

UF Infra B2 2024

— Documentation technique

DE DONATO Tony
BARBESIER Axel

Table des matières

.....	0
Configuration du VPN	2
VPS Ionos.....	2
Configuration OpenVPN	4
Routage IP	6
Créer les certificats des clients.....	7
Connexion Client.....	9
Gestion de l'utilisateur	12
Modifier le mot de passe	12
Révocation du certificat	12
Désactivation du compte système	12
Suppression d'un utilisateur	13
Vérification de la connexion utilisateur.....	13
Gestion et Maintenance.....	13
Schéma d'infrastructure	14

Configuration du VPN

VPS Ionos

Dans un premier temps, nous avons besoin d'un server pour héberger nos services. Pour cela nous avons choisis d'utiliser un VPS Ionos.

	CPU	RAM	NVMe	Prix
VPS Linux XS	1 vCore	1 Go	10 Go	<div>Seulement</div> <div>1€</div> <div>HT/mois (1,20 € TTC)</div> <div>Frais de création : 10 € HT</div> <div>Configurer</div>

Nous n'aurons pas besoin d'une grande puissance puisque notre serveur ne traitera pas énormément d'informations. Le Vps de base suffit donc.

On accède ensuite à l'interface mise à disposition par Ionos pour le management du server :

IONOS MENU

Rechercher des fonctions, des domaines ou de l'...

Dernière connexion : 19/06/2024 21:51:32 depuis 92.184.100.153 (France)

Serveurs

Actions Réseau

Type	Nom	État	IP	Taille	SE	Centre de calcul
Vpn			87.106.163.94	VPS 1-1-10	Ubuntu 24.04	Germany

On peut ici vérifier que le système d'exploitation installé nous convient ou le modifier si ce n'est pas le cas.

État	IP	Taille	SE	Centre de calcul
	87.106.163.94	VPS 1-1-10	Ubuntu 24.04	Germany

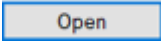
Pour ma part, le server tourne sous Ubuntu 24.04 et est basé en Allemagne.

La section « Données de connexion » nous donne les informations nécessaire pour se connecter à distance à notre server et l'administrer.

Fonctions

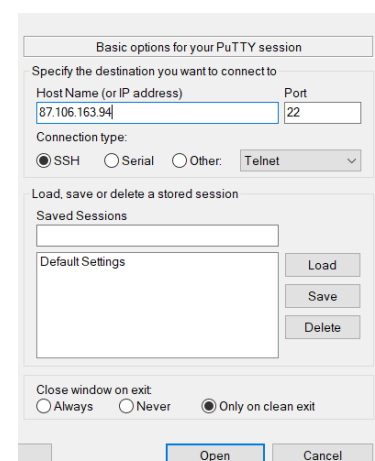
Données de connexion :	
Hôte :	87.106.163.94
Utilisateur :	root
Mot de passe initial :	Afficher mot de passe

Je conseil de s'y connecter avec Putty, il suffit alors de renseigner l'adresse IP du server et de se connecter sur le port 22 (SSH) en telnet.

On clique sur  ce qui ouvre une fenêtre nous permettant de nous connecter en utilisant l'utilisateur et le mot de passe donné par lonos.



```
87.106.163.94 - PuTTY
login as: root
root@87.106.163.94's password: █
```



Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): 87.106.163.94 Port: 22

Connection type: ☒ SSH ☐ Serial ☐ Other: Telnet

Load, save or delete a stored session

Saved Sessions: [Empty field]

Default Settings: [Empty field]

Load Save Delete

Close window on exit: ☐ Always ☐ Never ☒ Only on clean exit

Open Cancel

Configuration OpenVPN

Nous pouvons ensuite passer à la configuration basic de notre vpn.

Dans un premier temps et comme toujours, nous mettons à jour le système.

```
sudo apt update && sudo apt upgrade -y
```

Ensuite on installe OpenVPN et Easy-RSA :

```
sudo apt install openvpn easy-rsa -y
```

On crée ensuite un répertoire pour Easy-RSA et on initialise la PKI (Public Key Infrastructure) :

```
make-cadir ~/openvpn-ca
```

```
cd ~/openvpn-ca
```

Nous modifions le fichier « vars » pour configurer les informations de certificat :

```
nano vars
```

Pour ma part voici comment je l'ai modifié.

```
set_var EASYRSA_REQ_COUNTRY    "FR"  
set_var EASYRSA_REQ_PROVINCE   "PACA"  
set_var EASYRSA_REQ_CITY       "Aix"  
set_var EASYRSA_REQ_ORG        "tony et axel Co"  
set_var EASYRSA_REQ_EMAIL      "tonydedo13@gmail.com"  
set_var EASYRSA_REQ_OU         "responsable vpn"
```

On initialise les variables et on crée l'autorité de certification (CA) :

```
./easysrsa clean-all
```

```
./easysrsa init-pki
```

```
./easysrsa build-ca
```

Puis on crée le certificat et la clé pour le serveur.

```
./easysrsa gen-req server nopass
```

```
./easysrsa sign-req server server
```

On crée le répertoire pour les clés s'il n'existe pas.

```
mkdir -p keys
```

Génération des paramètres Diffie-Hellman et génération des clés de cryptage HMAC :

```
./ easysrsa gen-dh
```

```
openvpn --genkey secret keys/ta.key
```

On copie alors les fichiers nécessaires dans le répertoire OpenVPN :

```
sudo cp ~/openvpn-ca/keys/{server.crt,server.key,ca.crt,dh2048.pem} /etc/openvpn
```

Nous créons ensuite un fichier de configuration pour le serveur :

```
sudo nano /etc/openvpn/server.conf
```

Voici un exemple de configuration :

On peut voir que je spécifie le port souhaité, le protocole utilisé, les fichiers à utiliser (que je viens de créer) ...

Mais ce sont globalement les éléments basiques à mettre dans la configuration d'un VPN.

```
port 1194
proto udp
dev tun
ca ca.crt
cert server.crt
key server.key
dh dh.pem
auth SHA256
tls-auth ta.key 0
topology subnet
server 10.8.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
push "redirect-gateway def1 bypass-dhcp"
push "dhcp-option DNS 8.8.8.8"
push "dhcp-option DNS 8.8.4.4"
keepalive 10 120
cipher AES-256-CBC
user nobody
group nogroup
persist-key
persist-tun
status openvpn-status.log
log-append /var/log/openvpn.log
verb 3
key-direction 0
explicit-exit-notify 1
```

Routage IP

Activer le routage IP en modifiant le fichier de configuration sysctl.

```
sudo nano /etc/sysctl.conf
```

Et décommenter la ligne suivante :

```
net.ipv4.ip_forward = 1
```

On applique ensuite la configuration.

```
sudo sysctl -p
```

Il faut alors configurer le pare-feu pour permettre le trafic OpenVPN.

```
sudo ufw allow 1194/udp
```

```
sudo ufw allow OpenSSH
```

```
sudo ufw enable
```

Puis démarrer et activer le service OpenVPN

```
sudo systemctl start openvpn@server
```

```
sudo systemctl enable openvpn@server
```

Créer les certificats des clients

Retourner au répertoire Easy-RSA et créer un certificat pour chaque client.

```
cd ~/openvpn-ca
```

Pour chaque client, exécuter :

```
./easyrsa build-key NOM_DU_CLIENT
```

 (Remplacer NOM_DU_CLIENT par le nom de votre client.)

Créer les répertoires nécessaires pour stocker les informations du client :

```
mkdir -p ~/client-configs/files
```

On copie ensuite les certificats et clés générés pour le client :

```
cd ~/openvpn-ca
```

```
cp pki/ca.crt ~/client-configs/files/
```

```
cp pki/issued/NOM_DU_CLIENT.crt ~/client-configs/files/
```

```
cp pki/private/NOM_DU_CLIENT.key ~/client-configs/files/
```

```
cp /etc/openvpn/ta.key ~/client-configs/files/
```

Remplacez NOM_DU_CLIENT par le nom de votre client.

Nous pouvons maintenant créer le fichier de configuration de base pour le client :

```
nano ~/client-configs/base.conf
```


On ajoute alors les infos de configuration dans base.conf, voici un exemple :

```
client
dev tun
proto udp
remote 87.106.163.94 1194 udp
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
auth SHA256
cipher AES-256-CBC
ignore-unknown-option block-outside-dns
block-outside-dns
key-direction 1
comp-lzo
verb 3
auth-user-pass
```

Remplacez « 87.106.163.94 » par l'adresse IP de votre serveur OpenVPN.

Pour faciliter la création du client nous avons créé un script pour générer les fichiers de configuration :

```
nano ~/client-configs/make_config.sh
```

Ajoutez les lignes suivantes dans make_config.sh :

```
#!/bin/bash

# First argument: Client identifier
KEY_DIR=~/.openvpn-ca/pki/private
OUTPUT_DIR=~/.client-configs/files
BASE_CONFIG=~/.client-configs/base.conf

cat ${BASE_CONFIG} \
  <(echo -e '<ca>') \
  ~/.openvpn-ca/pki/ca.crt \
  <(echo -e '</ca>\n<cert>') \
  ~/.openvpn-ca/pki/issued/${1}.crt \
  <(echo -e '</cert>\n<key>') \
  ~/.openvpn-ca/pki/private/${1}.key \
  <(echo -e '</key>\n<tls-auth>') \
  /etc/openvpn/ta.key \
  <(echo -e '</tls-auth>') \
  > ${OUTPUT_DIR}/${1}.ovpn
```

Rendre le script exécutable :

```
chmod 700 ~/client-configs/make_config.sh
```

On génère ensuite les configurations pour un client :

```
sudo adduser NOM_DU_CLIENT
```

(On est amené à entrer un mot de passe que l'on devra transmettre à l'utilisateur)

```
~/client-configs/make_config.sh NOM_DU_CLIENT
```

Remplacez NOM_DU_CLIENT par le nom de votre client.

Normalement, cela a dû vous créer une fichier NOM_DU_CLIENT.ovpn comme ceci :

```
root@ubuntu:~/client-configs/files# ls -a
.  ..  ca.crt  client1.crt  client1.key  client1.ovpn  ta.key
```


Il suffit maintenant de récupérer le contenu de ce fichier et de ce fichier et de le transmettre au client.

Connexion Client

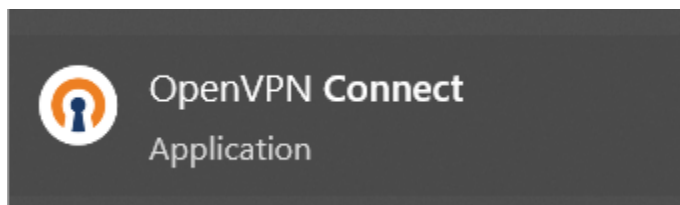
L'utilisateur doit télécharger le logiciel client « openvpn Connect » au lien suivant :

<https://openvpn.net/client/client-connect-vpn-for-windows/>

Il doit ensuite exécuter le fichier .msi :

 openvpn-connect-3.4.4.3412_signed.msi

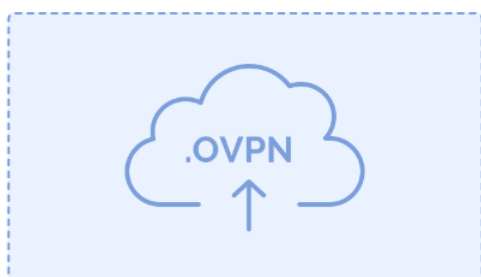
Une fenêtre s'ouvre alors l'invitant à poursuivre l'installation, une fois celle-ci complétée l'utilisateur peut lancer le logiciel client :



Et arriver ici :

Il doit alors cliquer sur « UPLOAD FILE » et glisser le fichier .ovpn fourni par le server.

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



BROWSE

 The image shows a screenshot of the 'OpenVPN Connect' application window. The title bar says 'OpenVPN Connect'. The main window has a dark blue header with a menu icon and the text 'Import Profile'. Below the header, there are two tabs: 'VIA URL' (selected) and 'UPLOAD FILE'. Under 'VIA URL', there is a text input field with the URL 'https://'. Below the input field, there is a note: 'Please note that you can only import profile using URL if it is supported by your VPN provider'. At the bottom of the window, there is an orange button labeled 'NEXT'.

L'utilisateur peut alors renseigner le nom qu'il souhaite donner à ce profile vpn et cliquer sur « CONNECT » :

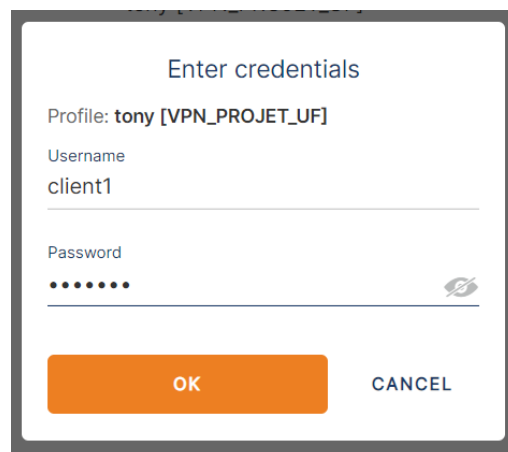
Profile Name

Tony [VPN_PROJET_UF]

Server Hostname (locked)

87.106.163.94

Il faut ensuite que l'utilisateur renseigne ses infos de connexion qu'on lui a normalement fournit :



Enter credentials

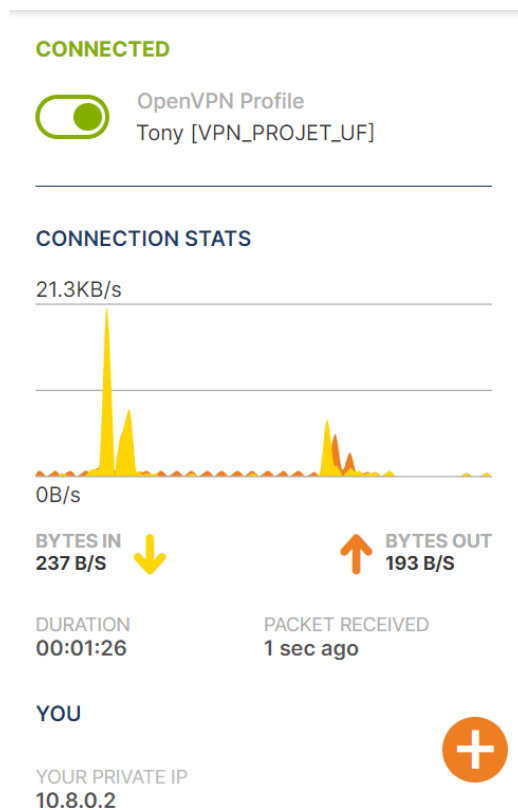
Profile: tony [VPN_PROJET_UF]

Username
client1

Password
.....

OK CANCEL

L'utilisateur est maintenant connecté :



Gestion de l'utilisateur

Modifier le mot de passe

Modifier le mot de passe de l'utilisateur :

```
sudo passwd NOM_DU_CLIENT
```

Révocation du certificat

Aller dans le répertoire Easy-RSA où on a configuré votre PKI :

```
cd ~/openvpn-ca
```

Révoquer le certificat du client :

```
./easyrsa revoke client1
```

Générer une nouvelle liste de révocation de certificats (CRL) :

```
./easyrsa gen-crl
```

Copiez le fichier CRL généré dans le répertoire OpenVPN et assurez-vous que `crl-verify` est inclus dans votre configuration serveur OpenVPN :

```
sudo cp /etc/openvpn/crl.pem /etc/openvpn/crl.pem
```

```
sudo systemctl restart openvpn@server
```

Désactivation du compte système

Pour désactiver le compte utilisateur sans le supprimer, vous pouvez verrouiller le compte :

```
sudo usermod -L client1
```

Suppression d'un utilisateur

Si vous souhaitez supprimer complètement un utilisateur du système, utilisez la commande `deluser` :

```
sudo deluser NOM_DU_CLIENT
```

Pour supprimer également le répertoire personnel de l'utilisateur, utilisez l'option `--remove-home` :

```
sudo deluser --remove-home NOM_DU_CLIENT
```

Vérification de la connexion utilisateur

Vous pouvez vérifier les utilisateurs connectés au VPN en consultant le fichier de statut généré par OpenVPN :

```
cat /var/log/openvpn-status.log
```

Ce fichier contient des informations sur les connexions actives, y compris les adresses IP assignées et les noms d'utilisateur.

Gestion et Maintenance

Logs : Consulter régulièrement les journaux (`/var/log/openvpn.log` et `/var/log/openvpn-status.log`) pour surveiller l'activité et détecter les problèmes.

Mises à jour : S'assurer que le serveur VPN et les bibliothèques associées sont à jour pour garantir la sécurité et les performances.

Backup : Sauvegarder régulièrement les configurations et clés/certificats PKI.

Schéma d'infrastructure

