Partie 1

Étape 1 : Mettre à jour et mettre à niveau Debian

Avant d'installer un logiciel, il est essentiel de mettre à jour et mettre à niveau votre système Debian. Exécutez les commandes suivantes :

sudo apt update && sudo apt upgrade -y

Étape 2 : Installer OpenVPN

Installation de OpenVPN sur le serveur Debian1 avec la commande suivante :
sudo apt install openvpn easy-rsa -y

```
root@debian1:/home/renman# sudo apt install openvpn easy-rsa
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libccid libpkcs11-helper1 opensc opensc-pkcs11 pcscd
Suggested packages:
  pcmciautils resolvconf openvpn-dco-dkms openvpn-systemd-resolved
The following NEW packages will be installed:
  easy-rsa libccid libpkcs11-helper1 opensc opensc-pkcs11 openvpn pcscd
0 upgraded, 7 newly installed, 0 to remove and 0 not upgraded.
Need to get 2,499 kB of archives.
After this operation, 7,628 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://deb.debian.org/debian bookworm/main amd64 libccid amd64 1.5.2-1 [367 kB]
Get:2 http://deb.debian.org/debian bookworm/main amd64 pcscd amd64 1.9.9-2 [89.7 kB]
Get:3 http://deb.debian.org/debian bookworm/main amd64 easy-rsa all 3.1.0-1 [54.8 kB]
Get:4 http://deb.debian.org/debian bookworm/main amd64 libpkcs11-helper1 amd64 1.29.0-1 [51.2 kB]
Get:5 http://deb.debian.org/debian bookworm/main amd64 opensc-pkcs11 amd64 0.23.0-0.3+deb12u1 [914 kB]
Get:6 http://deb.debian.org/debian bookworm/main amd64 opensc amd64 0.23.0-0.3+deb12u1 [371 kB]
Get:7 http://deb.debian.org/debian bookworm/main amd64 openvpn amd64 2.6.3-1+deb12u2 [651 kB]
Fetched 2,499 kB in 3s (965 kB/s)
Preconfiguring packages ...
Selecting previously unselected package libccid.
(Reading database ... 177156 files and directories currently installed.)
Preparing to unpack .../0-libccid_1.5.2-1_amd64.deb ...
Unpacking libccid (1.5.2-1) ...
Selecting previously unselected package pcscd.
Preparing to unpack .../1-pcscd_1.9.9-2_amd64.deb ...
Unpacking pcscd (1.9.9-2) ...
Selecting previously unselected package easy-rsa.
Preparing to unpack .../2-easy-rsa_3.1.0-1_all.deb ...
Unpacking easy-rsa (3.1.0-1) ...
Selecting previously unselected package libpkcs11-helper1:amd64.
Preparing to unpack .../3-libpkcs11-helper1_1.29.0-1_amd64.deb ...
Unpacking libpkcs11-helper1:amd64 (1.29.0-1) ...
Selecting previously unselected package opensc-pkcs11:amd64.
```

Étape 3 : Générer les certificats et les clés

OpenVPN s'appuie sur des certificats et des clés pour l'authentification du client et du serveur. Pour générer ces fichiers, on utilise le script easy-rsa inclus :

```
root@debian1:/home/renman# make-cadir ~/openvpn-ca && cd ~/openvpn-ca
root@debian1:~/openvpn-ca# ls
easyrsa  openssl-easyrsa.cnf  vars  x509-types
root@debian1:~/openvpn-ca# nano vars
```

on modifie le fichier vars pour configurer les variables de l'autorité de certification :

nano vars:

```
set_var EASYRSA_REQ_COUNTRY     "FR"
set_var EASYRSA_REQ_PROVINCE    "Ile-de-France"
set_var EASYRSA_REQ_CITY        "Paris"
set_var EASYRSA_REQ_ORG "My company"
set_var EASYRSA_REQ_EMAIL       "renover.manirafasha@gmail.com"
set_var EASYRSA_REQ_OU          "My Organizational Unit"
```

On génére les certificats et clés requis :

./easyrsa init-pki

```
root@debian1:~/openvpn-ca# ./easyrsa init-pki
* Notice:

  init-pki complete; you may now create a CA or requests.

  Your newly created PKI dir is:
  * /root/openvpn-ca/pki

root@debian1:~/openvpn-ca#
```

./easyrsa build-ca

```
root@debian1:~/openvpn-ca# ./easyrsa build-ca
* Notice:
Using Easy-RSA configuration from: /root/openvpn-ca/vars

* WARNING:

  Move your vars file to your PKI folder, where it is safe!

* Notice:
Using SSL: openssl OpenSSL 3.0.13 30 Jan 2024 (Library: OpenSSL 3.0.13 30 Jan 2024)


Enter New CA Key Passphrase:
Re-Enter New CA Key Passphrase:
Using configuration from /root/openvpn-ca/pki/3512c8f0/temp.c53454f8
.....+...+.++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*........+..+.......+...+........+......+...+...+....+..
..+.+.........++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*....+...+.+...........+......+....+...+......+......
++++++++++++++++++++++++++++++++++++++++++
.....+.+...+.........+.................+...+.+.........++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.....+.....
++++++++++++++++++++++++++++++++++*....+...+..+...+.......+...........+....+......+...........+...+.......+...+............+...+....
...................+...+...............+.....+.+...........+...+.......+..............+.+........+......+.+..............+....+......
...+.....+......+...+.+......+.....+.........+.+...+...+...+..+.......+..+.+...+.........+..+..........+......+.....+.........+....+
+............+....+.........+............+...+......+...+..+.+..............+.+..............+..+.............................+.+.
.++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
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.
-----
Common Name (eg: your user, host, or server name) [Easy-RSA CA]:renman

* Notice:

CA creation complete and you may now import and sign cert requests.
Your new CA certificate file for publishing is at:
/root/openvpn-ca/pki/ca.crt
```

./easyrsa gen-req server nopass:

```
root@debian1:~/openvpn-ca# ./easyrsa gen-req server nopass
* Notice:
Using Easy-RSA configuration from: /root/openvpn-ca/vars

* WARNING:

  Move your vars file to your PKI folder, where it is safe!

* Notice:
Using SSL: openssl OpenSSL 3.0.13 30 Jan 2024 (Library: OpenSSL 3.0.13 30 Jan 2024)

.+.+...+.....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.........+++++++++++++++++++++++++++++++++++++++
.+....+..+....+...+........+....+......+..+......+.........+.+......+...+..+......+...+.+.....+....+.........+..+.+..+.+.....
.......................+...+..............+.........+...+..+...+.+.........+...+...+..+...+......+...+......+......+........
...+............+...+......+...+...+..+.............+..+...+..........+..+...+...+.+......+...+...........+...................
....+.......+......+.....+...+....+...+.....+....+...+...............+........+...+.+...+..+....+......+......+...+..+.......
+++
.............+...+.+..+.......+..+...+............+....+..+.........+...+.+.........+......+..+....+...............+.....+.+.
++++++++++*..+.............+............+.....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*..+...+......+
.+.+........+............+...+......................+...+.+.........+........+...........+..+.......+.....+.+........+.+.......
......+...+...+....+...+........+......+.........+.........+.......+...+.....+...+.............+..+.........+.+............+.
...+.+........+......+.......................+...+...........+...+......+....+........+....+.....................+.+........+..
...................+.........+......+.....+.+...+..+...+......+...+.......+......+................+..+.+................+..+....
...+...+.+.....+.+.............+..+.............+......+..+..........+........+.+.....+.+..+......+......+....+......+...+..+.
..+...+.................+......+..........+.....+......+.+..+.............+............+..+.+..+.+.....+...........+....+...
+.+......+...+.................+....+.....+......+.......+...+..............+.+........+.+.........+.....+.+......+.....+....
.......++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
-----
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.
-----
Common Name (eg: your user, host, or server name) [server]:renman
* Notice:

Keypair and certificate request completed. Your files are:
req: /root/openvpn-ca/pki/reqs/server.req
key: /root/openvpn-ca/pki/private/server.key
```

./easyrsa sign-req server server

```
root@debian1:~/openvpn-ca# ./easyrsa sign-req server server
* Notice:
Using Easy-RSA configuration from: /root/openvpn-ca/vars

* WARNING:

  Move your vars file to your PKI folder, where it is safe!

* Notice:
Using SSL: openssl OpenSSL 3.0.13 30 Jan 2024 (Library: OpenSSL 3.0.13 30 Jan 2024)


You are about to sign the following certificate.
Please check over the details shown below for accuracy. Note that this request
has not been cryptographically verified. Please be sure it came from a trusted
source or that you have verified the request checksum with the sender.

Request subject, to be signed as a server certificate for 825 days:

subject=
    commonName                = renman


Type the word 'yes' to continue, or any other input to abort.
  Confirm request details: yes

Using configuration from /root/openvpn-ca/pki/00180bc9/temp.86c4a332
Enter pass phrase for /root/openvpn-ca/pki/private/ca.key:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'renman'
Certificate is to be certified until Nov  8 21:05:31 2026 GMT (825 days)

Write out database with 1 new entries
Database updated

* Notice:
Certificate created at: /root/openvpn-ca/pki/issued/server.crt
```

./easyrsa gen-dh

```
root@debian1:~/openvpn-ca# ./easyrsa gen-dh
* Notice:
Using Easy-RSA configuration from: /root/openvpn-ca/vars

* WARNING:

  Move your vars file to your PKI folder, where it is safe!

* Notice:
Using SSL: openssl OpenSSL 3.0.13 30 Jan 2024 (Library: OpenSSL 3.0.13 30 Jan 2024)

Generating DH parameters, 2048 bit long safe prime
.....................................................................................+.....
.....................................................................................+.....
.....................................................................+.....................
.....................................................................+.....+...........
+...................................................................+.................+.................+.
.....................................................................+.....................................
.....................................................................+.....................................
.....................................................................+.....................................
.....................................................................+.....................................
.....................+.....................................................................+.....................
.....................+.....................................................................+.....................
.....................+.....+.............+.....................................................+.....................
.....................+.....................................................+.....................................
.............+.....................................................................+.....................
.....................+.....................................................................+.....................
.....................+.....................................................................+.....................
```

openvpn --genkey secret pki/ta.key

```
root@debian1:~/openvpn-ca# openvpn --genkey secret pki/ta.key
root@debian1:~/openvpn-ca#
```

Ces certificats et clés seront stockés dans le répertoire **/root/openvpn-ca/pki.**

Étape 4 : Configurer OpenVPN

Après avoir généré les certificats et les clés, on procède à la configuration d'OpenVPN. on Crée Un nouveau fichier de configuration avec la commande suivante :

sudo cp /usr/share/doc/openvpn/examples/sample-config-files/server.conf /etc/openvpn/server.conf

on copie les fichiers nécessaires dans le répertoire OpenVPN :

cp /root/openvpn-ca/pki/{ca.crt,dh.pem,ta.key} /etc/openvpn

```
root@debian1:~/openvpn-ca# cp /root/openvpn-ca/pki/{ca.crt,dh.pem,ta.key} /etc/openvpn
root@debian1:~/openvpn-ca#
```

cp /root/openvpn-ca/pki/issued/server.crt /etc/openvpn

cp /root/openvpn-ca/pki/private/server.key /etc/openvpn

```
root@debian1:~/openvpn-ca# cp /root/openvpn-ca/pki/issued/server.crt /etc/openvpn
root@debian1:~/openvpn-ca# cp /root/openvpn-ca/pki/private/server.key /etc/openvpn
root@debian1:~/openvpn-ca#
```

on modifie /etc/openvpn/server.conf pour qu'il corresponde à ce qui suit :

```
ca ca.crt
cert server.crt
key server.key  # This file should be kept secret

# Diffie hellman parameters.
# Generate your own with:
#   openssl dhparam -out dh2048.pem 2048
dh dh.pem
```

On enregistre et on ferme le fichier.

Étape 5 : Activer le transfert IP

on modifie la configuration sysctl :

sudo nano /etc/sysctl.conf

```
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
#  Enabling this option disables Stateless Address Autoconfiguration
#  based on Router Advertisements for this host
```

Appliquez les changements :

sudo sysctl -p

```
root@debian1:~/openvpn-ca# sudo sysctl -p
net.ipv4.ip_forward = 1
root@debian1:~/openvpn-ca#
```

Partie 2: Configuration du Démarrage Automatique et du Pare-feu

Étape 6 : Démarrer et activer OpenVPN

On démarre et on active le service OpenVPN :

sudo systemctl start openvpn@server

```
root@debian1:~/openvpn-ca# sudo systemctl start openvpn@server
root@debian1:~/openvpn-ca# sudo systemctl enable openvpn@server
Created symlink /etc/systemd/system/multi-user.target.wants/openvpn@server.service → /lib/sys
temd/system/openvpn@.service.
root@debian1:~/openvpn-ca#
```

sudo systemctl status openvpn@server.service

```
oot@debian1:~/openvpn-ca# systemctl status openvpn@server.service
 openvpn@server.service - OpenVPN connection to server
    Loaded: loaded (/lib/systemd/system/openvpn@.service; enabled; preset: enabled)
    Active: active (running) since Tue 2024-08-06 00:45:29 BST; 45min ago
      Docs: man:openvpn(8)
            https://community.openvpn.net/openvpn/wiki/Openvpn24ManPage
            https://community.openvpn.net/openvpn/wiki/HOWTO
  Main PID: 15396 (openvpn)
    Status: "Initialization Sequence Completed"
     Tasks: 1 (limit: 10)
    Memory: 2.5M
       CPU: 118ms
    CGroup: /system.slice/system-openvpn.slice/openvpn@server.service
            └─15396 /usr/sbin/openvpn --daemon ovpn-server --status /run/openvpn/serve
```

Le @server spécifie le fichier de configuration qu'on a créé précédemment.

Étape 7 : Installation et Configuration du pare-feu

apt install ufw -y

```
oot@debian1:~/openvpn-ca# apt install ufw
eading package lists... Done
uilding dependency tree... Done
eading state information... Done
he following additional packages will be installed:
 iptables libip6tc2
uggested packages:
 firewalld rsyslog
he following NEW packages will be installed:
 iptables libip6tc2 ufw
 upgraded, 3 newly installed, 0 to remove and 0 not up
eed to get 548 kB of archives.
fter this operation, 3,411 kB of additional disk space
o you want to continue? [Y/n] y
```

Configurer ufw pour n'autoriser que les connexions HTTP (port 80),ssh et OpenVPN :

sudo ufw allow 1194/udp
sudo ufw allow 80/tcp
sudo ufw allow 22/tcp
sudo ufw enable
ufw status

NB:J'ai autorisé les connexions ssh car j'accede à mes machines virtuelles via ssh.

```
root@debian1:~/openvpn-ca# ufw allow 1194/udp
Rules updated
Rules updated (v6)
root@debian1:~/openvpn-ca# ufw allow 80/tcp
Rules updated
Rules updated (v6)
root@debian1:~/openvpn-ca# ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? n
Aborted
root@debian1:~/openvpn-ca# ufw allow 22/tcp
Rules updated
Rules updated (v6)
root@debian1:~/openvpn-ca# ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? n
Aborted
root@debian1:~/openvpn-ca# ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
root@debian1:~/openvpn-ca# ufw status
Status: active

To                         Action      From
--                         ------      ----
1194/udp                   ALLOW       Anywhere
80/tcp                     ALLOW       Anywhere
22/tcp                     ALLOW       Anywhere
1194/udp (v6)              ALLOW       Anywhere (v6)
80/tcp (v6)                ALLOW       Anywhere (v6)
22/tcp (v6)                ALLOW       Anywhere (v6)

root@debian1:~/openvpn-ca#
```

Étape 8 : Se connecter au serveur OpenVPN

Avec le serveur OpenVPN opérationnel, vous pouvez vous y connecter à partir d'un ordinateur client. Installez le logiciel client OpenVPN et téléchargez le fichier de configuration client à partir du serveur :

$ ./easyrsa gen-req client1 nopass

$ ./easyrsa sign-req client client1

$ cp pki/private/client1.key /etc/openvpn/client/

$ cp pki/issued/client1.crt /etc/openvpn/client/

$ cp pki/{ca.crt,ta.key} /etc/openvpn/client/

On génère le certificat CSR et la clé privée du client1

```
root@debian1:~/openvpn-ca# ./easyrsa gen-req client1 nopass
* Notice:
Using Easy-RSA configuration from: /root/openvpn-ca/vars

* WARNING:

  Move your vars file to your PKI folder, where it is safe!

* Notice:
Using SSL: openssl OpenSSL 3.0.13 30 Jan 2024 (Library: OpenSSL 3.0.13 30 Jan 2024)

...+...+..+....+.....+............+...+......+.+...+.....+......+.+.........+...+..+.........+.+......+.
++*.+...+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*........+...+...+..+.......+..
+
.++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*...++++++++++++++++++++++++++++++++
+....+......+.................+...+...............+.......+...............+...+........+................
.+...+...+...+.....+.+........+.+..+....+..............+.+.........+.....+....+.....+..........+..+...+.
++++++++++++++++++++++++++++++++++++++++++++++++++
-----
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.
-----
Common Name (eg: your user, host, or server name) [client1]:client1
* Notice:

Keypair and certificate request completed. Your files are:
req: /root/openvpn-ca/pki/reqs/client1.req
key: /root/openvpn-ca/pki/private/client1.key
```

Signature du certificat du client1

```
root@debian1:~/openvpn-ca# ./easyrsa sign-req client client1
* Notice:
Using Easy-RSA configuration from: /root/openvpn-ca/vars

* WARNING:

  Move your vars file to your PKI folder, where it is safe!

* Notice:
Using SSL: openssl OpenSSL 3.0.13 30 Jan 2024 (Library: OpenSSL 3.0.13 30 Jan 2024)


You are about to sign the following certificate.
Please check over the details shown below for accuracy. Note that this request
has not been cryptographically verified. Please be sure it came from a trusted
source or that you have verified the request checksum with the sender.

Request subject, to be signed as a client certificate for 825 days:

subject=
    commonName                = client1


Type the word 'yes' to continue, or any other input to abort.
  Confirm request details: yes

Using configuration from /root/openvpn-ca/pki/03c8bdc8/temp.de0b68f5
Enter pass phrase for /root/openvpn-ca/pki/private/ca.key:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'client1'
Certificate is to be certified until Nov  9 08:52:17 2026 GMT (825 days)

Write out database with 1 new entries
Database updated

* Notice:
Certificate created at: /root/openvpn-ca/pki/issued/client1.crt
```

On copie les fichiers nécessaires vers le répertoire du client

```
root@debian1:~/openvpn-ca# ls /etc/openvpn/client/
root@debian1:~/openvpn-ca# cp pki/private/client1.key /etc/openvpn/client/
root@debian1:~/openvpn-ca# cp pki/issued/client1.crt /etc/openvpn/client/
root@debian1:~/openvpn-ca# cp pki/{ca.crt,ta.key} /etc/openvpn/client/
root@debian1:~/openvpn-ca# ls /etc/openvpn/client/
ca.crt  client1.crt  client1.key  ta.key
root@debian1:~/openvpn-ca#
```

Création d' un fichier de configuration client dans le répertoire /root/openvpn-ca :

```
cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf
/root/openvpn-ca/
```

```
root@debian1:~/openvpn-ca#  cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf /root/openvpn-ca/
root@debian1:~/openvpn-ca#
```

Modification du fichier :`/root/openvpn-ca/client.conf` à l'aide de `nano` et configurez les variables :

```
# file can be used for all clients.
ca ca.crt
cert client1.crt
key client1.key

# Verify server certificate by checking that the
# certificate has the correct key usage set.
# This is an important precaution to protect against
# a potential attack discussed here:
#  http://openvpn.net/howto.html#mitm
#
# To use this feature, you will need to generate
# your server certificates with the keyUsage set to
#   digitalSignature, keyEncipherment
# and the extendedKeyUsage to
#   serverAuth
# EasyRSA can do this for you.
remote-cert-tls server

# If a tls-auth key is used on the server
# then every client must also have the key.
tls-auth ta.key 1

# Select a cryptographic cipher.
# If the cipher option is used on the server
# then you must also specify it here.
# Note that v2.4 client/server will automatically
# negotiate AES-256-GCM in TLS mode.
# See also the data-ciphers option in the manpage
cipher AES-256-CBC
```

Création d'un script (conf_gen.sh)pour compiler la configuration de base avec les fichiers de certificat, clé et chiffrement nécessaires :

```
  GNU nano 7.2                                    config_gen.sh
#!/bin/bash
# Premier argument : identifiant du client

# Répertoires et fichiers de base
KEY_DIR=/etc/openvpn/client
OUTPUT_DIR=/root
BASE_CONFIG=/root/openvpn-ca/client.conf

# Génération du fichier de configuration .ovpn
cat ${BASE_CONFIG} \
    <(echo -e '<ca>') \
    ${KEY_DIR}/ca.crt \
    <(echo -e '</ca>\n<cert>') \
    ${KEY_DIR}/${1}.crt \
    <(echo -e '</cert>\n<key>') \
    ${KEY_DIR}/${1}.key \
    <(echo -e '</key>\n<tls-crypt>') \
    ${KEY_DIR}/ta.key \
    <(echo -e '</tls-crypt>') \
    > ${OUTPUT_DIR}/${1}.ovpn
```

on rend le script exécutable:

```
chmod 700 /root/openvpn-ca/config_gen.sh
```

```
root@debian1:~/openvpn-ca# chmod 700 /root/openvpn-ca/config_gen.sh
root@debian1:~/openvpn-ca#
```

Cette commande va créer un fichier client1.ovpn dans le répertoire /root/. On copie ce fichier sur notre ordinateur client et nous l'utilisons pour vous connecter au serveur OpenVPN.
Installation de openvpn sur le poste de l'utilisateur 1:

sudo apt install openvpn easy-rsa -y

```
root@debian3:/home/renman# apt install openvpn easy-rsa -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libccid libpkcs11-helper1 opensc opensc-pkcs11 pcscd
Suggested packages:
  pcmciautils resolvconf openvpn-dco-dkms openvpn-systemd-resolved
The following NEW packages will be installed:
  easy-rsa libccid libpkcs11-helper1 opensc opensc-pkcs11 openvpn pcscd
0 upgraded, 7 newly installed, 0 to remove and 0 not upgraded.
Need to get 2,499 kB of archives.
After this operation, 7,628 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian bookworm/main amd64 libccid amd64 1.5.2-1 [36
Get:2 http://deb.debian.org/debian bookworm/main amd64 pcscd amd64 1.9.9-2 [89.7
```

test de connexion du client vers le serveur:

openvpn --config /root/client1.ovpn

```
root@debian3:~# openvpn --config /root/client1.ovpn
2024-08-06 15:55:35 Note: Kernel support for ovpn-dco missing, disabling data channel offload.
2024-08-06 15:55:35 OpenVPN 2.6.3 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] [DCO]
2024-08-06 15:55:35 library versions: OpenSSL 3.0.13 30 Jan 2024, LZO 2.10
2024-08-06 15:55:35 DCO version: N/A
2024-08-06 15:55:35 TCP/UDP: Preserving recently used remote address: [AF_INET]192.168.1.25:1194
2024-08-06 15:55:35 Socket Buffers: R=[212992->212992] S=[212992->212992]
2024-08-06 15:55:35 UDPv4 link local: (not bound)
2024-08-06 15:55:35 UDPv4 link remote: [AF_INET]192.168.1.25:1194
2024-08-06 15:55:35 NOTE: UID/GID downgrade will be delayed because of --client, --pull, or --up-delay
2024-08-06 15:55:35 TLS: Initial packet from [AF_INET]192.168.1.25:1194, sid=08ab0685 6f194b77
2024-08-06 15:55:35 VERIFY OK: depth=1, CN=renman
2024-08-06 15:55:35 VERIFY KU OK
2024-08-06 15:55:35 Validating certificate extended key usage
2024-08-06 15:55:35 ++ Certificate has EKU (str) TLS Web Server Authentication, expects TLS Web Server Authentication
2024-08-06 15:55:35 VERIFY EKU OK
2024-08-06 15:55:35 VERIFY OK: depth=0, CN=renman
2024-08-06 15:55:35 Control Channel: TLSv1.3, cipher TLSv1.3 TLS_AES_256_GCM_SHA384, peer certificate: 2048 bit RSA, signature: RSA-SHA256
2024-08-06 15:55:35 [renman] Peer Connection Initiated with [AF_INET]192.168.1.25:1194
2024-08-06 15:55:35 TLS: move_session: dest=TM_ACTIVE src=TM_INITIAL reinit_src=1
2024-08-06 15:55:35 TLS: tls_multi_process: initial untrusted session promoted to trusted
2024-08-06 15:55:35 PUSH: Received control message: 'PUSH_REPLY,route 192.168.1.0 255.255.255.0,route 10.8.0.1,topology net30,ping 10,ping-restart 120,ifconfig 10.8.0.6 10.8.0.5,peer-id 0,c
ipher AES-256-GCM,protocol-flags cc-exit tls-ekm dyn-tls-crypt,tun-mtu 1500'
2024-08-06 15:55:35 OPTIONS IMPORT: --ifconfig/up options modified
2024-08-06 15:55:35 OPTIONS IMPORT: route options modified
2024-08-06 15:55:35 OPTIONS IMPORT: tun-mtu set to 1500
2024-08-06 15:55:35 net_route_v4_best_gw query: dst 0.0.0.0
2024-08-06 15:55:35 net_route_v4_best_gw result: via 192.168.1.1 dev enp0s3
2024-08-06 15:55:35 ROUTE_GATEWAY 192.168.1.1/255.255.255.0 IFACE=enp0s3 HWADDR=08:00:27:e8:e0:cf
2024-08-06 15:55:35 TUN/TAP device tun0 opened
2024-08-06 15:55:35 net_iface_mtu_set: mtu 1500 for tun0
2024-08-06 15:55:35 net_iface_up: set tun0 up
2024-08-06 15:55:35 net_addr_ptp_v4_add: 10.8.0.6 peer 10.8.0.5 dev tun0
2024-08-06 15:55:35 net_route_v4_add: 192.168.1.0/24 via 10.8.0.5 dev [NULL] table 0 metric -1
```

Partie 3: Activer l'Authentification à Deux Facteurs

Installation du paquet Google Authenticator sur le serveur OpenVPN :

sudo apt-get install libpam-google-authenticator

```
root@debian1:~/openvpn-ca# sudo apt-get install libpam-google-authenticator
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  libpam-google-authenticator
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 45.5 kB of archives.
After this operation, 138 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian bookworm/main amd64 libpam-google-authenticator amd64 2019
1231-2 [45.5 kB]
Fetched 45.5 kB in 0s (259 kB/s)
Selecting previously unselected package libpam-google-authenticator.
(Reading database ... 177763 files and directories currently installed.)
Preparing to unpack .../libpam-google-authenticator_20191231-2_amd64.deb ...
Unpacking libpam-google-authenticator (20191231-2) ...
Setting up libpam-google-authenticator (20191231-2) ...
Processing triggers for man-db (2.11.2-2) ...
root@debian1:~/openvpn-ca#
```

Configurer OpenVPN pour Utiliser Google Authenticator
Configurer PAM pour Utiliser Google Authenticator :

Éditez le fichier: /etc/pam.d/openvpn :

renman@debian1: ~

```
  GNU nano 7.2                                    openvpn *
auth required pam_google_authenticator.so
```

Configurer OpenVPN pour Utiliser Google Authenticator :

On ajoute les lignes suivantes au fichier de configuration du serveur OpenVPN (/etc/openvpn/server.conf) :

```
plugin /usr/lib/openvpn/openvpn-plugin-auth-pam.so openvpn
reneg-sec 0

# Notify the client that when the server restarts so it
# can automatically reconnect.
explicit-exit-notify 1
```

Redémarrage du service OpenVPN pour appliquer les modifications :

```
root@debian1:/etc/pam.d# sudo systemctl restart openvpn@server
root@debian1:/etc/pam.d# ^C
root@debian1:/etc/pam.d#
```

Configurer Google Authenticator pour Chaque Utilisateur
Chaque utilisateur VPN doit configurer Google Authenticator sur leur propre machine.On se Connecte en tant qu'utilisateur et on exécute :

installation de paquet google authenticator

```
root@debian3:/home/renman# sudo apt-get install libpam-google-authenticator
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  libpam-google-authenticator
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 45.5 kB of archives.
After this operation, 138 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian bookworm/main amd64 libpam-google-authenticator amd64 20191231-2 [45.5 kB]
Fetched 45.5 kB in 0s (206 kB/s)
Selecting previously unselected package libpam-google-authenticator.
(Reading database ... 177447 files and directories currently installed.)
Preparing to unpack .../libpam-google-authenticator_20191231-2_amd64.deb ...
Unpacking libpam-google-authenticator (20191231-2) ...
Setting up libpam-google-authenticator (20191231-2) ...
Processing triggers for man-db (2.11.2-2) ...
root@debian3:/home/renman#
```

google-authenticator:

```
root@debian3:/home/renman# google-authenticator

Do you want authentication tokens to be time-based (y/n) y
Warning: pasting the following URL into your browser exposes the OTP secret to Google:
  https://www.google.com/chart?chs=200x200&chld=M|0&cht=qr&chl=otpauth://totp/root@debian3%3Fsecret%3DM5YD2ZHIXJS3CEBZEKX4TKFU2I%26issuer%3Ddebian3
```

```
Do you want me to update your "/root/.google_authenticator" file? (y/n) Y

Do you want to disallow multiple uses of the same authentication
token? This restricts you to one login about every 30s, but it increases
your chances to notice or even prevent man-in-the-middle attacks (y/n) Y

By default, a new token is generated every 30 seconds by the mobile app.
In order to compensate for possible time-skew between the client and the server,
we allow an extra token before and after the current time. This allows for a
time skew of up to 30 seconds between authentication server and client. If you
experience problems with poor time synchronization, you can increase the window
from its default size of 3 permitted codes (one previous code, the current
code, the next code) to 17 permitted codes (the 8 previous codes, the current
code, and the 8 next codes). This will permit for a time skew of up to 4 minutes
between client and server.
Do you want to do so? (y/n) Y

If the computer that you are logging into isn't hardened against brute-force
login attempts, you can enable rate-limiting for the authentication module.
By default, this limits attackers to no more than 3 login attempts every 30s.
Do you want to enable rate-limiting? (y/n) Y
root@debian3:/home/renman#
```