovpn
```

Relancer le serveur VPN en cas d'arrêt

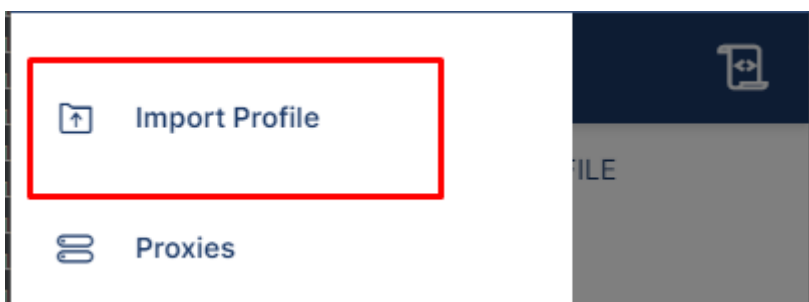
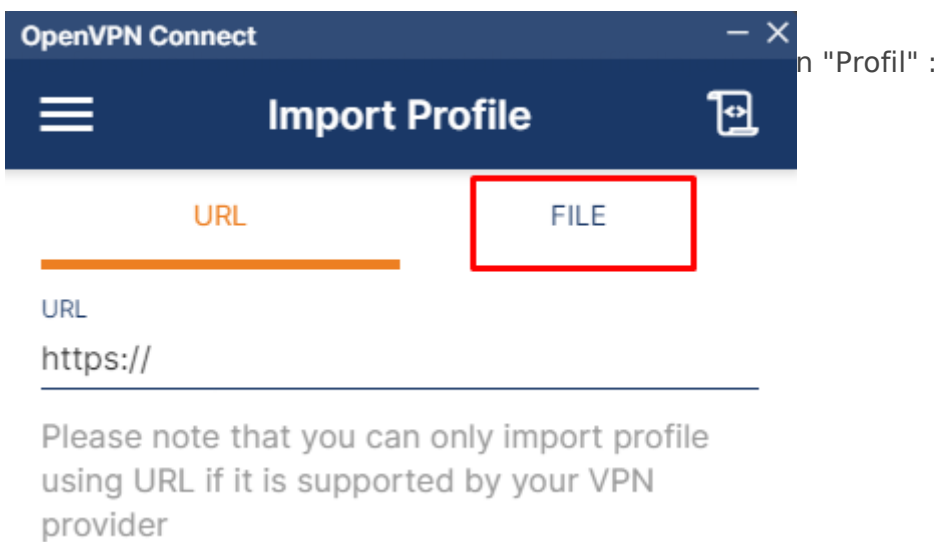
```
sudo docker start <docker_name>
```

Utiliser Open VPN Connect

“ Le tuto pour se connecter à un VPN de type OpenVPN depuis un ordinateur ou un téléphone se fait avec un fichier `.ovpn` ”

Depuis un ordinateur

Il faut tout d'abord installer [Installer Open VPN Connect](#) sur l'ordinateur disposant du fichier en `.ovpn`.



Sélectionner l'onglet "File". Une fois sur l'onglet File, deux façon s'offre a vous. La première glisser déposer le fichier ou la deuxième, appuyer sur le bouton "Browse" et parcourir les dossiers pour sélectionner votre fichier.

Maintenant que le fichier est chargé, vous pouvez donner un nom au fichier Profil et si vous le souhaitez cocher la case "Save Private Key Password" pour éviter de taper le mot de passe a chaque connexion.

Drag and drop to upload .OVPN profile.
You can import **only one profile** at a time.



BROWSE



Imported Profile

Profile Name

Exemple Profil [user1]

Server Hostname (locked)

my-server-name.com

Save Private Key Password

Vous pouvez dès a présent appuyer sur le bouton "Connect" pour utiliser le fichier de .ovpn pour se connecter au VPN. Saisissez le mot de passe et valider.

Profile: Exemple Profil [user1]

Private Key Password



OK

CANCEL

CONNECTED



OpenVPN Profile
Exemple Profil [user1]

CONNECTION STATS

25.7KB/s

0B/s

BYTES IN
427 B/S



BYTES OUT
2.13 KB/S

DURATION
00:00:15

PACKET RECEIVED
0 sec ago

Si vous avez la même interface que l'image ci-dessous c'est que tout fonctionne et que vous êtes bien connecté au VPN.

Depuis un Téléphone

Installer l'application mobile depuis l'AppStore ou le PlayStore :



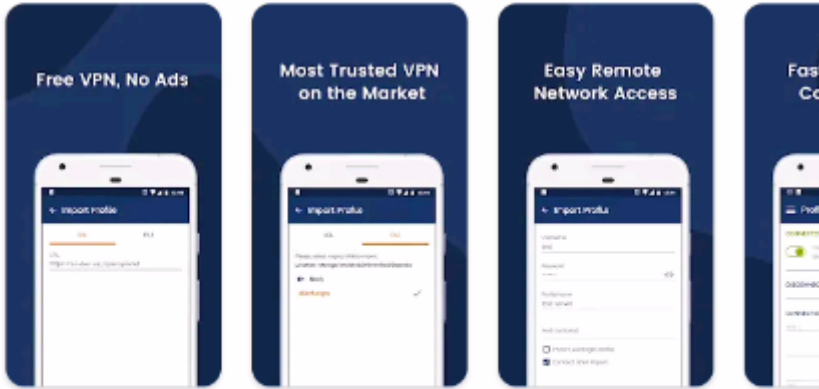
OpenVPN Connect – O...
OpenVPN

Installer

4,1 ★
169 k avis ⓘ

Plus de 10 M
Téléchargement
s

3
PEGI 3 ⓘ



Application officielle pour les services compatibles Access Server, OpenVPN Cloud et OpenVPN.

Les commandes OVPN Server

Liste des commandes possibles sur OpenVPN Serveur

Attention si le VPN se trouve dans un conteneur Docker vous devez soit vous connecter dans le conteneur, soit exécuter les commandes OVPN via l'invite de commande de Docker.

Se connecter dans un conteneur

```
sudo docker exec -it <container_name> bash
```

Lancer une commande dans un conteneur

```
sudo docker exec <container-name> <command>
```

Exemple :

```
sudo docker exec openvpn_server ovpn_listclients
```

Exécuter la commande via le CLI de Docker

```
sudo docker <container_name> <la_commande_ovpn>
```

Liste des utilisateurs enregistré sur le VPN

```
ovpn_listclients
```

Liste des utilisateurs connecté sur le VPN

```
ovpn_status
```

Supprimer un utilisateur du VPN

```
ovpn_revokeclient <user>
```

Récupérer le fichier de conf d'un utilisateur

```
ovpn_getclient <user> > monuser.ovpn
```

Récupérer le fichier de conf d'un utilisateur depuis l'extérieur du conteneur

```
sudo docker exec openvpn ovpn_getclient <user> > monuser.ovpn
```