

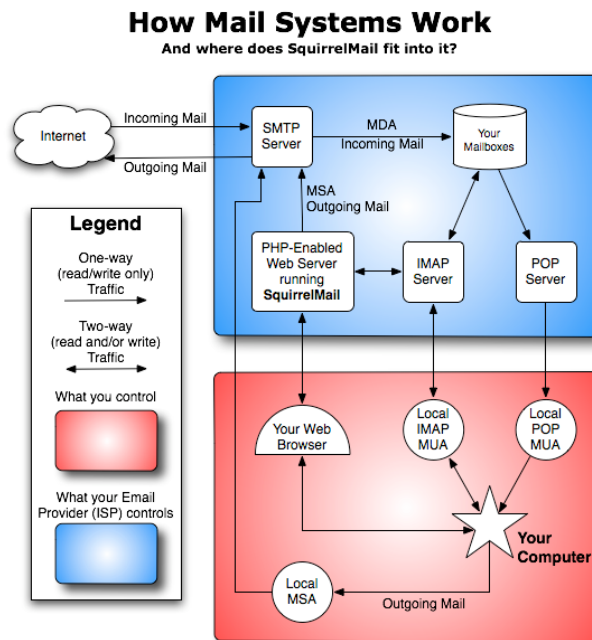


TP-Protocoles-SMTP-POP3-IMAP4

20.01.2024

Le mail c'est en mode **TCP** connecté... ☺

Auteur : Pascal Fougeray



Source : Je ne sais plus ☺

Le but du TP n'était pas de faire du mail!!! mais de **voir ce qu'est un protocole et TCP**

1 Préambule

- Ce TP peut être fait chez vous, il n'y a aucune difficulté majeure, il ne va pas vous occuper 2h30! ☺
- Ce TP utilise la VM simplement et le logiciel Wireshark
- Il suffit juste d'installer 3 serveurs de mails...
- On va voir la "**lourdeur**" de TCP et voir les **ACK** et **SYN**!!!
- Peut-être que vous n'aurez pas le temps de tout faire, si SMTP est fait c'est déjà bien
- **Prenez des notes sur ce que vous comprenez, ces notes vous y aurez le droit de les avoir avec vous au CT!**

2 Introduction

Dans ce TP, je vous propose de **voir** ce qu'est :

- 1 protocole en mode connecté, ici 3 différents SMTP, POP3 et IMAP4 de communication pour les mails
- **Installer** 3 serveurs de mails rapidement
- **Utiliser** la commande **netstat**
- **Capturer** avec wireshark un échange SMTP



- **Capturer** avec wireshark un échange POP3
- **Capturer** avec wireshark un échange IMPA4
- **Bien comprendre TCP**

Le mail c'est en mode connecté!!!

3 Voir et parler un protocole

L'objectif est de "parler" un protocole d'application

Pour cela vous allez saisir les commandes en mode texte et recevoir les réponses à la console.

ATTENTION : vous parlez avec un serveur qui n'est pas INTELLIGENT!!!

Vous êtes connecté à lui et c'est à vous de vous adapter!!!

On va choisir

- le protocole SMTP pour envoyer un mail
- les protocoles IMAP et POP pour lire un mail
- D'envoyer un mail anonyme et voir que l'adresse de l'expéditeur n'est pas vérifiée!

Pas question de :

- Envoyer des images, vidéos ou autre chose...!
- D'utiliser votre compte personnel de mails...!

À partir de maintenant on travaille que dans la VM!

Remarque : Si vous ne faites que la partie 3.1 SMTP c'est suffisant, les parties 3.2 Lire avec Pop3 et 3.3 Lire avec Imap4 c'est identique d'un point de vue principe. C'est à dire qu'on installe un serveur qui écoute un port spécifique 110 pour le protocole POP et 143 pour le protocole IMAP et qu'il s'appuie tous les 2 sur la couche 4 TCP.

Donc si tout marche du premier coup et que vous êtes en avance et que vous avez très très très très... envie de savoir faites les 3.2 et 3.3 sinon et bien on passe au TP suivant.

3.1 SMTP

1. **Lancez** wireshark et **sélectionnez** l'interface **localhost** de la VM
2. **Sélectionnez** le protocole SMTP
3. **Ouvrez** un terminal
4. **Lancez** la commande **netstat -lnpt4**
Le port 25 est-il ouvert?
Non ?, allez il faut installer un serveur SMTP
5. **Installez** le serveur SMTP Postfix : **apt install postfix**
Ça ne marche pas ? Vous êtes root ? Et oui pour installer un serveur il faut être root ☺
(a) **Choisissez** le choix **local uniquement** comme indiqué sur la figure suivante!!!
(b) **Faites ok** ou **valeur par défaut** pour le reste.

```

Postfix Configuration
: de votre serveur de messagerie la plus adaptée à vos besoins.

configuration actuelle inchangée.
t directement en SMTP.
en SMTP ou grâce à un utilitaire comme fetchmail. Les messages sortants son
une autre machine, nommée un smarthost.
: pour les utilisateurs locaux. Il n'y a pas de mise en réseau.
agerie :

Pas de configuration
Site Internet
Internet avec un « smarthost »
Système satellite
Local uniquement
<Ok>                                <Annuler>

```

6. **Lancez** la commande **netstat -lnpt4**
Le port 25 est-il ouvert?
Oui ?, On peut envoyer un mail et "**parler**" le SMTP avec le serveur.



3.1.1 On envoie un mail

ATTENTION à chaque commande, allez voir ce qui se passe entre vous qui êtes le client et le serveur en regardant les segments ajoutés sur Wireshark !

1. **Relevez** l'@IP de l'interface enp0s9.

C'est l'interface reliée à l'interface VBoxnet0 du HOST. Elle doit avoir une @IP **192.168.56.xxx**/24
Si pas d'@IP, **lancez** la commande **dhclient enp0s9** et **vérifiez** !

2. **Connectez** vous au serveur SMTP en lançant la commande : **telnet localhost 25 -b 192.168.56.xxx**
Le serveur doit se présenter

```
etudiant@debian-11-GNS3:~$ telnet localhost 25 -b 192.168.56.131
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 debian-12-GNS3.lan ESMTP Postfix (Debian/GNU)
```

3. **Remarquez** le nom du serveur : **debian-12-GNS3.lan** <- ça c'est chez moi à la fac c'est peut-être même surement différent !

4. Que voit-on sur Wireshark ?

5. **Soyez poli** et **dites** lui bonjour : **HELO Bonjour je suis un etudiant-e**

6. **Que voit-on sur Wireshark ?**

7. **Identifiez** l'émetteur du mail : **MAIL FROM : <mbappe@fft.fr>**

ATTENTION : pas d'espace et le : plus les <> doivent être tapés ici et pas d'accents ni caractères étendus dans l'adresse mail !

Et oui, c'est mbappe qui envoie le mail ☺, si vous préférez mettre une autre personne, vous pouvez ☺

8. **Que voit-on sur Wireshark ?**

9. **Identifiez** le récepteur du mail : **RCPT TO : <etudiant@debian-12-GNS3.lan>**

ATTENTION : ici il faut mettre une vraie adresse d'un vrai compte, donc on prend le compte étudiant.

10. Que voit-on sur Wireshark ?

11. On va écrire notre mail : **DATA**

12. **Que voit-on sur Wireshark ?**

13. **Tapez** votre mail : **Cher étudiant que vous êtes bon, si bon que vous pouvez aller chercher un café pour le prof ☺**

Vous tapez ce que vous voulez et pour dire que votre mail est terminé il faut :

14. **Que voit-on sur Wireshark ?**

15. **Terminez** par une ligne contenant seulement : un . <— ceci est un point ☺

16. **Que voit-on sur Wireshark ?**

17. **Quittez** le serveur, donc **déconnectez** vous : **QUIT**

18. **Que voit-on sur Wireshark ?**

Vous devez avoir quelque chose comme cela sur votre console.

```
etudiant@debian-11-GNS3:~$ telnet localhost 25 -b 192.168.56.131
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 debian-11-GNS3.lan ESMTP Postfix (Debian/GNU)
HELO Bonjour je suis un etudiant
250 debian-11-GNS3.lan
MAIL FROM:<mbappe@fft.fr>
250 2.1.0 Ok
RCPT TO:<etudiant@debian-11-GNS3.lan>
250 2.1.5 Ok
```



DATA

354 End data with <CR><LF>.<CR><LF>

Cher étudiant que vous êtes bon, si bon que vous pouvez aller chercher un café pour le prof
.

250 2.0.0 Ok: queued as A96A840F67

QUIT

221 2.0.0 Bye

Connection closed by foreign host.

3.1.2 Vérification

1. Si tout s'est bien passé dans le meilleur des mondes 😊, **allez** dans le répertoire **/var/mail**
2. **Regardez** à quel groupe appartient le fichier **etudiant**

```
etudiant@debian-11-GNS3:/var/mail$ ls -al
total 12
-rw----- 1 etudiant mail 495 16 janv. 13:01 etudiant
```

3. **Essayez** de le supprimer : **rm etudiant**
Et oui... vous ne pouvez pas, il appartient au serveur. **Groupe mail!!!**
4. **Lisez** le contenu du fichier **etudiant**

```
etudiant@debian-11-GNS3:~$ cd /var/mail/
etudiant@debian-11-GNS3:/var/mail$ ls
etudiant
etudiant@debian-11-GNS3:/var/mail$ cat etudiant
```

```
From mbappe@fft.fr Mon Jan 16 12:27:45 2023
Return-Path: <mbappe@fft.fr>
X-Original-To: etudiant@debian-11-GNS3.lan
Delivered-To: etudiant@debian-11-GNS3.lan
Received: from Bonjour?je?suis?un?etudiant (unknown [192.168.56.131])
    by debian-11-GNS3.lan (Postfix) with SMTP id A96A840F67
    for <etudiant@debian-11-GNS3.lan>; Mon, 16 Jan 2023 12:26:57 +0100 (CET)
```

Cher étudiant que vous êtes bon, si bon que vous pouvez aller chercher un café pour le prof

5. **Expliquez** ce que vous comprenez par rapport à ce que vous avez fait.

3.1.3 Analyse du protocole

Expliquez ce que vous relevez dans Wireshark

1. Avec **seulement** le protocole SMTP de choisi

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.56.131	192.168.56.131	SMTP	117	220 debian-11-GNS3.lan ESMTP Postfix (Debian/GNU)
5	54.6473671	192.168.56.131	192.168.56.131	SMTP	100	C: HELO Bonjour je suis un etudiant
8	54.6479233	192.168.56.131	192.168.56.131	SMTP	90	S: 250 debian-11-GNS3.lan
63	7107146	192.168.56.131	192.168.56.131	SMTP	93	C: MAIL FROM:<mbappe@fft.fr>
63	7299656	192.168.56.131	192.168.56.131	SMTP	80	S: 250 2.1.0 Ok
75	5142900	192.168.56.131	192.168.56.131	SMTP	105	C: RCPT TO:<etudiant@debian-11-GNS3.lan>
75	5344154	192.168.56.131	192.168.56.131	SMTP	80	S: 250 2.1.5 Ok
84	3340132	192.168.56.131	192.168.56.131	SMTP	72	C: DATA
84	3341391	192.168.56.131	192.168.56.131	SMTP	103	S: 354 End data with <CR><LF>.<CR><LF>
92	3772761	192.168.56.131	192.168.56.131	SMTP	105	C: DATA fragment, 39 bytes
102	264211	192.168.56.131	192.168.56.131	SMTP	102	C: DATA fragment, 96 bytes
110	739039	192.168.56.131	192.168.56.131	SMT...	69	RCPT TO:<etudiant@debian-11-GNS3.lan> , Cher étudiant que vous êtes bon, si bon que vo
110	755475	192.168.56.131	192.168.56.131	SMTP	102	S: 250 2.0.0 Ok: queued as A545440F67
113	114945	192.168.56.131	192.168.56.131	SMTP	72	C: QUIT
113	115402	192.168.56.131	192.168.56.131	SMTP	81	S: 221 2.0.0 Bye

2. Avec le protocole TCP de choisi



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.56.131	127.0.0.1	TCP	74	54473 → 25 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval=3566448907 TSecr=0 WS=128
2	0.000012181	127.0.0.1	192.168.56.131	TCP	74	25 → 54473 [SYN, ACK] Seq=1 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM=1 TSval=2732366176 TSecr=3566448907
3	0.000023341	192.168.56.131	127.0.0.1	TCP	60	54473 → 25 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=3566448907 TSecr=2732366176
4	0.000057469	127.0.0.1	192.168.56.131	SMTP	117	S: 250 debian-11-GNS3.lan ESMTP Postfix (Debian/GNU)
5	0.000075358	192.168.56.131	127.0.0.1	TCP	60	54473 → 25 [ACK] Seq=1 Ack=52 Win=65536 Len=0 TSval=3566448916 TSecr=2732366185
6	54.6473671	192.168.56.131	127.0.0.1	SMTP	100	C: HELLO Bonjour je suis un etudiant
7	54.6474134	127.0.0.1	192.168.56.131	TCP	60	25 → 54473 [ACK] Seq=52 Ack=35 Win=65536 Len=0 TSval=2732420824 TSecr=3566503554
8	54.6479233	127.0.0.1	192.168.56.131	SMTP	90	S: 250 debian-11-GNS3.lan
9	54.6480177	192.168.56.131	127.0.0.1	TCP	60	54473 → 25 [ACK] Seq=35 Ack=76 Win=65536 Len=0 TSval=3566503555 TSecr=2732420824
10	63.7197146	192.168.56.131	127.0.0.1	SMTP	93	C: MAIL FROM:<mbappe@fft.fr>
11	63.7299056	127.0.0.1	192.168.56.131	TCP	80	S: 250 2.1.0 Ok
12	63.7300253	192.168.56.131	127.0.0.1	TCP	60	54473 → 25 [ACK] Seq=62 Ack=90 Win=65536 Len=0 TSval=3566512637 TSecr=2732429906
13	75.5142909	192.168.56.131	127.0.0.1	SMTP	105	C: RCPT TO:<etudiant@debian-11-GNS3.lan>
14	75.5344154	127.0.0.1	192.168.56.131	SMTP	80	S: 250 2.1.5 Ok
15	75.5344330	192.168.56.131	127.0.0.1	TCP	60	54473 → 25 [ACK] Seq=101 Ack=104 Win=65536 Len=0 TSval=3566524442 TSecr=2732441711
16	84.3340132	192.168.56.131	127.0.0.1	SMTP	72	C: DATA
17	84.3341391	192.168.56.131	127.0.0.1	SMTP	103	S: 354 End data with <CR><LF>.<CR><LF>
18	84.3341553	192.168.56.131	127.0.0.1	TCP	60	54473 → 25 [ACK] Seq=107 Ack=141 Win=65536 Len=0 TSval=3566533241 TSecr=2732450510
19	92.3772761	192.168.56.131	127.0.0.1	SMTP	105	C: DATA Fragment, 39 bytes
20	92.4203582	127.0.0.1	192.168.56.131	TCP	60	25 → 54473 [ACK] Seq=141 Ack=146 Win=65536 Len=0 TSval=2732458596 TSecr=3566541284
21	102.264211	192.168.56.131	127.0.0.1	SMTP	102	C: DATA Fragment, 96 bytes
22	102.264253	127.0.0.1	192.168.56.131	TCP	60	25 → 54473 [ACK] Seq=141 Ack=242 Win=65536 Len=0 TSval=2732468440 TSecr=3566551171
23	110.739039	192.168.56.131	127.0.0.1	SMTP	65	RCPT TO:<etudiant@debian-11-GNS3.lan> , Cher etudiant que vous êtes bon, si bon que vous pouvez aller
24	110.739095	127.0.0.1	192.168.56.131	TCP	60	25 → 54473 [ACK] Seq=141 Ack=245 Win=65536 Len=0 TSval=2732476915 TSecr=3566559646
25	110.755475	127.0.0.1	192.168.56.131	SMTP	102	S: 250 2.0.0 Ok: queued as A545440F67
26	110.755499	192.168.56.131	127.0.0.1	TCP	60	54473 → 25 [ACK] Seq=245 Ack=177 Win=65536 Len=0 TSval=3566559663 TSecr=2732476932
27	113.114945	192.168.56.131	127.0.0.1	SMTP	72	C: QUIT
28	113.115402	127.0.0.1	192.168.56.131	SMTP	81	S: 221 2.0.0 Bye
29	113.115464	192.168.56.131	127.0.0.1	TCP	60	54473 → 25 [ACK] Seq=251 Ack=192 Win=65536 Len=0 TSval=3566562023 TSecr=2732479292
30	113.115522	127.0.0.1	192.168.56.131	TCP	60	25 → 54473 [FIN, ACK] Seq=192 Ack=251 Win=65536 Len=0 TSval=2732479292 TSecr=3566562023
31	113.115973	192.168.56.131	127.0.0.1	TCP	60	54473 → 25 [FIN, ACK] Seq=251 Ack=193 Win=65536 Len=0 TSval=3566562023 TSecr=2732479292
32	113.116057	127.0.0.1	192.168.56.131	TCP	60	25 → 54473 [ACK] Seq=193 Ack=252 Win=65536 Len=0 TSval=2732479292 TSecr=3566562023

3. **Expliquez** ce que sont tous ces ACK, SYN en plus, en **relisant** le cours TCP vs UDP sur ecampus.
4. **Concluez sur ce qu'est un protocole !!!**
5. **Pourquoi le mail c'est en mode connecté donc TCP et non en UDP? <- Question qui peut tomber au CT ☺**

On peut... si vous avez tout fait vite et qu'on n'est pas en retard... faire la même chose avec les 2 autres protocoles POP et IMAP

Je ne vais pas détailler comme pour SMTP, mais la méthode est la même, bien voir ce qui se passe sur wireshark !

3.2 Lire avec POP3

1. **Installez** le serveur SMTP Postfix : **apt install popa3d**
Ça ne marche pas ? Vous êtes root ?
2. **Lancez** la commande **netstat -lnpt4**
Le port 110 est-il ouvert ?
Oui ? , On peut lire un mail et “**parler**” avec le serveur.
3. **Connectez** vous au serveur POP en lançant la commande : **telnet localhost 110 -b 192.168.56.xxx**
Le serveur doit se présenter
etudiant@debian-11-GNS3:\$ telnet localhost 110 -b 192.168.56.131
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
+OK
4. **Identifiez** le lecteur
 - (a) **user etudiant**
 - (b) **pass Etudiant1**
5. **Regardez** sur Wireshark, c'est cool le MDP en clair.. **imaginez un pirate ;)**
6. **Voir** ses mails : **stat**
Donne le nombre de messages présent dans la file d'attente, ainsi que le volume total des messages en octets
7. **Lister** les dossiers de la boîte mail : **list**
Donne la liste des messages en attente, avec pour chaque message
— * Son numéro d'ordre dans la file
— * Sa taille en octets
8. Lire le premier message : **top 1 1**
top x y permet de récupérer les **x** premières lignes du message **y**, *wahou c'est vachement pratique* ☺
Les lignes d'en-tête ne sont pas comptabilisées.



9. Déconnectez : **QUIT**

Le résultat :

```

etudiant@debian-11-GNS3:$ telnet localhost 110 -b 192.168.56.131
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
+OK
user etudiant
+OK
pass Etudiant1
+OK
stat
+OK 1 419
list
+OK
1 419
.
top 1 1
+OK
Return-Path: <mbappe@fft.fr>
X-Original-To: etudiant@debian-11-GNS3.lan
Delivered-To: etudiant@debian-11-GNS3.lan
Received: from Bonjour?je?suis?un?etudiant (unknown [192.168.56.131])
    by debian-11-GNS3.lan (Postfix) with SMTP id CFF2140BF8
    for <etudiant@debian-11-GNS3.lan>; Mon, 16 Jan 2023 13:09:16 +0100 (CET)

```

Cher étudiant que vous êtes bon, si bon que vous pouvez aller chercher un café pour le prof

QUIT

+OK

Connection closed by foreign host.

3.2.1 Analyse du protocole

Expliquez ce que vous relevez dans Wireshark

1. Avec **seulement** le protocole POP de choisi

No.	Time	Source	Destination	Protocol	Length	Info
4	0.001128449	127.0.0.1	192.168.56.131	POP	71	S: +OK
5	0.002033177	192.168.56.131	192.168.56.131	POP	71	C: user etudiant
6	7.932589081	127.0.0.1	192.168.56.131	POP	71	S: +OK
7	28.8802946...	192.168.56.131	192.168.56.131	POP	82	C: pass Etudiant1
8	28.9094480...	192.168.56.131	192.168.56.131	POP	71	S: +OK
9	50.4738408...	192.168.56.131	192.168.56.131	POP	72	C: stat
10	50.4740457...	192.168.56.131	192.168.56.131	POP	77	S: +OK 1 419
11	56.9019153...	192.168.56.131	192.168.56.131	POP	72	C: list
12	56.9021878...	192.168.56.131	192.168.56.131	POP	71	S: +OK
13	56.9022944...	192.168.56.131	192.168.56.131	POP	76	S: top 1 1
14	65.0162426...	192.168.56.131	192.168.56.131	POP	71	S: +OK
15	65.0163361...	192.168.56.131	192.168.56.131	POP	485	C: Cher étudiant que vous êtes bon, si bon que vous pouvez aller chercher un café pour le prof
16	65.0170190...	192.168.56.131	192.168.56.131	POP	69	S: +OK
17	69.3016839...	192.168.56.131	192.168.56.131	POP	72	C: QUIT
18	69.3019379...	192.168.56.131	192.168.56.131	POP	71	S: +OK

2. Avec le protocole TCP de choisi

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.56.131	127.0.0.1	TCP	74	54263 → 110 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval=3568533610 TSecr=0 WS=128
2	0.000011781	127.0.0.1	192.168.56.131	TCP	74	110 → 54263 [ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM=1 TSval=2734450879 TSecr=0
3	0.000023566	192.168.56.131	127.0.0.1	TCP	66	54263 → 110 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=3568533610 TSecr=2734450879
4	0.001128449	127.0.0.1	192.168.56.131	POP	71	S: +OK
5	0.002033177	192.168.56.131	192.168.56.131	POP	66	54263 → 110 [ACK] Seq=1 Ack=6 Win=65536 Len=0 TSval=3568533611 TSecr=2734450880
6	7.932589081	127.0.0.1	192.168.56.131	TCP	66	110 → 54263 [ACK] Seq=6 Ack=16 Win=65536 Len=0 TSval=2734450811 TSecr=3568541542
7	7.932589081	127.0.0.1	192.168.56.131	POP	71	S: +OK
8	7.932641000	192.168.56.131	127.0.0.1	TCP	66	54263 → 110 [ACK] Seq=16 Ack=11 Win=65536 Len=0 TSval=3568541542 TSecr=2734450811
9	28.8802946...	192.168.56.131	192.168.56.131	POP	82	C: pass Etudiant1
10	28.9094480...	192.168.56.131	192.168.56.131	POP	71	S: +OK
11	50.4738408...	192.168.56.131	192.168.56.131	TCP	66	54263 → 110 [ACK] Seq=32 Ack=16 Win=65536 Len=0 TSval=3568562519 TSecr=2734479788
12	50.4740457...	192.168.56.131	192.168.56.131	POP	72	C: stat
13	50.4740980...	192.168.56.131	192.168.56.131	TCP	66	54263 → 110 [ACK] Seq=38 Ack=27 Win=65536 Len=0 TSval=3568584084 TSecr=2734501353
14	56.9019153...	192.168.56.131	192.168.56.131	POP	71	S: +OK
15	56.9021878...	192.168.56.131	192.168.56.131	POP	72	C: list
16	56.9022944...	192.168.56.131	192.168.56.131	POP	71	S: +OK
17	56.9023991...	192.168.56.131	192.168.56.131	TCP	66	54263 → 110 [ACK] Seq=44 Ack=32 Win=65536 Len=0 TSval=3568590512 TSecr=2734507781
18	65.0162426...	192.168.56.131	192.168.56.131	POP	76	S: top 1 1
19	65.0163361...	192.168.56.131	192.168.56.131	TCP	66	54263 → 110 [ACK] Seq=44 Ack=42 Win=65536 Len=0 TSval=3568590512 TSecr=2734507781
20	65.0166345...	192.168.56.131	192.168.56.131	TCP	71	S: +OK
21	65.0168361...	192.168.56.131	192.168.56.131	TCP	66	54263 → 110 [ACK] Seq=53 Ack=47 Win=65536 Len=0 TSval=3568598626 TSecr=2734515895
22	65.0168917...	192.168.56.131	192.168.56.131	TCP	485	C: Cher étudiant que vous êtes bon, si bon que vous pouvez aller chercher un café pour le prof
23	65.0170190...	192.168.56.131	192.168.56.131	POP	66	54263 → 110 [ACK] Seq=53 Ack=466 Win=65152 Len=0 TSval=3568598626 TSecr=2734515895
24	65.0170273...	192.168.56.131	192.168.56.131	POP	66	54263 → 110 [ACK] Seq=53 Ack=469 Win=65152 Len=0 TSval=3568598627 TSecr=2734515896
25	69.3016839...	192.168.56.131	192.168.56.131	POP	72	C: QUIT
26	69.3019379...	192.168.56.131	192.168.56.131	POP	71	S: +OK
27	69.3019727...	192.168.56.131	192.168.56.131	TCP	66	54263 → 110 [ACK] Seq=59 Ack=474 Win=65536 Len=0 TSval=3568602912 TSecr=2734520181
28	69.3026212...	192.168.56.131	192.168.56.131	TCP	66	110 → 54263 [FIN, ACK] Seq=474 Ack=59 Win=65536 Len=0 TSval=2734520181 TSecr=3568602912
29	69.3028623...	192.168.56.131	192.168.56.131	TCP	66	54263 → 110 [FIN, ACK] Seq=59 Ack=475 Win=65536 Len=0 TSval=3568602912 TSecr=2734520181
30	69.3029046...	192.168.56.131	192.168.56.131	TCP	66	110 → 54263 [ACK] Seq=475 Ack=60 Win=65536 Len=0 TSval=2734520181 TSecr=3568602912



3. **Expliquez** ce que sont tous ces ACK, SYN en plus, en **relisant** et oui encore ... le cours TCP vs UDP sur ecampus.

3.3 Lire avec IMAP4

ATTENTION imap c'est lourd comme protocole

1. **Installez** le serveur SMTP Postfix : **apt install dovecot-imapd**
Ça ne marche pas ? Vous êtes root ?
2. **Lancez** la commande **netstat -lnpt4**
Le port 143 est-il ouvert ?
Oui ? , On peut lire un mail et "**parler**" avec le serveur.
3. **Identifiez** le lecteur
(a) **1 login étudiant Etudiant1** <- OUI OUI il faut taper le 1 au début puis 2 etc ...!!!
4. **Regardez** sur Wireshark, c'est cool le MDP en clair... **imaginez** un pirate ;)
5. **Lister** les dossiers de la boîte mail : **2 list "" "*" <- OUI OUI il faut taper le 2 au début puis 3 etc ...!!!**
6. Voir les mails :
7. **3 examine inbox**
8. **Lire** le premier message : **4 fetch 1 body[]** , *wahou c'est vachement pratique 😊*
9. **Déconnectez** : **5 logout**

Vous devez voir quelque chose comme ça

```
etudiant@debian-11-GNS3:$ telnet localhost 143 -b 192.168.56.131
```

```
Trying ::1...
```

```
Connected to localhost.
```

```
Escape character is '^['.
```

```
* OK [CAPABILITY IMAP4rev1 SASL-IR LOGIN-REFERRALS ID ENABLE IDLE LITERAL+ STARTTLS AUTH=PLAIN] Dove
```

```
1 login étudiant Etudiant1
```

```
1 OK [CAPABILITY IMAP4rev1 SASL-IR LOGIN-REFERRALS ID ENABLE IDLE SORT SORT=DISPLAY
```

```
THREAD=REFERENCES THREAD=REFS THREAD=ORDEREDSUBJECT MULTIAPPEND URL-PARTIAL
```

```
CATENATE UNSELECT CHILDREN NAMESPACE UIDPLUS LIST-EXTENDED I18NLEVEL=1
```

```
CONDSTORE QRESYNC ESEARCH ESORT SEARCHRES WITHIN CONTEXT=SEARCH LIST-STATUS
```

```
BINARY MOVE SNIPPET=FUZZY PREVIEW=FUZZY STATUS=SIZE SAVEDATE LITERAL+ NOTIFY SPECIAL-USE] Logged in
```

```
2 list "" "*"
```

```
* LIST (\HasNoChildren) "/" INBOX
```

```
2 OK List completed (0.006 + 0.000 + 0.005 secs).
```

```
3 examine inbox
```

```
* FLAGS (\Answered \Flagged \Deleted \Seen \Draft)
```

```
* OK [PERMANENTFLAGS ()] Read-only mailbox.
```

```
* 1 EXISTS
```

```
* 1 RECENT
```

```
* OK [UNSEEN 1] First unseen.
```

```
* OK [UIDVALIDITY 1673872681] UIDs valid
```

```
* OK [UIDNEXT 2] Predicted next UID
```

```
3 OK [READ-ONLY] Examine completed (0.003 + 0.000 + 0.002 secs).
```

```
4 fetch 1 body[]
```

```
* 1 FETCH (BODY[] {419}
```

```
Return-Path: <mbappe@fft.fr>
```

```
X-Original-To: etudiant@debian-11-GNS3.lan
```

```
Delivered-To: etudiant@debian-11-GNS3.lan
```

```
Received: from Bonjour?je?suis?un?etudiant (unknown [192.168.56.131])
```

```
by debian-11-GNS3.lan (Postfix) with SMTP id CFF2140BF8
```

```
for <etudiant@debian-11-GNS3.lan>; Mon, 16 Jan 2023 13:09:16 +0100 (CET)
```

Cher étudiant que vous êtes bon, si bon que vous pouvez aller chercher un café pour le prof



```

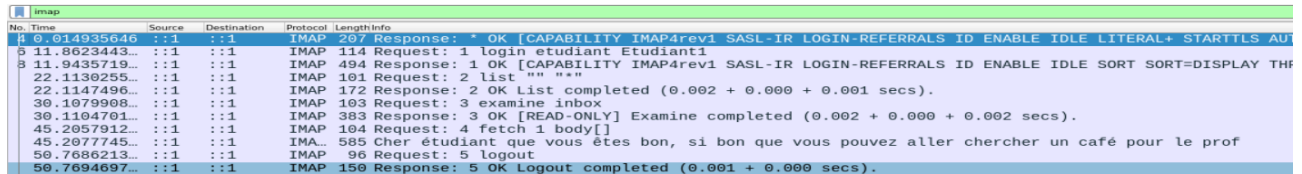
)
4 OK Fetch completed (0.002 + 0.000 + 0.001 secs).
logout
logout BAD Error in IMAP command: Invalid command name (0.001 + 0.000 secs).
5 logout
* BYE Logging out
5 OK Logout completed (0.001 + 0.000 secs).
Connection closed by foreign host.
etudiant@debian-11-GNS3:$

```

3.3.1 Analyse du protocole

Expliquez ce que vous relevez dans Wireshark

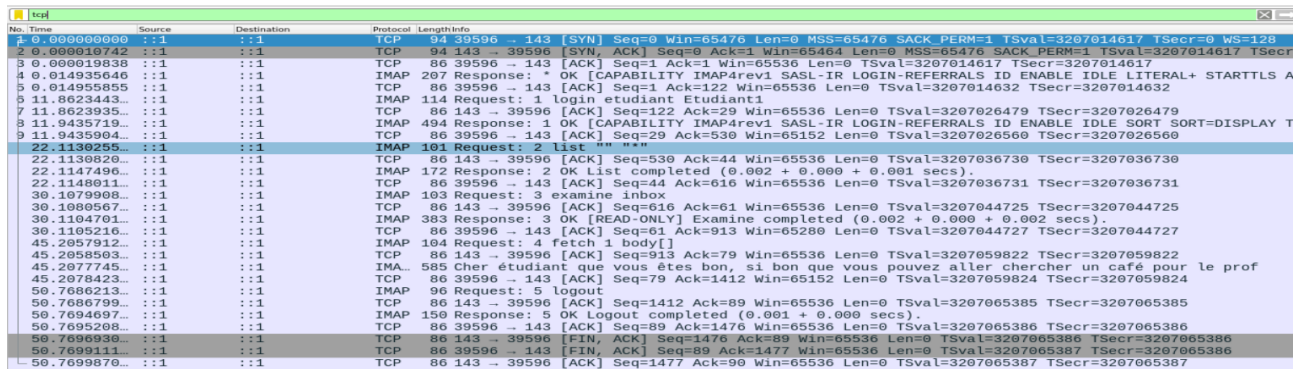
1. Avec **seulement** le protocole IMAP de choisi



Wireshark IMAP protocol filter and packet list. The filter is set to 'imap'. The packet list shows the following packets:

No.	Time	Source	Destination	Protocol	Length	Info
0	0.014935646	:::1	:::1	IMAP	207	Response: * OK [CAPABILITY IMAP4rev1 SASL-IR LOGIN-REFERRALS ID ENABLE IDLE LITERAL+ STARTTLS AUTH=PLAIN] IMAP4rev1
1	11.8623443	:::1	:::1	IMAP	114	Request: 1 login etudiant Etudiant1
2	11.9435719	:::1	:::1	IMAP	494	Response: 1 OK [CAPABILITY IMAP4rev1 SASL-IR LOGIN-REFERRALS ID ENABLE IDLE SORT SORT=DISPLAY TH
3	22.1138255	:::1	:::1	IMAP	101	Request: 2 list "" ""
4	22.1147496	:::1	:::1	IMAP	172	Response: 2 OK List completed (0.002 + 0.000 + 0.001 secs).
5	30.1079908	:::1	:::1	IMAP	103	Request: 3 examine inbox
6	30.1104701	:::1	:::1	IMAP	383	Response: 3 OK [READ-ONLY] Examine completed (0.002 + 0.000 + 0.002 secs).
7	45.2057912	:::1	:::1	IMAP	104	Request: 4 fetch 1 body[]
8	45.2077745	:::1	:::1	IMAP	585	Cher étudiant que vous êtes bon, si bon que vous pouvez aller chercher un café pour le prof
9	50.7686213	:::1	:::1	IMAP	96	Request: 5 logout
10	50.7694697	:::1	:::1	IMAP	150	Response: 5 OK Logout completed (0.001 + 0.000 secs).

2. Avec le protocole TCP de choisi



Wireshark TCP protocol filter and packet list. The filter is set to 'tcp'. The packet list shows the following packets:

No.	Time	Source	Destination	Protocol	Length	Info
0	0.000000000	:::1	:::1	TCP	94	39596 -> 143 [SYN] Seq=0 Win=65476 Len=0 MSS=65476 SACK_PERM=1 TSval=3207014617 TSecr=0 WS=128
1	0.000010742	:::1	:::1	TCP	94	143 -> 39596 [ACK] Seq=0 Ack=1 Win=65476 Len=0 MSS=65476 SACK_PERM=1 TSval=3207014617 TSecr=0 WS=128
2	0.000019838	:::1	:::1	TCP	86	39596 -> 143 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=3207014617 TSecr=3207014617
3	0.014935646	:::1	:::1	IMAP	207	Response: * OK [CAPABILITY IMAP4rev1 SASL-IR LOGIN-REFERRALS ID ENABLE IDLE LITERAL+ STARTTLS AUTH=PLAIN] IMAP4rev1
4	0.014955855	:::1	:::1	TCP	86	39596 -> 143 [ACK] Seq=1 Ack=122 Win=65536 Len=0 TSval=3207014632 TSecr=3207014632
5	11.8623443	:::1	:::1	IMAP	114	Request: 1 login etudiant Etudiant1
6	11.8623935	:::1	:::1	TCP	86	143 -> 39596 [ACK] Seq=122 Ack=29 Win=65536 Len=0 TSval=3207026479 TSecr=3207026479
7	11.9435719	:::1	:::1	IMAP	494	Response: 1 OK [CAPABILITY IMAP4rev1 SASL-IR LOGIN-REFERRALS ID ENABLE IDLE SORT SORT=DISPLAY TH
8	11.9435904	:::1	:::1	TCP	86	39596 -> 143 [ACK] Seq=29 Ack=530 Win=65152 Len=0 TSval=3207026560 TSecr=3207026560
9	22.1138255	:::1	:::1	IMAP	101	Request: 2 list "" ""
10	22.1138820	:::1	:::1	TCP	86	143 -> 39596 [ACK] Seq=530 Ack=44 Win=65536 Len=0 TSval=3207036730 TSecr=3207036730
11	22.1147496	:::1	:::1	IMAP	172	Response: 2 OK List completed (0.002 + 0.000 + 0.001 secs).
12	22.1148011	:::1	:::1	TCP	86	39596 -> 143 [ACK] Seq=44 Ack=616 Win=65536 Len=0 TSval=3207036731 TSecr=3207036731
13	30.1079908	:::1	:::1	IMAP	103	Request: 3 examine inbox
14	30.1080567	:::1	:::1	TCP	86	143 -> 39596 [ACK] Seq=616 Ack=61 Win=65536 Len=0 TSval=3207044725 TSecr=3207044725
15	30.1104701	:::1	:::1	IMAP	383	Response: 3 OK [READ-ONLY] Examine completed (0.002 + 0.000 + 0.002 secs).
16	30.1105216	:::1	:::1	TCP	86	39596 -> 143 [ACK] Seq=61 Ack=913 Win=65280 Len=0 TSval=3207044727 TSecr=3207044727
17	45.2057912	:::1	:::1	IMAP	104	Request: 4 fetch 1 body[]
18	45.2058563	:::1	:::1	TCP	86	143 -> 39596 [ACK] Seq=913 Ack=79 Win=65536 Len=0 TSval=3207059822 TSecr=3207059822
19	45.2077745	:::1	:::1	IMAP	585	Cher étudiant que vous êtes bon, si bon que vous pouvez aller chercher un café pour le prof
20	45.2078423	:::1	:::1	TCP	86	39596 -> 143 [ACK] Seq=79 Ack=1412 Win=65152 Len=0 TSval=3207059824 TSecr=3207059824
21	50.7686213	:::1	:::1	IMAP	96	Request: 5 logout
22	50.7686799	:::1	:::1	TCP	86	143 -> 39596 [ACK] Seq=1412 Ack=89 Win=65536 Len=0 TSval=3207065385 TSecr=3207065385
23	50.7694697	:::1	:::1	IMAP	150	Response: 5 OK Logout completed (0.001 + 0.000 secs).
24	50.7695208	:::1	:::1	TCP	86	39596 -> 143 [ACK] Seq=89 Ack=1476 Win=65536 Len=0 TSval=3207065386 TSecr=3207065386
25	50.7696930	:::1	:::1	TCP	86	143 -> 39596 [FIN, ACK] Seq=1476 Ack=89 Win=65536 Len=0 TSval=3207065386 TSecr=3207065386
26	50.7699111	:::1	:::1	TCP	86	39596 -> 143 [FIN, ACK] Seq=89 Ack=1477 Win=65536 Len=0 TSval=3207065387 TSecr=3207065387
27	50.7699870	:::1	:::1	TCP	86	143 -> 39596 [ACK] Seq=1477 Ack=90 Win=65536 Len=0 TSval=3207065387 TSecr=3207065387

3. **Expliquez** ce que sont tous ces ACK, SYN en plus, en **relisant...** encore une dernière fois ☺ le cours **TCP vs UDP** sur ecampus.

4 Conclusion

Un protocole c'est quoi?

Vous voyez entre POP et IMAP ce ne sont pas les mêmes commandes !!!

Le but du TP n'était pas de faire du mail !!! mais de **voir ce qu'est un protocole et TCP**

