Table des matières

1) Questions préliminaires

Pour faire en sorte que le client interne puisse acceder au réseau externe, il faut que le firewall route ses paquets (faire du SNAT)

Pour le faire, il faut :

- Regarder si dans le fichier cat /proc/sys/net/ipv4/ip_forward, on trouve 1

```
root@debian:~# cat /proc/sys/net/ipv4/ip_forward
1
```

Cela veut dire que le routage en ipv4 est actif

- Il faut aussi regarder dans le fichier /etc/sysctl.conf et dé commenter la ligne :

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

Pour activer le routage dans le system et on fait sysctl -p pour relancer la conf

1) Configuration de la machine Open VPN :

- IP = 10.214.3.2

- Masque = 255.255.0.0

- Route = 10.214.255.254

Configuration du client interne :

- IP = 192.168.1.1

- Masque = 255.255.255.0

- Route = 192.168.1.254

Configuration du Firewall :

- IP = 10.214.4.2

- Masque = 255.255.0.0

- Route = 10.214.255.254

- IP = 192.168.1.254

- Masque = 255.255.255.0

D'abord on doit vider toutes les règles d'iptables

```
iptables -F
iptables -X
iptables -t nat -F
iptables -t nat -X
iptables -t mangle -F
iptables -t mangle -X
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT
```

Maintenant on ajoute la règle pour que le pc interne puisse communiquer avec l'extérieur

-Iptables = iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j SNAT --to-source 10.214.4.2

2) Pour installer le paquet openvpn j'utilise la commande suivante :

```
root@debian:~/lzo-1.08# apt-get install openvpn
```

Pour installer la librairie lzo j'utilise les commandes suivantes :

```
root@debian:~/lzo-1.08# wget
http://www.oberhumer.com/opensource/lzo/download/lzo-1.08.tar.gz
```

```
root@debian:~/lzo-1.08# tar -zxvf lzo-1.08.tar.gz
```

```
root@debian:~/lzo-1.08# cd lzo-1.08
```

```
root@debian:~/lzo-1.08# apt install make
```

```
root@debian:~/lzo-1.08# ./configure && make && make check && make test
```

2) Premiers tests

1) Maintenant sur le pc passerelle on exécute la commande suivante :

```
openvpn --dev tun0 --ifconfig 192.168.10.1 192.168.10.2
```

Et maintenant sur le pc openVPN on lance la commande suivante :

```
root@debian:~/lzo-1.08# openvpn --remote 10.214.4.254 --dev tun0 --ifconfig
192.168.10.2 192.168.10.1

Fri Oct 18 11:45:15 2019 disabling NCP mode (--ncp-disable) because not in
P2MP client or server mode
Fri Oct 18 11:45:15 2019 OpenVPN 2.4.7 x86_64-pc-linux-gnu [SSL (OpenSSL)]
[LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Feb 20 2019
Fri Oct 18 11:45:15 2019 library versions: OpenSSL 1.1.1c  28 May 2019, LZO
2.10
Fri Oct 18 11:45:15 2019 ******* WARNING *******: All encryption and
authentication features disabled -- All data will be tunnelled as clear text and
will not be protected against man-in-the-middle changes. PLEASE DO
RECONSIDER THIS CONFIGURATION!
Fri Oct 18 11:45:15 2019 TUN/TAP device tun0 opened
Fri Oct 18 11:45:15 2019 /sbin/ip link set dev tun0 up mtu 1500
Fri Oct 18 11:45:15 2019 /sbin/ip addr add dev tun0 local 192.168.10.2 peer
192.168.10.1
Fri Oct 18 11:45:15 2019 TCP/UDP: Preserving recently used remote address:
[AF_INET]10.214.4.254:1194
Fri Oct 18 11:45:15 2019 UDP link local (bound): [AF_INET][undef]:1194
Fri Oct 18 11:45:15 2019 UDP link remote: [AF_INET]10.214.4.254:1194
Fri Oct 18 11:45:16 2019 Peer Connection Initiated with
[AF_INET]10.214.4.254:1194
Fri Oct 18 11:45:17 2019 WARNING: this configuration may cache passwords in
memory -- use the auth-nocache option to prevent this
Fri Oct 18 11:45:17 2019 Initialization Sequence Completed
```

On voit bien que la liaison a été initié. Pour prouver que la connexion fonctionne bien je ping le pc en interne :

```
root@debian:~# ping 192.168.10.1
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data.
64 bytes from 192.168.10.1: icmp_seq=1 ttl=64 time=1.88 ms
64 bytes from 192.168.10.1: icmp_seq=2 ttl=64 time=2.39 ms
```

```
64 bytes from 192.168.10.1: icmp_seq=3 ttl=64 time=2.05 ms
^C
--- 192.168.10.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 6ms
rtt min/avg/max/mdev = 1.878/2.106/2.393/0.214 ms
```

2) En tapant la commande ifconfig sur les deux machines on peut voir qu'une nouvelle interface est apparue (surligné en violet)

```
root@debian:~# ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:a3:58:7e:e7  txqueuelen 0  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.214.3.2  netmask 255.255.0.0  broadcast 0.0.0.0
        ether 08:00:27:bb:32:58  txqueuelen 1000  (Ethernet)
        RX packets 127716  bytes 162202497 (154.6 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 78564  bytes 8715248 (8.3 MiB)
        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 1  (Boucle locale)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 192.168.10.2  netmask 255.255.255.255  destination 192.168.10.1
        inet6 fe80::920d:a4a6:4fb5:b333  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 13  bytes 876 (876.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 14  bytes 924 (924.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

3) On lance wireshark

On voit sur l'interface tun0 :

```
No.     Time           Source                Destination           Protocol Length Info
     1 0.000000000    192.168.10.2          192.168.10.1          ICMP     84     Echo
(ping) request  id=0x73c9, seq=1/256, ttl=64 (reply in 2)

Frame 1: 84 bytes on wire (672 bits), 84 bytes captured (672 bits) on interface
0
Internet Protocol Version 4, Src: 192.168.10.2, Dst: 192.168.10.1
  ...
    Protocol: ICMP (1)
    Source: 192.168.10.2
    Destination: 192.168.10.1
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
   ...
```

On voit que la source est ma machine.

On voit aussi que c'est un ping normal, on voit que c'est un ICMP de type 8

Sur l'interface eth0 :

```
No.     Time           Source                Destination           Protocol Length Info
     64 28.524932774   10.1.1.1              10.1.1.254            OpenVPN  126
MessageType:
...
    Protocol: UDP (17)
     Source: 10.1.1.1
    Destination: 10.1.1.254
User Datagram Protocol, Src Port: 1194, Dst Port: 1194
    Source Port: 1194
    Destination Port: 1194
OpenVPN Protocol
    Type: 0x45 [opcode/key_id]
    Session ID: 92967658324032
    HMAC: 01179dc0a80a02c0a80a0108007be7744f000130
    Packet-ID: 464149760
  ...
[Malformed Packet: OpenVPN]
```

On voit que sur l'interface ethernet, le paquet est transmis par UDP aux IP normales. Il y a un entete VPN avec l'ID de session ou même l'ID de paquet.

A chaque paquet, on nous dit que le paquet est mal formé

2.4 J'utilise telnet sur un serveur

```
telnet telehack.com
```

```
No.     Time           Source                Destination           Protocol Length Info
     87 2.492837628    10.1.1.1              64.13.139.230         TELNET   77
Telnet Data ...

Frame 87: 77 bytes on wire (616 bits), 77 bytes captured (616 bits) on interface
0
Internet Protocol Version 4, Src: 10.1.1.1, Dst: 64.13.139.230
    Protocol: TCP (6)
    Source: 10.1.1.1
    Destination: 64.13.139.230
Transmission Control Protocol, Src Port: 60374, Dst Port: 23, Seq: 40, Ack: 1112,
Len: 11
    Source Port: 60374
    TCP payload (11 bytes)
Telnet
    Command Suboption
...
```

On voit que rien n'est protégé, rien n'est chiffré

3) Ajout d'une clé de chiffrement partagée

1) On crée une clé partagée sur le serveur (firewall) avec la commande suivante :

```
openvpn --genkey --secret static.key
```

Le fichier static.key contient une clé de chiffrement de 2048 bits

2) Pour transférer la clé sur le client de façon sécurisée on utilise la commande scp :

```
scp static.key root@10.214.3.2:/root/static.key
```

3) Maintenant on se déplace dans le répertoire /root et on relance le VPN entre les deux machines en ajoutant --secret /Chemin_vers_clé

```
root@debian:~# openvpn --remote 10.214.4.254 --dev tun0 --ifconfig
192.168.10.2 192.168.10.1 --secret static.key
```

```
Fri Oct 18 13:17:41 2019 disabling NCP mode (--ncp-disable) because not in
P2MP client or server mode
Fri Oct 18 13:17:41 2019 OpenVPN 2.4.7 x86_64-pc-linux-gnu [SSL (OpenSSL)]
[LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Feb 20 2019
Fri Oct 18 13:17:41 2019 library versions: OpenSSL 1.1.1c  28 May 2019, LZO
2.10
Fri Oct 18 13:17:41 2019 WARNING: INSECURE cipher with block size less than
128 bit (64 bit).  This allows attacks like SWEET32.  Mitigate by using a --cipher
with a larger block size (e.g. AES-256-CBC).
Fri Oct 18 13:17:41 2019 WARNING: INSECURE cipher with block size less than
128 bit (64 bit).  This allows attacks like SWEET32.  Mitigate by using a --cipher
with a larger block size (e.g. AES-256-CBC).
Fri Oct 18 13:17:41 2019 TUN/TAP device tun0 opened
Fri Oct 18 13:17:41 2019 /sbin/ip link set dev tun0 up mtu 1500
Fri Oct 18 13:17:41 2019 /sbin/ip addr add dev tun0 local 192.168.10.2 peer
192.168.10.1
Fri Oct 18 13:17:41 2019 TCP/UDP: Preserving recently used remote address:
[AF_INET]10.214.4.254:1194
Fri Oct 18 13:17:41 2019 UDP link local (bound): [AF_INET][undef]:1194
Fri Oct 18 13:17:41 2019 UDP link remote: [AF_INET]10.214.4.254:1194
Fri Oct 18 13:17:49 2019 Peer Connection Initiated with
[AF_INET]10.214.4.254:1194
Fri Oct 18 13:17:50 2019 WARNING: this configuration may cache passwords in
memory -- use the auth-nocache option to prevent this
Fri Oct 18 13:17:50 2019 Initialization Sequence Completed
```

Pour vérifier le bon fonctionnement du réseau on ping les adresses dans le VPN :

```
root@debian:~# ping 192.168.10.1
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data.
64 bytes from 192.168.10.1: icmp_seq=1 ttl=64 time=2.85 ms
64 bytes from 192.168.10.1: icmp_seq=2 ttl=64 time=2.71 ms
64 bytes from 192.168.10.1: icmp_seq=3 ttl=64 time=2.37 ms
^C
--- 192.168.10.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 7ms
rtt min/avg/max/mdev = 2.374/2.646/2.852/0.209 ms
```

4) Pour gagner de la bande passante on ajoute `--comp-lzo --keepalive 10 60 --float` à la commande précédente :

```
root@debian:~# openvpn --remote 10.214.4.254 --dev tun0 --ifconfig
192.168.10.2 192.168.10.1 --secret static.key --comp-lzo --keepalive 10 60 –
float
```

```
Fri Oct 18 13:27:01 2019 disabling NCP mode (--ncp-disable) because not in
P2MP client or server mode
Fri Oct 18 13:27:01 2019 OpenVPN 2.4.7 x86_64-pc-linux-gnu [SSL (OpenSSL)]
[LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Feb 20 2019
Fri Oct 18 13:27:01 2019 library versions: OpenSSL 1.1.1c  28 May 2019, LZO
2.10
Fri Oct 18 13:27:01 2019 WARNING: INSECURE cipher with block size less than
128 bit (64 bit).  This allows attacks like SWEET32.  Mitigate by using a --cipher
with a larger block size (e.g. AES-256-CBC).
Fri Oct 18 13:27:01 2019 WARNING: INSECURE cipher with block size less than
128 bit (64 bit).  This allows attacks like SWEET32.  Mitigate by using a --cipher
with a larger block size (e.g. AES-256-CBC).
Fri Oct 18 13:27:01 2019 TUN/TAP device tun0 opened
Fri Oct 18 13:27:01 2019 /sbin/ip link set dev tun0 up mtu 1500
Fri Oct 18 13:27:01 2019 /sbin/ip addr add dev tun0 local 192.168.10.2 peer
192.168.10.1
Fri Oct 18 13:27:01 2019 TCP/UDP: Preserving recently used remote address:
[AF_INET]10.214.4.254:1194
Fri Oct 18 13:27:01 2019 UDP link local (bound): [AF_INET][undef]:1194
Fri Oct 18 13:27:01 2019 UDP link remote: [AF_INET]10.214.4.254:1194
Fri Oct 18 13:27:01 2019 Peer Connection Initiated with
[AF_INET]10.214.4.254:1194
Fri Oct 18 13:27:03 2019 WARNING: this configuration may cache passwords in
memory -- use the auth-nocache option to prevent this
Fri Oct 18 13:27:03 2019 Initialization Sequence Completed
```

LZO est un certain algorithme de compression, donc s'il compresse, moins de données sont transmises et la bande passante est moins utilisée (d'où le –comp dans la commande, pour compression)

5) On crée le fichier de configuration pour le client et le serveur

Client :

```
dev tun
remote 10.1.1.254
ifconfig 192.168.10.2 192.168.10.1
secret /home/adrian/static.key
comp-lzo
keepalive 10 60
float
```

Pour le serveur, on enlève juste la ligne de l'ip passerelle

Serveur :

```
dev tun
```

```
ifconfig 192.168.10.2 192.168.10.1
secret /root/static.key
comp-lzo
keepalive 10 60
float
```

On arrive bien à lancer via les fichiers

```
root@Adrian-PC:/home/adrian# openvpn open
Fri Oct 18 22:43:33 2019 disabling NCP mode (--ncp-disable) because not in
P2MP client or server mode
Fri Oct 18 22:43:33 2019 OpenVPN 2.4.4 x86_64-pc-linux-gnu [SSL
...
```