RAPPORT DE PROJET: VPN

REALISE PAR BAMHAMMED METEHRI & ADRIAN TARNAR

Sommaire:

a) VPN en général

VPN ou Virtual Private Networking, est un ensemble de technologies qui permettent à un appareil de se connecter via un tunnel protégé à un autre réseau. Cela donne une connexion sécurisée et chiffrée. Les VPN nous permettent de nous cacher lorsque nous surfons sur le Web.

Plutôt que d'envoyer votre trafic Internet (par exemple vos recherches en ligne, vos chargements et vos téléchargements) directement à votre fournisseur d'accès Internet (FAI), un VPN achemine d'abord votre trafic via un serveur VPN. Ainsi, lorsque vos données sont finalement transmises sur Internet, elles semblent provenir du serveur VPN et non de votre appareil personnel.

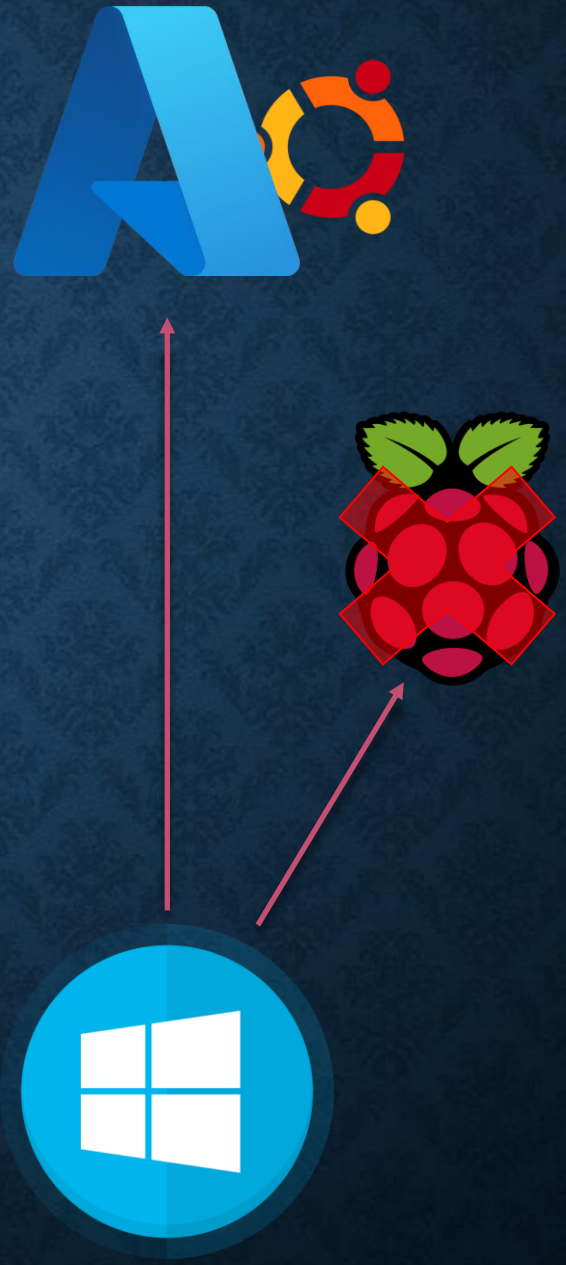
b) Choix de la solution

En terme de matériel:

Tout D'abord, on a commencé par configure le Raspberry qu'on a, ensuite après avoir que c'est impossible d'ouvrir les ports de nos box, j'ai parti vers un VPS (Virtual Private Server) notamment vers un Microsoft Azure VM.

En terme des outils:

- OpenVPN : On choisi cette solution pour le vpn parmi plusieurs, pour sa communauté et sa documentation vaste
- easy-rsa: C'est l'outils lequelle on utilisera pour générer les clées et les cetificats de connexion.



2. Coté Server

```
// d'abord je me connecte vers ma machine à l'aide de ssh
// j'ai mets une ip fixe pour ma VM

/home/mobaxterm $     ssh bam@20.224.137.222
```

```
// j'initialise un mot de passe pout mon root, pour passer et travailler en root
bam@vpnserver:~$ sudo su
root@vpnserver:/home/bam# passwd
New password:
Retype new password:
passwd: password updated successfully

// j'ai fait une mise à jour ensuite j'installe les outies necessaires {Easy-Rsa 3 , OpenVPN}
root@vpnserver:/home/bam# apt update
root@vpnserver:/home/bam# apt upgrad

root@vpnserver:/home/bam# apt install openvpn

root@vpnserver:/home/bam# apt install easy-rsa
```

a) Génération des fichiers d'authentification

```
root@vpnserver:/home/bam# cp -r /usr/share/easy-rsa /etc/        // je cree une copie du fichier easy-rsa dans /etc/
root@vpnserver:/home/bam# cd /etc/easy-rsa/
root@vpnserver:/etc/easy-rsa# ./easyrsa init-pki    // je cree le dossier pki qui va contenir mes fichier d'authentication

init-pki complete; you may now create a CA or requests.
Your newly created PKI dir is: /etc/easy-rsa/pki

root@vpnserver:/etc/easy-rsa# ./easyrsa build-ca nopass  // je cree mon fichier ca.crt

Using SSL: openssl OpenSSL 1.1.1f  31 Mar 2020
Generating RSA private key, 2048 bit long modulus (2 primes)
.................................+++++
.................................+++++
e is 65537 (0x010001)
Can't load /etc/easy-rsa/pki/.rnd into RNG
140507530831168:error:2406F079:random number generator:RAND_load_file:Cannot open
file:../crypto/rand/randfile.c:98:Filename=/etc/easy-rsa/pki/.rnd
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]:server

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

a) Génération des fichiers d'authentification

```
root@vpnserver:/etc/easy-rsa# ./easyrsa gen-dh   // je genere mon fichier Diffie-Hellman qui va difinir la méthode d'échangemnt des clés

Using SSL: openssl OpenSSL 1.1.1f  31 Mar 2020
Generating DH parameters, 2048 bit long safe prime, generator 2
This is going to take a long time
.....................................................+.....................................................
.....................................................................................................
.....................................+.........+.....................................................
.............................+.....................................................++++++++++*

DH parameters of size 2048 created at /etc/easy-rsa/pki/dh.pem

root@vpnserver:/etc/easy-rsa# ./easyrsa build-server-full server nopass // je genere les fichiers de mon serveur

Using SSL: openssl OpenSSL 1.1.1f  31 Mar 2020
Generating a RSA private key
..................................+++++
.....+++++
writing new private key to '/etc/easy-rsa/pki/private/server.key.gEW6nCRXWj'
-----
Using configuration from /etc/easy-rsa/pki/safessl-easyrsa.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'server'
Certificate is to be certified until Mar 26 22:45:01 2025 GMT (1080 days)
Write out database with 1 new entries
Data Base Updated



root@vpnserver:/etc/easy-rsa# ./easyrsa gen-crl // je genere le fichier crl.pem qui a comme role bloquer les clients qui ont pas les
certificats d'auth, à acceder à mon serveur
Using SSL: openssl OpenSSL 1.1.1f  31 Mar 2020
Using configuration from /etc/easy-rsa/pki/safessl-easyrsa.cnf

An updated CRL has been created.
CRL file: /etc/easy-rsa/pki/crl.pem
```

a) Génération des fichiers d'authentification

```
// je copie mes fichiers de serveur vers /etc/openvpn/server
root@vpnserver:/etc/easy-rsa# cp -rp /etc/easy-rsa/pki/{ca.crt,dh.pem,ta.key,crl.pem,issued,private} /etc/openvpn/server/
// je cree mon premier client ‘bamhammed’ cela va generer les fichiers ./private/Bamhammed.key et ./issued/Bamhammed.crt
root@vpnserver:/etc/easy-rsa# ./easyrsa build-client-full bamhammed nopass

Using SSL: openssl OpenSSL 1.1.1f  31 Mar 2020
Generating a RSA private key
.....................+++++
.+++++
writing new private key to '/etc/easy-rsa/pki/private/bamhammed.key.JHKeo40c2V'
-----
Using configuration from /etc/easy-rsa/pki/safessl-easyrsa.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'bamhammed'
Certificate is to be certified until Mar 26 22:48:04 2025 GMT (1080 days)

Write out database with 1 new entries
Data Base Updated
// je cree mon premier client ‘client2’ cela va generer les fichiers ./private/client2.key et ./issued/client2.crt
root@vpnserver:/etc/easy-rsa# ./easyrsa build-client-full client2 nopass

Using SSL: openssl OpenSSL 1.1.1f  31 Mar 2020
Generating a RSA private key
..........+++++
...............................+++++
writing new private key to '/etc/easy-rsa/pki/private/client2.key.nCn8cC8F60'
-----
Using configuration from /etc/easy-rsa/pki/safessl-easyrsa.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'client2'
Certificate is to be certified until Mar 26 22:48:52 2025 GMT (1080 days)

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

a) Génération des fichiers d’authentification

```
// je finie la configuration de mes fichiers d'authentification par les arranger dans les dossier
/etc/openvpn/server --> pour les fichiers de mon serveur
/etc/openvpn/client --> pour les fichiers de mes clients



root@vpnserver:/etc/easy-rsa# mkdir /etc/openvpn/client/{bamhammed,client2}

root@vpnserver:/etc/easy-rsa# cp -rp /etc/easy-rsa/pki/{ca.crt,issued/bamhammed.crt,private/bamhammed.key} /etc/openvpn/client/Bamhammed

root@vpnserver:/etc/easy-rsa# cp -rp /etc/easy-rsa/pki/{ca.crt,issued/client2.crt,private/client2.key} /etc/openvpn/client/client2
```

a) Génération des fichiers d'authentification

```
// Ensuite on va commencer à configure notre serveur


root@vpnserver:/etc/easy-rsa# cd /etc/openvpn/server/
// on va désarchiver notre server.conf
root@vpnserver:/etc/openvpn/server# gunzip server.conf.gz
// Et on le déplace vers notre dossoer de serveur /etc/openvpn/server
root@vpnserver:/etc/openvpn/server# nano /etc/openvpn/server/server.conf
// on va active le ip forawardig se qui nous permettera d'encapsuler toutes les données via notre tunnel et d'être sur que toutes les données
passe par ce tunnel
root@vpnserver:/etc/openvpn/server# sed -i 's/#net.ipv4.ip_forward=1/net.ipv4.ip_forward=1/' /etc/sysctl.conf

root@vpnserver:/etc/openvpn/server# sysctl -system
...
* Applying /etc/sysctl.conf ...
net.ipv4.ip_forward = 1
// on configure notre firewall on autorisant le ssh (22tcp) et le vpn (1194udp)
root@vpnserver:/etc/openvpn/server# ufw allow 1194/udp
Rules updated
Rules updated (v6)

root@vpnserver:/etc/openvpn/server# ufw allow ssh
Rules updated
Rules updated (v6)

root@vpnserver:/etc/openvpn/server# nano /etc/ufw/before.rules
// on accepte toutes les données encapsule renvoyé par notre tunnel
root@vpnserver:/etc/openvpn/server# sed -i 's/DEFAULT_FORWARD_POLICY="DROP"/DEFAULT_FORWARD_POLICY="ACCEPT"/' /etc/default/ufw

root@vpnserver:/etc/openvpn/server# ufw reload
Firewall not enabled (skipping reload)
// on Activera notre firewall
root@vpnserver:/etc/openvpn/server# ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
```

```
root@vpnserver:/etc/openvpn/server# systemctl enable --now openvpn-server@server // maintenant on peux lancer notre serveur
Created symlink /etc/systemd/system/multi-user.target.wants/openvpn-server@server.service → /lib/systemd/system/openvpn-server@.service.

root@vpnserver:/etc/openvpn/server# systemctl status openvpn-server@server
● openvpn-server@server.service - OpenVPN service for server
     Loaded: loaded (/lib/systemd/system/openvpn-server@.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2022-04-11 23:01:24 UTC; 22s ago
       Docs: man:openvpn(8)
             https://community.openvpn.net/openvpn/wiki/Openvpn24ManPage
             https://community.openvpn.net/openvpn/wiki/HOWTO
   Main PID: 16362 (openvpn)
     Status: "Initialization Sequence Completed"
      Tasks: 1 (limit: 19197)
     Memory: 840.0K
     CGroup: /system.slice/system-openvpn\x2dserver.slice/openvpn-server@server.service
             └─16362 /usr/sbin/openvpn --status /run/openvpn-server/status-server.log --status-version 2 --suppress-timestamps --config
server.conf
Apr 11 23:01:24 vpnserver systemd[1]: Starting OpenVPN service for server...
Apr 11 23:01:24 vpnserver systemd[1]: Started OpenVPN service for server.
// on remarque que y a une autre interface qui tun0 , et elle représente notre tunnel
root@vpnserver:/etc/openvpn/server# ip add s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0d:3a:22:68:3e brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.4/24 brd 10.0.0.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::20d:3aff:fe22:683e/64 scope link
       valid_lft forever preferred_lft forever
3: enP62193s1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc mq master eth0 state UP group default qlen 1000
    link/ether 00:0d:3a:22:68:3e brd ff:ff:ff:ff:ff:ff
    altname enP62193p0s2
4: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 100
    link/none
    inet 10.8.0.1/24 brd 10.8.0.255 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::7ce:8ee:76c5:65b3/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
```

2. Coté Client

```
root@vpnserver:/etc/openvpn/client# ls
bamhammed  client2

root@vpnserver:/etc/openvpn/client# cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf ./bamhammed/
root@vpnserver:/etc/openvpn/client# cp ./bamhammed/client.conf ./client2/
root@vpnserver:/etc/openvpn/client# cd bamhammed/
root@vpnserver:/etc/openvpn/client/bamhammed# ls -l
total 20
-rw------- 1 root root 4477 Apr 11 22:48 bamhammed.crt
-rw------- 1 root root 1704 Apr 11 22:48 bamhammed.key
-rw------- 1 root root 1184 Apr 11 22:43 ca.crt
-rw-r--r-- 1 root root 3586 Apr 11 23:16 client.conf

// Pour terminer notre conf, on cree notre fichier ovpn et on ajoute
root@vpnserver:/etc/openvpn/client/bamhammed#
root@vpnserver:/etc/openvpn/client/bamhammed# mv client.conf bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# nano bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# echo "<ca>" >> bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# cat ca.crt >> bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# echo "</ca>" >> bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# echo "<cert>" >> bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# cat bamhammed.crt >> bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# echo "</cert>" >> bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# echo "<key>" >> bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# cat bamhammed.key >> bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# echo "</key>" >> bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed# nano bamhammed.ovpn
root@vpnserver:/etc/openvpn/client/bamhammed#

// Finalement pour récuper les conf on un scp (ssh copy)
// scp bam@20.224.137.222 /etc/openvpn/client ./
```

a) Génération des fichiers d'authentification

Conclusion

Le projet c'est commencé tard pour moi ,j'avais l'occasion d'approfondir mes connaissances sur les détails de chaque outils, il me manquera des temps pour le rend parfait, j'ai vécu des problèmes (les ports de la box, travailler avec le Raspberry). Mais au final c'était très intéressant d'avoir ce que on peux faire en infra et ce que on peux cree, et ça nous a permis de revoir ce que on a vu sur les cours de linux linux et réseaux