

UNIVERSITÉ NATIONALE DU VIETNAM À HANOÏ

INSTITUT FRANCOPHONE INTERNATIONAL



CONCEPTION ET ARCHITECTURE DES RÉSEAUX

RAPPORT DE

TRAVAUX PRATIQUES SUR LES OUTILS RÉSEAUX

Juin 2019

Étudiants :

Cleg Peter OVIL

Odelet VALCIN

Mike Arley MORIN

Afi Elolo Gisèle DEKPE

Enseignant : **M. Nguyen Hong Quang**

Année Académique : 2019 - 2020

Table des matières

1	Les informations sur les interfaces des machines au début de cette partie	4
1.1	Liste des interfaces notre machine	4
1.2	Adresse IP de notre machine	4
1.3	Adresse MAC de la carte réseau	4
1.4	Adresse et Masque du réseau	4
1.4.1	Adresse du réseau	4
1.4.2	Masque du réseau	5
1.5	Table de routage de la machine	5
1.6	Informations sur la machine d'adresse IP 112.137.140.41	5
1.7	Liste des routeurs par lesquels passent les datagrammes entre notre machine et la machine 112.137.140.41	5
1.8	Le(s) serveur(s) de nom pour les domaines fpt.com.vn et ifi.edu.vn (le domaine de l'IFI)	6
2	Explications sur la configuration des interfaces wifi sous Linux	6
3	Analyse des captures des trames réalisées	7
4	Explications sur le fonctionnement de l'outil <i>mtr</i>	9
5	Analyse détaillée du protocole TCP	11

Table des figures

1	Table de routage	5
2	Résultats obtenus nslookup	5
3	La liste des routeurs avec traceroute	5
4	Les serveurs de nom avec nslookup	6
5	Configuration de l'interface Wifi	6
6	Capture de trame	7
7	Table ARP	8
8	Fonctionnement de mtr, vu sur Wireshark	9
9	Fonctionnement de mtr	9
10	Topologie du réseau	10
11	Succession des interfaces	10
12	Capture TCP	11
13	12

1 Les informations sur les interfaces des machines au début de cette partie

1.1 Liste des interfaces notre machine

Résultat de la commande ifconfig - a

```
igore@igore:~$ ifconfig -a
enol: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether ec:f4:bb:45:37:9a txqueuelen 1000 (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
    device interrupt 20 memory 0xf7c00000-f7c20000

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 1000 (Boucle locale)
    RX packets 109800 bytes 9270148 (9.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 109800 bytes 9270148 (9.2 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.43.189 netmask 255.255.255.0 broadcast 192.168.43.255
    inet6 fe80::5f0d:6638:7fc7:259 prefixlen 64 scopeid 0x20<link>
    ether 18:cf:5e:20:1b:f0 txqueuelen 1000 (Ethernet)
    RX packets 2677096 bytes 2233840919 (2.2 GB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2236256 bytes 1838322941 (1.8 GB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

1.2 Adresse IP de notre machine

192.168.43.189

1.3 Adresse MAC de la carte réseau

18 :cf :5e :20 :1b :f0

1.4 Adresse et Masque du réseau

1.4.1 Adresse du réseau

192.168.43.0

1.4.2 Masque du réseau

255.255.255.0

1.5 Table de routage de la machine

```
igore@igore:~$ route
Table de routage IP du noyau
Destination      Passerelle      Genmask          Indic Metric Ref     Use Iface
default          _gateway        0.0.0.0          UG     600    0       0 wlp2s0
link-local       0.0.0.0         255.255.0.0      U      1000   0       0 wlp2s0
192.168.43.0     0.0.0.0         255.255.255.0    U      600    0       0 wlp2s0
```

FIGURE 1 – Table de routage

1.6 Informations sur la machine d'adresse IP 112.137.140.41

```
igore@igore:~$ nslookup 112.137.140.41
** server can't find 41.140.137.112.in-addr.arpa: NXDOMAIN
```

FIGURE 2 – Résultats obtenus nslookup

Le serveur semble ne pas être accessible.

1.7 Liste des routeurs par lesquels passent les datagrammes entre notre machine et la machine 112.137.140.41

```
igore@igore:~$ traceroute 112.137.140.41
traceroute to 112.137.140.41 (112.137.140.41), 64 hops max
 1  192.168.43.1  3.497ms  3.100ms  3.058ms
 2  * * *
 3  10.51.166.57  213.283ms  24.017ms  28.685ms
 4  10.51.140.97  30.581ms  36.171ms  20.952ms
 5  10.51.41.158  36.179ms  27.070ms  30.463ms
 6  10.51.31.237  29.648ms  29.649ms  29.866ms
 7  113.164.224.9  32.920ms  27.389ms  29.857ms
 8  * * *
 9  113.171.34.2  216.697ms  27.276ms  27.830ms
10  125.235.241.5  29.332ms  29.202ms  30.021ms
11  27.68.228.25  32.480ms  31.903ms  39.812ms
12  27.68.228.190  37.125ms  28.992ms  31.919ms
13  27.68.228.197  32.390ms  41.288ms  40.392ms
14  27.68.229.10  34.274ms  29.461ms  29.785ms
15  112.137.140.41  32.052ms  28.232ms  30.691ms
```

FIGURE 3 – La liste des routeurs avec traceroute

1.8 Le(s) serveur(s) de nom pour les domaines fpt.com.vn et ifi.edu.vn (le domaine de l'IFI)

```
igore@igore:~$ nslookup ftp.com.vn
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   ftp.com.vn
Address: 123.30.184.86

igore@igore:~$ nslookup ifi.edu.vn
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   ifi.edu.vn
Address: 112.137.140.40
Name:   ifi.edu.vn
Address: 64:ff9b::7089:8c28
```

FIGURE 4 – Les serveurs de nom avec nslookup

2 Explications sur la configuration des interfaces wifi sous Linux

```
igore@igore:~$ nano /etc/network/interfaces
Fichier Édition Affichage Rechercher Terminal Aide
GNU nano 2.9.3 /etc/network/interfaces

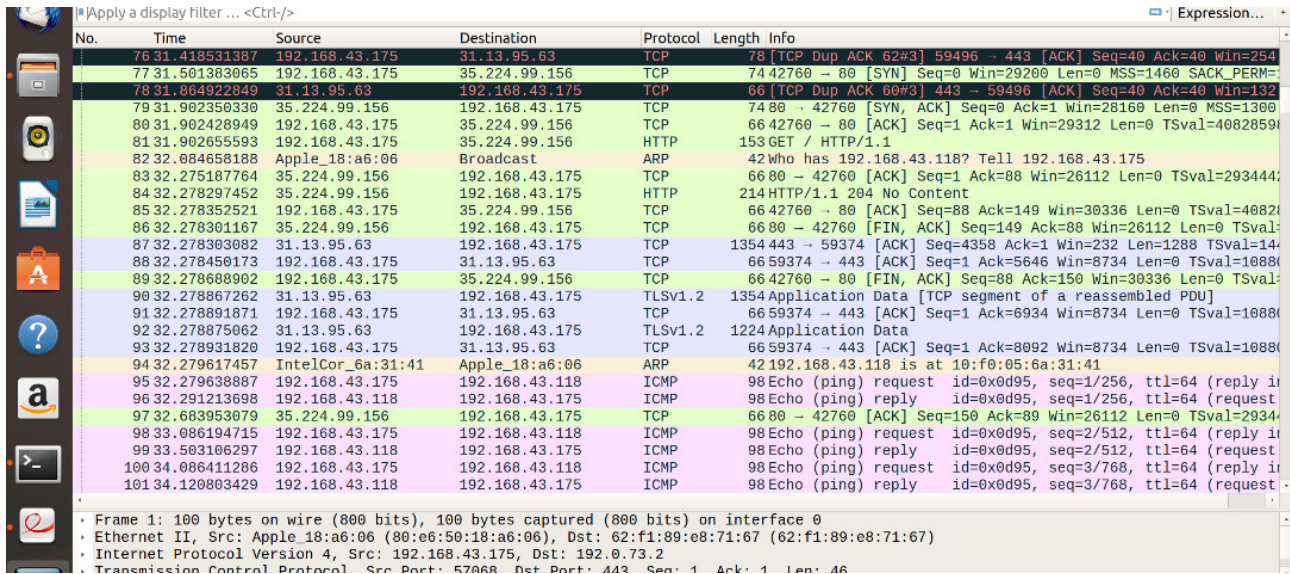
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback
```

FIGURE 5 – Configuration de l'interface Wifi

L'ajout d'adresse se fera de façon dynamique grâce à la commande "auto". Pour le faire de façon statique, il faut ajouter l'adresse ciblée.

Outils pour la capture des trames

3 Analyse des captures des trames réalisées



No.	Time	Source	Destination	Protocol	Length	Info
76	31.418531387	192.168.43.175	31.13.95.63	TCP	78	[TCP Dup ACK 62#3] 59496 → 443 [ACK] Seq=40 Ack=40 Win=254
77	31.501383065	192.168.43.175	35.224.99.156	TCP	74	42760 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=
78	31.864922849	31.13.95.63	192.168.43.175	TCP	66	[TCP Dup ACK 60#3] 443 → 59496 [ACK] Seq=40 Ack=40 Win=132
79	31.902350330	35.224.99.156	192.168.43.175	TCP	74	80 → 42760 [SYN, ACK] Seq=0 Ack=1 Win=26160 Len=0 MSS=1300
80	31.902428949	192.168.43.175	35.224.99.156	TCP	66	42760 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=4082659
81	31.902655593	192.168.43.175	35.224.99.156	HTTP	153	GET / HTTP/1.1
82	32.084658188	Apple_18:a6:06	Broadcast	ARP	42	Who has 192.168.43.118? Tell 192.168.43.175
83	32.275187764	35.224.99.156	192.168.43.175	TCP	66	80 → 42760 [ACK] Seq=1 Ack=88 Win=26112 Len=0 TSval=293444
84	32.278297452	35.224.99.156	192.168.43.175	HTTP	214	HTTP/1.1 204 No Content
85	32.278352521	192.168.43.175	35.224.99.156	TCP	66	42760 → 80 [ACK] Seq=88 Ack=149 Win=30336 Len=0 TSval=4082
86	32.278301167	35.224.99.156	192.168.43.175	TCP	66	80 → 42760 [FIN, ACK] Seq=149 Ack=88 Win=26112 Len=0 TSval=
87	32.278303082	31.13.95.63	192.168.43.175	TCP	1354	443 → 59374 [ACK] Seq=4358 Ack=1 Win=232 Len=1288 TSval=14
88	32.278450173	192.168.43.175	31.13.95.63	TCP	66	59374 → 443 [ACK] Seq=1 Ack=5646 Win=8734 Len=0 TSval=1088
89	32.278688902	192.168.43.175	35.224.99.156	TCP	66	42760 → 80 [FIN, ACK] Seq=88 Ack=150 Win=30336 Len=0 TSval=
90	32.278867262	31.13.95.63	192.168.43.175	TLSv1.2	1354	Application Data [TCP segment of a reassembled PDU]
91	32.278891871	192.168.43.175	31.13.95.63	TCP	66	59374 → 443 [ACK] Seq=1 Ack=6934 Win=8734 Len=0 TSval=1088
92	32.278875062	31.13.95.63	192.168.43.175	TLSv1.2	1224	Application Data
93	32.278931820	192.168.43.175	31.13.95.63	TCP	66	59374 → 443 [ACK] Seq=1 Ack=8092 Win=8734 Len=0 TSval=1088
94	32.279617457	IntelCor_6a:31:41	Apple_18:a6:06	ARP	42	192.168.43.118 is at 10:f0:05:6a:31:41
95	32.279638887	192.168.43.175	192.168.43.118	ICMP	98	Echo (ping) request id=0x0d95, seq=1/256, ttl=64 (reply in
96	32.291213698	192.168.43.118	192.168.43.175	ICMP	98	Echo (ping) reply id=0x0d95, seq=1/256, ttl=64 (request
97	32.683953079	35.224.99.156	192.168.43.175	TCP	66	80 → 42760 [ACK] Seq=150 Ack=89 Win=26112 Len=0 TSval=2934
98	33.086194715	192.168.43.175	192.168.43.118	ICMP	98	Echo (ping) request id=0x0d95, seq=2/512, ttl=64 (reply in
99	33.503106297	192.168.43.118	192.168.43.175	ICMP	98	Echo (ping) reply id=0x0d95, seq=2/512, ttl=64 (request
100	34.086411286	192.168.43.175	192.168.43.118	ICMP	98	Echo (ping) request id=0x0d95, seq=3/768, ttl=64 (reply in
101	34.120803429	192.168.43.118	192.168.43.175	ICMP	98	Echo (ping) reply id=0x0d95, seq=3/768, ttl=64 (request

Frame 1: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0
 Ethernet II, Src: Apple 18:a6:06 (80:e6:50:18:a6:06), Dst: 62:f1:89:e8:71:67 (62:f1:89:e8:71:67)
 Internet Protocol Version 4, Src: 192.168.43.175, Dst: 192.0.73.2
 Transmission Control Protocol, Src Port: 57668, Dst Port: 443, Seq: 1, Ack: 1, Len: 46

FIGURE 6 – Capture de trame


```
64 bytes from 192.168.43.118: icmp_seq=4 ttl=64 time=257 ms
64 bytes from 192.168.43.118: icmp_seq=5 ttl=64 time=82.9 ms
64 bytes from 192.168.43.118: icmp_seq=6 ttl=64 time=508 ms
^C
--- 192.168.43.118 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5005ms
rtt min/avg/max/mdev = 32.999/248.686/508.443/169.523 ms
root@192:/usr/bin# arp -a
_gateway (192.168.43.1) at 62:f1:89:e8:71:67 [ether] on wlp3s0
? (192.168.43.18) at <incomplete> on wlp3s0
ignore (192.168.43.189) at 18:cf:5e:20:1b:f0 [ether] on wlp3s0
odelet-Latitude-3440 (192.168.43.206) at 48:51:b7:6e:5f:6e [ether] on wlp3s0
ovil (192.168.43.118) at 10:f0:05:6a:31:41 [ether] on wlp3s0
root@192:/usr/bin# arp -d 192.168.43.118
root@192:/usr/bin# arp -a
_gateway (192.168.43.1) at 62:f1:89:e8:71:67 [ether] on wlp3s0
? (192.168.43.18) at <incomplete> on wlp3s0
ignore (192.168.43.189) at 18:cf:5e:20:1b:f0 [ether] on wlp3s0
odelet-Latitude-3440 (192.168.43.206) at 48:51:b7:6e:5f:6e [ether] on wlp3s0
root@192:/usr/bin# ping 192.168.43.118
PING 192.168.43.118 (192.168.43.118) 56(84) bytes of data.
64 bytes from 192.168.43.118: icmp_seq=1 ttl=64 time=206 ms
64 bytes from 192.168.43.118: icmp_seq=2 ttl=64 time=416 ms
64 bytes from 192.168.43.118: icmp_seq=3 ttl=64 time=34.4 ms
64 bytes from 192.168.43.118: icmp_seq=4 ttl=64 time=58.1 ms
64 bytes from 192.168.43.118: icmp_seq=5 ttl=64 time=485 ms
^C
--- 192.168.43.118 ping statistics ---
6 packets transmitted, 5 received, 16% packet loss, time 5005ms
rtt min/avg/max/mdev = 34.429/240.349/485.666/183.361 ms
root@192:/usr/bin# arp -a
_gateway (192.168.43.1) at 62:f1:89:e8:71:67 [ether] on wlp3s0
? (192.168.43.18) at <incomplete> on wlp3s0
ignore (192.168.43.189) at 18:cf:5e:20:1b:f0 [ether] on wlp3s0
odelet-Latitude-3440 (192.168.43.206) at 48:51:b7:6e:5f:6e [ether] on wlp3s0
ovil (192.168.43.118) at 10:f0:05:6a:31:41 [ether] on wlp3s0
root@192:/usr/bin#
```

FIGURE 7 – Table ARP

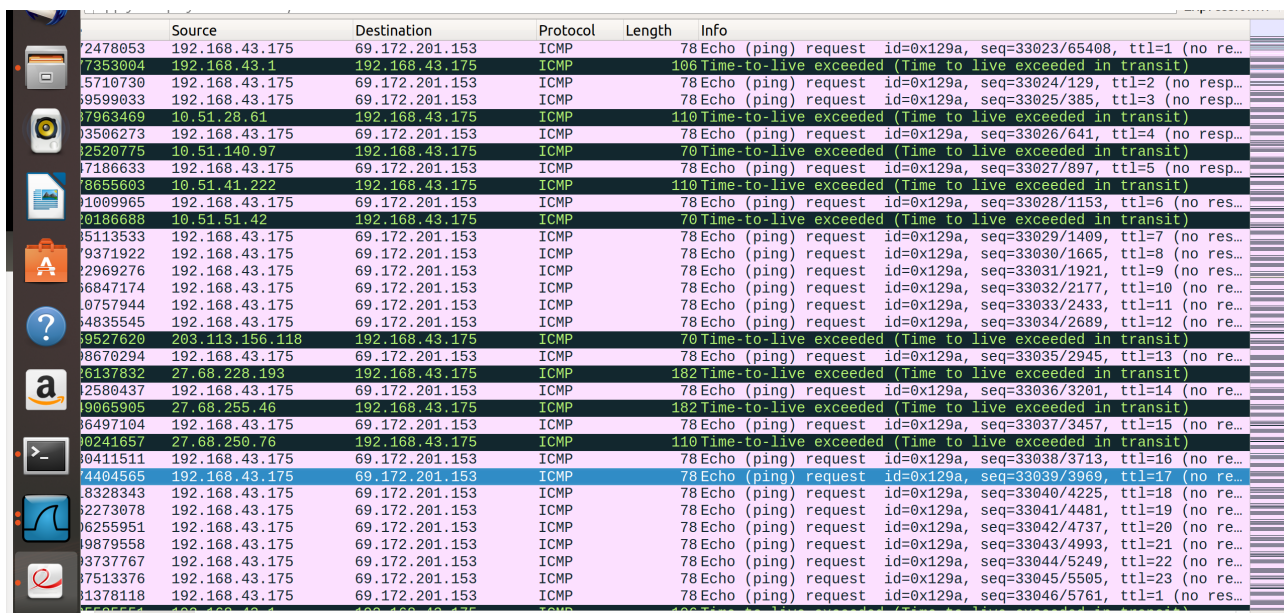
Interprétation des résultats obtenus : Nous avons effectué un test avec la commande **arp - a**, ce qui nous a donné une liste avec deux (02) utilisateurs (ignore et odelet). Nous avons effectué un ping vers une troisième machine dont l'adresse est **192.168.43.118** (voir figure 8).

Sur la figure 7, on remarque à la ligne **82** que le ping demande le propriétaire de l'adresse en question (who has...). A la ligne **94**, on obtient la réponse avec l'adresse MAC de la machine ciblée.

On revient à la figure 8, vers la fin, on exécute de nouveau la commande **arp - a**, on remarque que le nombre d'hôtes est passé est trois (03) : **ignore, odelet et ovil**.

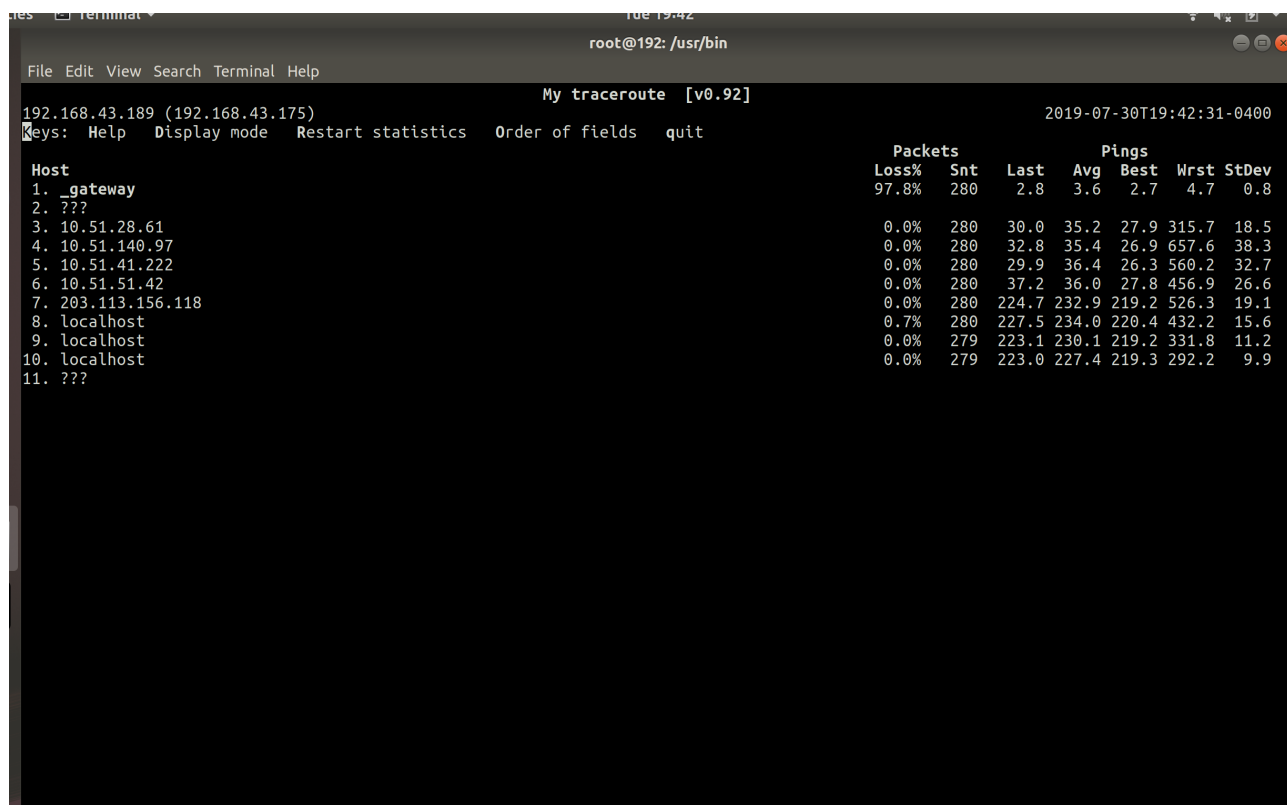
Analyse des routes suivies par les paquets (l'outil mtr)

4 Explications sur le fonctionnement de l'outil *mtr*

A screenshot of the Wireshark network protocol analyzer showing a capture of mtr traffic. The packet list on the left shows various ICMP Echo (ping) requests and Time-to-live exceeded messages. The packet details pane on the right shows the structure of an ICMP Echo request, including the type, code, identifier, and sequence number. The packet bytes pane on the right shows the raw data of the packet.

No.	Source	Destination	Protocol	Length	Info
2478053	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33023/65408, ttl=1 (no re...
7353804	192.168.43.1	192.168.43.175	ICMP	106	Time-to-live exceeded (Time to live exceeded in transit)
15710730	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33024/129, ttl=2 (no resp...
9599033	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33025/385, ttl=3 (no resp...
7963469	10.51.28.61	192.168.43.175	ICMP	110	Time-to-live exceeded (Time to live exceeded in transit)
3506273	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33026/641, ttl=4 (no resp...
2520775	10.51.140.97	192.168.43.175	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
7186633	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33027/897, ttl=5 (no resp...
8655603	10.51.41.222	192.168.43.175	ICMP	110	Time-to-live exceeded (Time to live exceeded in transit)
1009965	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33028/1153, ttl=6 (no res...
8186688	10.51.51.42	192.168.43.175	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
5113533	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33029/1409, ttl=7 (no res...
9371922	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33030/1665, ttl=8 (no res...
2969276	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33031/1921, ttl=9 (no res...
6847174	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33032/2177, ttl=10 (no re...
8757944	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33033/2433, ttl=11 (no re...
4835545	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33034/2689, ttl=12 (no re...
9527620	203.113.156.118	192.168.43.175	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
8670294	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33035/2945, ttl=13 (no re...
76137832	27.68.228.193	192.168.43.175	ICMP	182	Time-to-live exceeded (Time to live exceeded in transit)
2580437	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33036/3201, ttl=14 (no re...
9065905	27.68.255.46	192.168.43.175	ICMP	182	Time-to-live exceeded (Time to live exceeded in transit)
6497104	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33037/3457, ttl=15 (no re...
10241657	27.68.250.76	192.168.43.175	ICMP	110	Time-to-live exceeded (Time to live exceeded in transit)
10411511	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33038/3713, ttl=16 (no re...
4404565	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33039/3969, ttl=17 (no re...
8328343	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33040/4225, ttl=18 (no re...
7273078	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33041/4481, ttl=19 (no re...
6255951	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33042/4737, ttl=20 (no re...
9879558	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33043/4993, ttl=21 (no re...
93737767	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33044/5249, ttl=22 (no re...
7513376	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33045/5505, ttl=23 (no re...
1378118	192.168.43.175	69.172.201.153	ICMP	78	Echo (ping) request id=0x129a, seq=33046/5761, ttl=1 (no res...

FIGURE 8 – Fonctionnement de mtr, vu sur Wireshark

A screenshot of a terminal window showing the output of the mtr command. The terminal displays a list of hosts and their IP addresses, along with statistics for each host, including loss percentage, sent packets, last RTT, average RTT, best RTT, worst RTT, and standard deviation.

```
root@192: /usr/bin
File Edit View Search Terminal Help
My traceroute [v0.92]
192.168.43.189 (192.168.43.175) 2019-07-30T19:42:31-0400
Keys: Help Display mode Restart statistics Order of fields quit

Host
1. _gateway
2. ???
3. 10.51.28.61
4. 10.51.140.97
5. 10.51.41.222
6. 10.51.51.42
7. 203.113.156.118
8. localhost
9. localhost
10. localhost
11. ???

Packets
Loss% Snt Last Avg Best Wrst StDev
97.8% 280 2.8 3.6 2.7 4.7 0.8

Pings
Loss% Snt Last Avg Best Wrst StDev
0.0% 280 30.0 35.2 27.9 315.7 18.5
0.0% 280 32.8 35.4 26.9 657.6 38.3
0.0% 280 29.9 36.4 26.3 560.2 32.7
0.0% 280 37.2 36.0 27.8 456.9 26.6
0.0% 280 224.7 232.9 219.2 526.3 19.1
0.7% 280 227.5 234.0 220.4 432.2 15.6
0.0% 279 223.1 230.1 219.2 331.8 11.2
0.0% 279 223.0 227.4 219.3 292.2 9.9
```

FIGURE 9 – Fonctionnement de mtr

La topologie du réseau et la succession des interfaces traversées nos paquets :

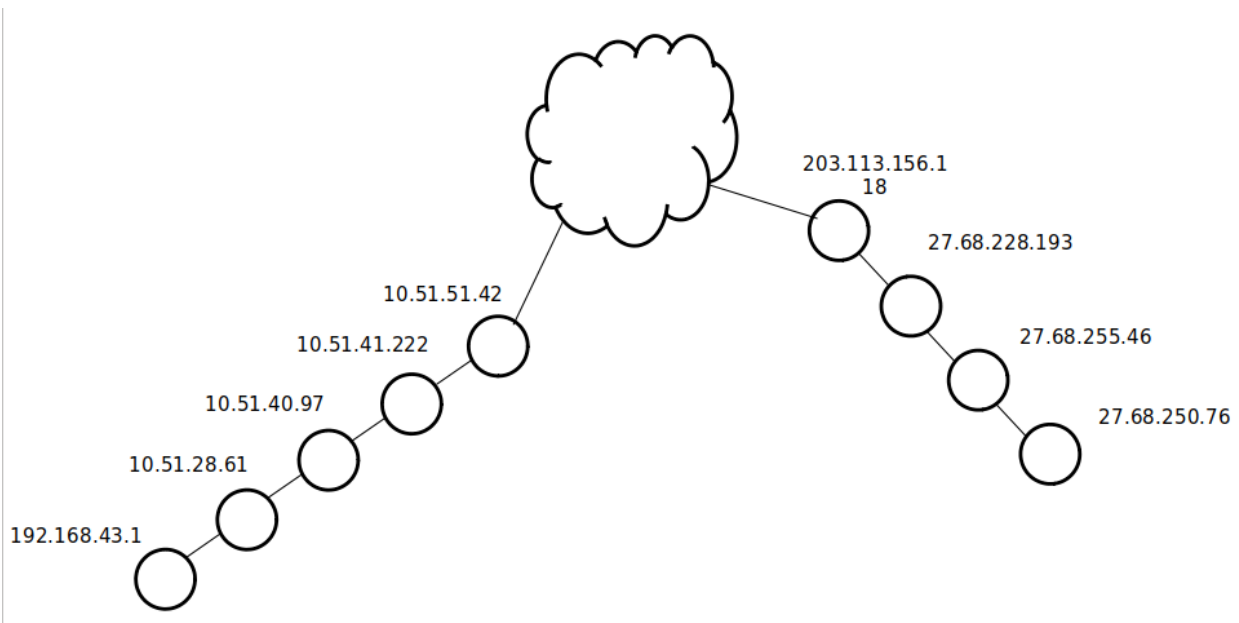


FIGURE 10 – Topologie du réseau

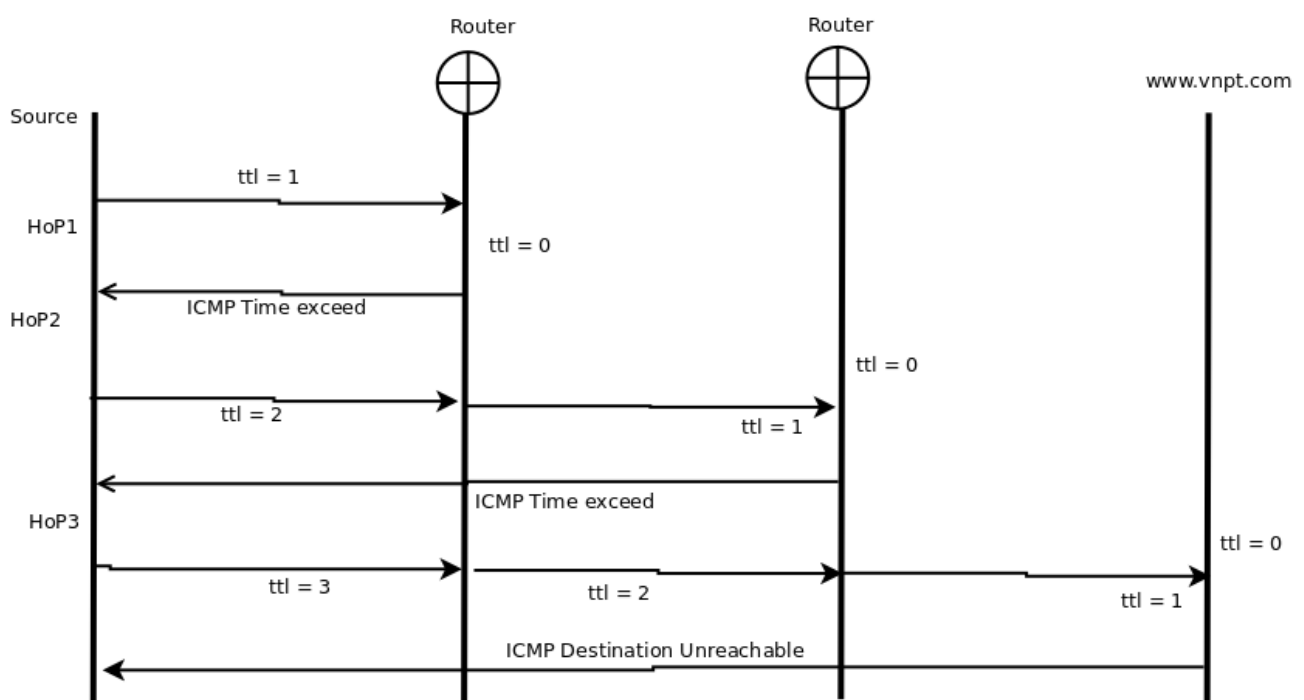


FIGURE 11 – Succession des interfaces

- 192.168.43.175 -> 192.168.43.1 -> 10.51.28.61
- 10.51.140.97 -> 10.51.41.222 -> 10.51.51.42
- 203.113.156.118 -> 27.68.228.193 -> 27.68.255.46 27.68.250.76

Fonctionnement de MTR

mtr envoie un paquet avec TTL=1 de la source a la destination, lorsque le premier routeur reçoit le paquet il réduit la valeur TTL de 1 ($1-1$) = 0 puis supprime le paquet et

envoi un message ICMP “temps dépasse” a la source et dresse une liste de l’adresse du routeur si la cible n’est pas encore atteinte in va envoyer un autre paquet avec le champ $TTL = TTL + 1$ jusqu’a ce qu’il trouve la cible.

5 Analyse détaillée du protocole TCP

Commande : `tcpdump -i wlp380 -w paquet.pcap`

Apply a display filter ... <Ctrl-/>

Expression...

No.	Time	Source	Destination	Protocol	Length	Info
10.000000	192.168.43.175	192.168.43.1	DNS	74	Standard query 0xf68a AAAA fad.ifi.edu.vn	
20.165578	192.168.43.1	192.168.43.175	DNS	74	Standard query response 0xf68a AAAA fad.ifi.edu.vn	
30.165922	192.168.43.175	112.137.140.42	TCP	74	47206 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SA	
40.213862	112.137.140.42	192.168.43.175	TCP	74	80 → 47206 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 M	
50.213899	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval	
60.214006	192.168.43.175	112.137.140.42	HTTP	256	GET /ififad/file.php/28/documents/WS_user-guide-a4	
70.263105	112.137.140.42	192.168.43.175	TCP	66	80 → 47206 [ACK] Seq=1 Ack=191 Win=6912 Len=0 TSva	
80.747102	112.137.140.42	192.168.43.175	HTTP	822	HTTP/1.1 303 See Other (text/html)	
90.747124	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=191 Ack=757 Win=30720 Len=0 T	
100.747398	192.168.43.175	112.137.140.42	HTTP	309	GET /ififad/login/index.php HTTP/1.1	
110.797503	112.137.140.42	192.168.43.175	TCP	66	80 → 47206 [ACK] Seq=757 Ack=434 Win=7936 Len=0 TS	
120.156627	112.137.140.42	192.168.43.175	TCP	1354	80 → 47206 [ACK] Seq=757 Ack=434 Win=7936 Len=1288	
130.157239	112.137.140.42	192.168.43.175	TCP	1354	80 → 47206 [ACK] Seq=2045 Ack=434 Win=7936 Len=128	
140.157262	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=434 Ack=3333 Win=35968 Len=0	
150.157341	112.137.140.42	192.168.43.175	TCP	1354	80 → 47206 [ACK] Seq=3333 Ack=434 Win=7936 Len=128	
160.198562	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=434 Ack=4621 Win=38528 Len=0	
170.215901	112.137.140.42	192.168.43.175	TCP	1354	80 → 47206 [ACK] Seq=4621 Ack=434 Win=7936 Len=128	
180.215940	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=434 Ack=5909 Win=41888 Len=0	
190.217327	112.137.140.42	192.168.43.175	TCP	1354	80 → 47206 [ACK] Seq=5909 Ack=434 Win=7936 Len=128	
200.217343	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=434 Ack=7197 Win=43648 Len=0	
210.220846	112.137.140.42	192.168.43.175	TCP	1354	80 → 47206 [ACK] Seq=7197 Ack=434 Win=7936 Len=128	
220.220892	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=434 Ack=8485 Win=46208 Len=0	
230.248796	112.137.140.42	192.168.43.175	TCP	1354	80 → 47206 [ACK] Seq=8485 Ack=434 Win=7936 Len=128	
240.248815	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=434 Ack=9773 Win=48768 Len=0	
250.250213	112.137.140.42	192.168.43.175	TCP	1354	80 → 47206 [ACK] Seq=9773 Ack=434 Win=7936 Len=128	
260.250223	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=434 Ack=11061 Win=51328 Len=0	
270.250833	112.137.140.42	192.168.43.175	HTTP	741	HTTP/1.1 200 OK (text/html)	
280.250847	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=434 Ack=11736 Win=54016 Len=0	
290.251657	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [FIN, ACK] Seq=434 Ack=11736 Win=54016	
300.284879	112.137.140.42	192.168.43.175	TCP	66	80 → 47206 [FIN, ACK] Seq=11736 Ack=435 Win=7936 L	
310.284901	192.168.43.175	112.137.140.42	TCP	66	47206 → 80 [ACK] Seq=435 Ack=11737 Win=54016 Len=0	
320.498621	192.168.43.175	31.13.95.63	TLSv1.2	97	Application Data	
330.557488	31.13.95.63	192.168.43.175	TCP	66	443 → 80 [ACK] Seq=1 Ack=32 Win=458 Len=0 TSval	

FIGURE 12 – Capture TCP

Phase de connexion

- La source envoie un SYN a la destination pour initialiser la connexion
- La destination répond avec [SYN, ACK] pour dire je suis prêt, j’accepte la connexion
- La source envoie un ACK pour dire ok je vais commencer
- La source fait la requête du document qu’il veut telecharger
- La destination envoie les paquets

Phase de connexion No 3, 4, 5

Phase de transfert 6 -> 27

Phase de déconnexion 28 -> 31

Les numéros initiaux SN utilisés dans les deux sens Seq=0 seq=0

Analyse du protocole Telnet et la capture des informations

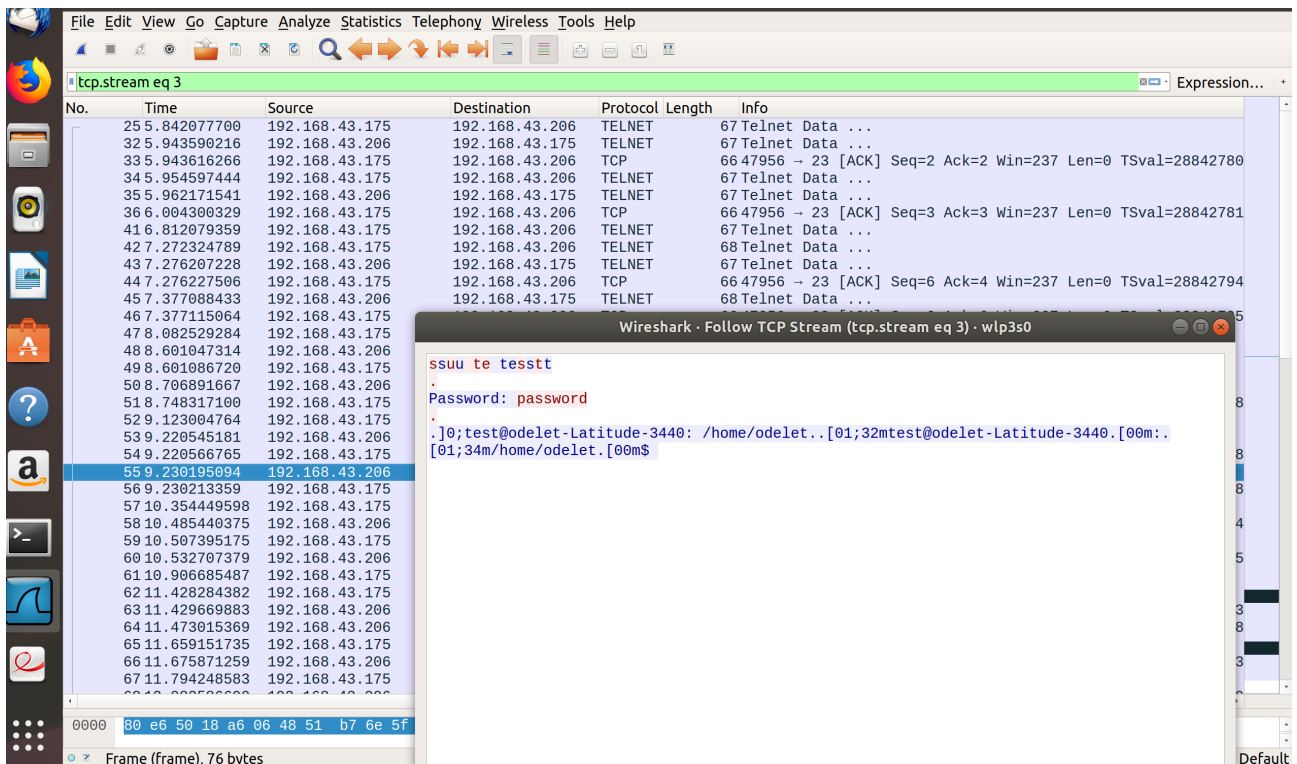


FIGURE 13 –

Port source et destination : respectivement **23** et **479 56**. On cherche le mot de passe et le username en faisant un clic droit sur l'un des paquets **Follow -> Tcp stream**.

La capture des paquets de Telnet nous montre que c'est pas un protocole sécurisé, on peut sniffer les informations confidentielles exemple *mot de passe* et *username* sont transférés en clair.

Références

- Cours de Conception et Architecture des Réseaux, M. Nguyen Hong Quang
- [https ://www.wireshark.org/](https://www.wireshark.org/)