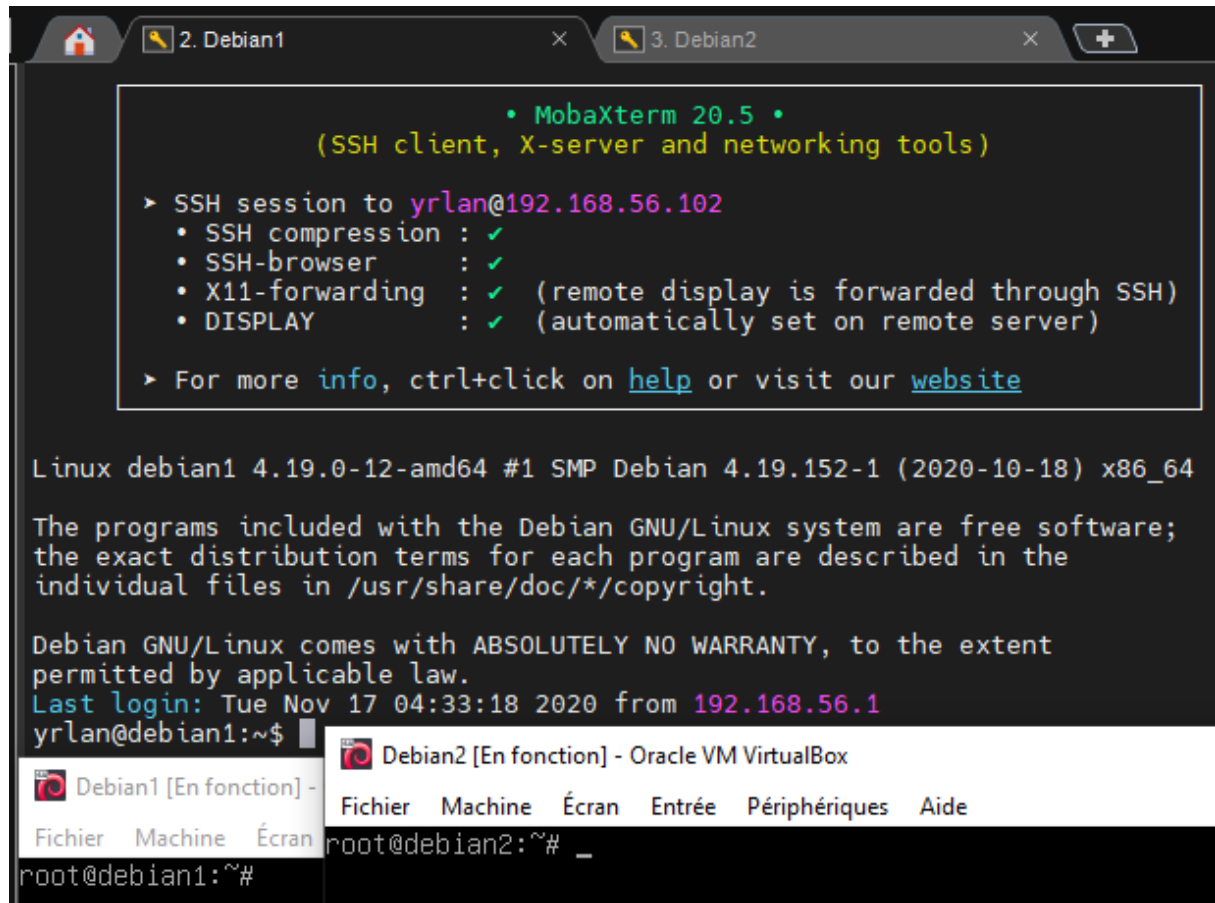


Rendu TP3 – Partie 2

B. Mise en place du laboratoire :



```
• MobaXterm 20.5 •
(SSH client, X-server and networking tools)

> SSH session to yrlan@192.168.56.102
  • SSH compression : ✓
  • SSH-browser      : ✓
  • X11-forwarding   : ✓ (remote display is forwarded through SSH)
  • DISPLAY          : ✓ (automatically set on remote server)

> For more info, ctrl+click on help or visit our website

Linux debian1 4.19.0-12-amd64 #1 SMP Debian 4.19.152-1 (2020-10-18) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Nov 17 04:33:18 2020 from 192.168.56.1
yrlan@debian1:~$
```

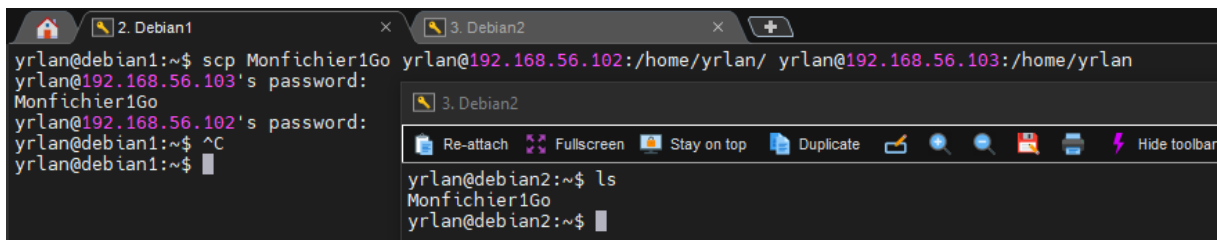
C. Transfert sécurisé avec scp :

```
Last login: Tue Nov 17 04:33:18 2020 from 192.168.56.1
yrlan@debian1:~$ pws
-bash: pws: command not found
yrlan@debian1:~$ ls -sh
total 0
yrlan@debian1:~$ pwd
/home/yrlan
yrlan@debian1:~$ dd if=/dev/zero of=Monfichier1Go bs=1 count=1000000
1000000+0 records in
1000000+0 records out
1000000 bytes (1.0 MB, 977 KiB) copied, 1.64381 s, 608 kB/s
yrlan@debian1:~$ ls -sh
total 980K
980K Monfichier1Go
yrlan@debian1:~$
```

1-2-3-4.

```
yrlan@debian2:~$ pwd
/home/yrlan
yrlan@debian2:~$
```

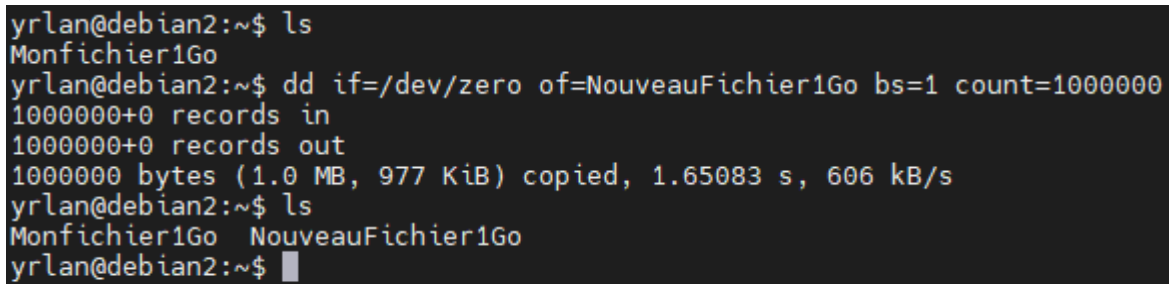
5.



```
yrlan@debian1:~$ scp Monfichier1Go yrlan@192.168.56.102:/home/yrlan/ yrlan@192.168.56.103:/home/yrlan
yrlan@192.168.56.103's password:
Monfichier1Go
yrlan@192.168.56.102's password:
yrlan@debian1:~$ ^C
yrlan@debian1:~$

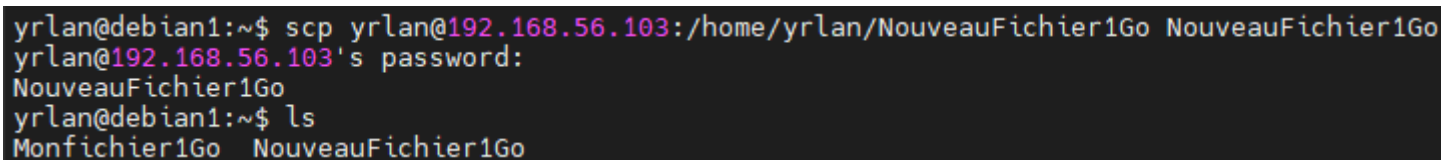
yrlan@debian2:~$ ls
Monfichier1Go
yrlan@debian2:~$
```

6-7-8. (avec ls -sh,



```
yrlan@debian2:~$ ls
Monfichier1Go
yrlan@debian2:~$ dd if=/dev/zero of=NouveauFichier1Go bs=1 count=1000000
1000000+0 records in
1000000+0 records out
1000000 bytes (1.0 MB, 977 KiB) copied, 1.65083 s, 606 kB/s
yrlan@debian2:~$ ls
Monfichier1Go  NouveauFichier1Go
yrlan@debian2:~$
```

9-10.

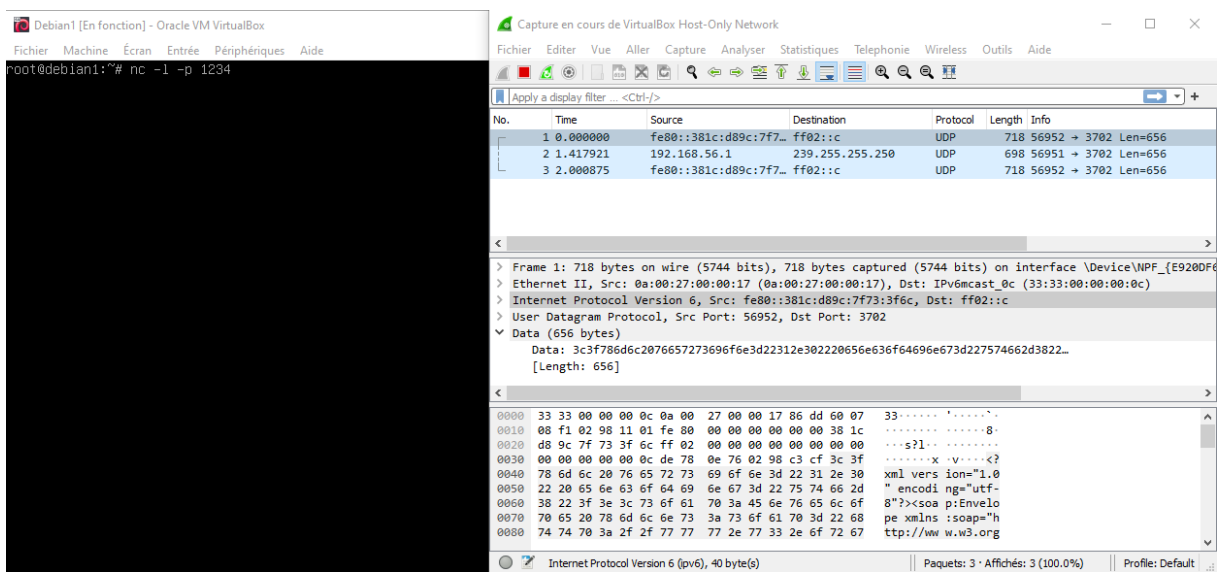


```
yrlan@debian1:~$ scp yrlan@192.168.56.103:/home/yrlan/NouveauFichier1Go NouveauFichier1Go
yrlan@192.168.56.103's password:
NouveauFichier1Go
yrlan@debian1:~$ ls
Monfichier1Go  NouveauFichier1Go
```

Pour supprimer les deux fichiers, on fais rm

D. Analyse de la communication avec Netcat :

1-2.



Debian1 [En fonction] - Oracle VM VirtualBox

root@debian1:~# nc -l -p 1234

Capture en cours de VirtualBox Host-Only Network

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	fe80::381c:d89c:7f7...	ff02::c	UDP	718	56952 → 3702 Len=656
2	1.417921	192.168.56.1	239.255.255.250	UDP	698	56951 → 3702 Len=656
3	2.000875	fe80::381c:d89c:7f7...	ff02::c	UDP	718	56952 → 3702 Len=656

Frame 1: 718 bytes on wire (5744 bits), 718 bytes captured (5744 bits) on interface \Device\NPF_{E9280F6...}

Ethernet II, Src: 0a:00:27:00:00:17 (0a:00:27:00:00:17), Dst: IPv6mcast_0c (33:33:00:00:00:0c)

Internet Protocol Version 6, Src: fe80::381c:d89c:7f73:3f6c, Dst: ff02::c

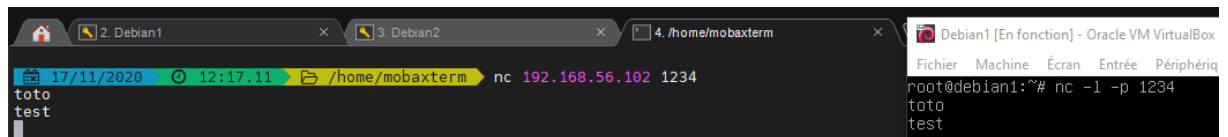
User Datagram Protocol, Src Port: 56952, Dst Port: 3702

Data (656 bytes)

Data: 3c3f786d6c20766572736966e3d22312e302220656e636f64696e673d227574662d3822... [Length: 656]

Internet Protocol Version 6 (IPv6), 40 byte(s) | Paquets: 3 · Affichés: 3 (100.0%) | Profil: Default

3-4.



6. Voici 3-way handshake du protocole TCP

4	118.369717	192.168.56.1	192.168.56.102	TCP	66 55739 → 1234 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
5	118.370424	192.168.56.102	192.168.56.1	TCP	66 1234 → 55739 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=64
6	118.370966	192.168.56.1	192.168.56.102	TCP	54 55739 → 1234 [ACK] Seq=1 Ack=1 Win=262656 Len=0

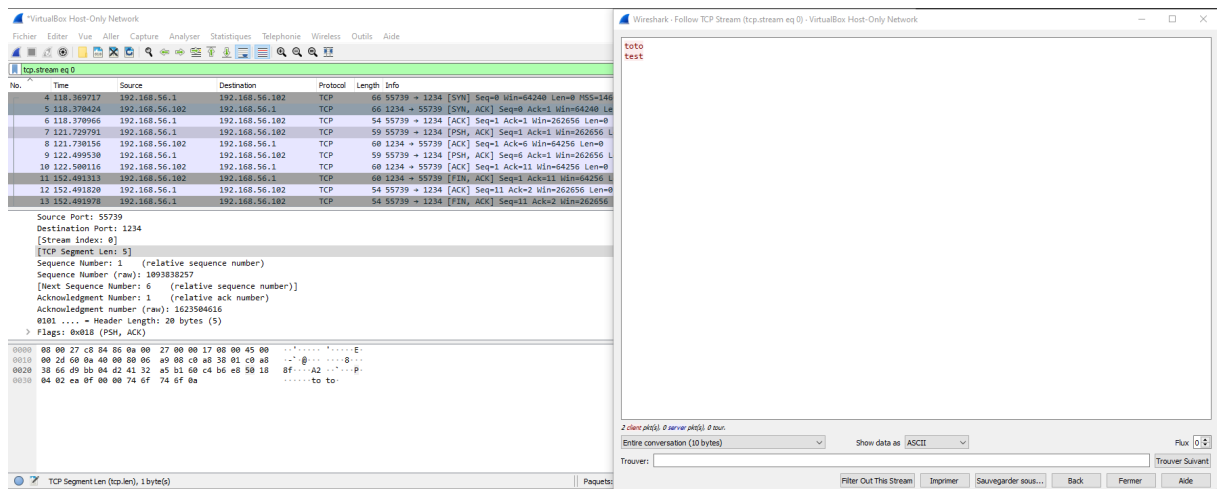
Selon le protocole de communication TCP, la connexion entre deux hôtes se fait en trois étapes : on appelle ça le « Three-way Handshake » ; voici ces 3 étapes :

SYN : Le client qui désire établir une connexion avec un serveur va envoyer un premier paquet SYN (Synchronized) au serveur.

SYN-ACK : Le serveur va répondre au client à l'aide d'un paquet SYN-ACK (Synchronize, Acknowledge).

ACK : Le client va envoyer un paquet ACK au serveur qui servira d'accusé de réception (Acknowledge).

7. On fait analyser → Suivre → TCP, on peut retrouver le message directement, Netcat n'est donc pas sécurisé



8.9. On refait la même chose mais on précise -u pour que le protocole soit en UDP, ce n'est toujours pas sécurisé car on retrouve les messages avec analyser → Suivre → UDP

