

# UNIVERSITÉ NATIONALE DU VIETNAM À HANOÏ

## INSTITUT FRANCOPHONE INTERNATIONAL

---



## CONCEPTION ET ARCHITECTURE DES RÉSEAUX

### RAPPORT DE TRAVAUX PRATIQUES

**Administration des réseaux et des services sous Linux**

Août 2019

Étudiants :

Cleg Peter OVILLE

Odelet VALCIN

Mike Arley MORIN

Afi Elolo Gisèle DEKPE

Enseignant : M. Nguyen Hong Quang

Année Académique : 2019 - 2020

# Table des matières

<b>Introduction</b>	<b>5</b>
<b>1 Première Partie : Réseau local autonome sans connexion vers des autres réseaux</b>	<b>6</b>
1.1 Montage du réseau d'équipe . . . . .	6
1.2 Configuration du réseau . . . . .	6
1.2.1 Configuration de l'interface interne du serveur . . . . .	6
1.2.2 Configuration d'un poste client . . . . .	7
1.2.3 Test de la connectivité par adresse IP des machines de votre réseau . . . . .	8
1.3 Installation et configuration du service de nom de serveur avec l'application ' <i>bind9</i> ' . . . . .	8
1.3.1 Installation du ' <i>bind9</i> ' . . . . .	9
1.3.2 Définition du serveur de courrier . . . . .	9
1.3.3 Configuration des zones de notre domaine . . . . .	9
1.3.4 Test du serveur de nom avec <i>dig</i> , <i>nslookup</i> , <i>host</i> . . . . .	12
1.3.5 Installation du serveur <b>ssh</b> et test de connexion à distance . . . . .	13
1.4 Installation et configuration du service de courrier sur le serveur . . . . .	14
1.4.1 Installation de <b>postfix</b> . . . . .	14
1.4.2 Configuration du service de courrier . . . . .	14
1.4.3 Test d'échange de courrier à l'intérieur de notre domaine : . . . . .	16
1.5 Installation et configuration du serveur web . . . . .	17
1.5.1 Installation d'Apache . . . . .	17
1.5.2 Affichage de la liste du groupe sur la page d'accueil . . . . .	18
1.6 Installation et configuration du serveur snmp . . . . .	19
1.6.1 Installation de <b>snmp</b> . . . . .	19
1.6.2 Installation et configuration de l'outil mrtg . . . . .	19
<b>2 Deuxième partie : Réseaux locaux interconnectés</b>	<b>21</b>

2.1	Installation et/ou configuration de l'interface externe du serveur . . . . .	21
2.2	Configuration du routage sur le serveur vers les réseaux des autres groupes . . . . .	21
2.3	Test de connectivité avec les adresses . . . . .	23
<b>Conclusion</b>		<b>24</b>
<b>ANNEXES</b>		<b>25</b>
<b>Références</b>		<b>28</b>

## Table des figures

1	Configuration de l'interface du serveur . . . . .	7
2	Configuration d'un poste client . . . . .	8
3	Test de connectivité . . . . .	8
4	Configuration du fichier '/etc/resolv.conf' . . . . .	9
5	Configuration du fichier '/etc/bind/named.conf.local.conf' . . . . .	10
6	Zones créées . . . . .	10
7	Création du fichier de la zone directe . . . . .	11
8	Création du fichier de la zone inverse . . . . .	12
9	Résultat de la commande dig . . . . .	13
10	Résultat de la commande nslookup . . . . .	13
11	Résultat de la commande host . . . . .	13
12	Résultat du test de la connexion avec ssh . . . . .	14
13	Configuration postfix 1/2 . . . . .	15
14	Configuration postfix 2/2 . . . . .	15
15	Test d'envoi de mail de OVIL vers IGORE 1/2 . . . . .	16
16	Test d'envoi de mail de OVIL vers IGORE 2/2 . . . . .	17
17	Création et configuration de l'hôte virtuel . . . . .	18
18	Liste des membres du groupe affichée depuis un navigateur . . . . .	19
19	Configuration du mrtg . . . . .	20
20	Affichage du traffic . . . . .	20
21	Modification du fichier /etc/sysctl.conf . . . . .	21
22	Définition des routes . . . . .	22
23	Configuration de la traduction . . . . .	22
24	Résultat obtenu après les configurations de routage . . . . .	23
25	Résultat des tests après correction des problèmes . . . . .	23

26	Modification du fichier main.cf 1/2 . . . . .	25
27	Modification du fichier main.cf 2/2 . . . . .	26
28	Modification du fichier master.conf . . . . .	26
29	Modification du fichier auth.conf . . . . .	26
30	Modification du fichier ssl.conf . . . . .	27
31	Modification du fichier mail.conf 1/2 . . . . .	27
32	Modification du fichier mail.conf 2/2 . . . . .	27

## **Introduction**

Ce projet d'administration des réseaux s'inscrit dans le cadre du cours de conception et architecture des réseaux. Le rapport présent décrit nos réalisations concernant le TP soumis à notre recherche ; ceci afin d'acquérir les connaissances sur les tâches d'un administrateur système et réseaux sous Linux. Ce travail comporte de ce fait deux (02) parties : celle dédiée au réseau local autonome sans connexion vers des autres réseaux et la seconde aux Réseaux locaux inter-connectés.

# 1 Première Partie : Réseau local autonome sans connexion vers des autres réseaux

**Phase de préparation :** En premier lieu, nous avons procédé à la préparation de notre serveur en y installant la distribution Debian 9 de Linux et sur nos différentes machines la distribution Ubuntu (16 et 18) de Linux ; ces derniers vont servir de postes clients. Nous pouvons maintenant passer aux configurations pour monter notre réseau au sein duquel les postes clients pourront communiquer entre eux. La tableau suivant présente le plan d'adressage de nos interfaces sur le serveur.

- **enp2s0 :**
  - Adresse : 172.16.6.2
  - Gateway : 172.16.6.1
  - Network : 172.16.6.0
  - Netmask : 255.255.255.0
- **enp3s0 :**
  - Adresse : 192.168.19.246
  - Gateway : 192.168.19.1
  - Network : 192.168.19.0
  - Netmask : 255.255.255.0

## 1.1 Montage du réseau d'équipe

Nous passons à l'affectation des adresses pour la mise en réseau du serveur et des postes clients :

- **Serveur** : 172.16.6.2
- **Poste 1** : 172.16.6.3
- **Poste 2** : 172.16.6.4
- **Poste 3** : 172.16.6.5
- **Poste 4** : 172.16.6.6
- **Poste 5** : 172.16.6.7
- **Poste 6** : 172.16.6.8

L'adresse de réseau est : **255.255.255.0**.

## 1.2 Configuration du réseau

### 1.2.1 Configuration de l'interface interne du serveur

La configuration du serveur a nécessité la modification du fichier **/etc/network/interfaces**.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
#allow-hotplug enp2s0
#iface enp2s0 inet dhcp

auto enp2s0
allow-hotplug enp2s0
iface enp2s0 inet static
address 172.16.6.2
netmask 255.255.255.0
network 172.16.6.0
gateway 172.16.6.1

#auto enp3s0
#allow-hotplug enp3s0
#iface enp3s0 inet dhcp

auto enp3s0
allow-hotplug enp3s0
iface enp3s0 inet static
address 192.168.19.246
netmask 255.255.255.0
network 192.168.19.0
gateway 192.168.19.1
```

[ Read 41 lines ]

FIGURE 1 – Configuration de l’interface du serveur

### 1.2.2 Configuration d’un poste client

Pour la configuration du poste client, nous avons fait ceci :

```

GNU nano 2.9.3                               /etc/network/interfaces

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

auto en0
iface en0 inet static
address 172.16.6.3
netmask 255.255.255.0
network 172.16.6.0
gateway 172.16.6.1

```

FIGURE 2 – Configuration d'un poste client

### 1.2.3 Test de la connectivité par adresse IP des machines de votre réseau

Ce test a été réalisé avec la commande **ping** sur l'adresses du serveur : **172.16.6.2**. Le résultat donne ce qui suit :

```

root@igore:/home/igore# ping 172.16.6.2
PING 172.16.6.2 (172.16.6.2) 56(84) bytes of data.
64 bytes from 172.16.6.2: icmp_seq=1 ttl=64 time=0.818 ms
64 bytes from 172.16.6.2: icmp_seq=2 ttl=64 time=0.279 ms
64 bytes from 172.16.6.2: icmp_seq=3 ttl=64 time=0.262 ms
64 bytes from 172.16.6.2: icmp_seq=4 ttl=64 time=0.233 ms
64 bytes from 172.16.6.2: icmp_seq=5 ttl=64 time=0.453 ms
64 bytes from 172.16.6.2: icmp_seq=6 ttl=64 time=0.320 ms
64 bytes from 172.16.6.2: icmp_seq=7 ttl=64 time=0.287 ms
64 bytes from 172.16.6.2: icmp_seq=8 ttl=64 time=0.273 ms
^C
--- 172.16.6.2 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7144ms
rtt min/avg/max/mdev = 0.233/0.365/0.818/0.183 ms

```

FIGURE 3 – Test de connectivité

## 1.3 Installation et configuration du service de nom de serveur avec l'application '*bind9*'

Dans cette partie, nous avons mis en place un serveur DNS permettant de faire le lien entre le nom de domaine et l'adresse IP du serveur et des machines connectées au serveur. Pour cela, nous avons commencé avec :

### 1.3.1 Installation du 'bind9'

Pour l'installation, la commande indiquée : **aptitude install bind9 bind9-doc**.

Après l'installation, nous avons :

- vérifié les fichiers installés,
- vérifié et configuré les fichiers des zones directe et inverse,
- démarré et testé le service serveur

Les lignes suivantes en font état.

### 1.3.2 Définition du serveur de courrier

Nous passons à la configuration du fichier "resolv.conf" en faisant : **nano /etc/resolv.conf**. A l'intérieur, on déclare le nom du domaine **groupe6.tpII.ifi** et on ajoute l'adresse IP du serveur DNS **172.16.6.2**.

```
File Edit View Search Terminal Help
GNU nano 2.7.4                               File: /etc/resolv.conf
search groupe6.tpII.ifi
nameserver 172.16.6.2
```

FIGURE 4 – Configuration du fichier '/etc/resolv.conf'

**Interprétation** : La première ligne indique quel domaine il faut ajouter aux noms si celui-ci n'est pas indiqué lors d'une demande de résolution de nom. La deuxième ligne indique le serveur DNS principal. Et c'est donc le serveur DNS qui sera chargé de donner le résultat s'il connaît la réponse.

### 1.3.3 Configuration des zones de notre domaine

- (i) Avant la configuration des zone, il faut d'abord penser à créer les zones de résolutions dans le fichier **/etc/bind/named.conf.local**.

The screenshot shows a terminal window titled "user1@serveur: ~". The title bar also displays "File Edit View Search Terminal Help" and "GNU nano 2.7.4". The main area of the terminal shows the contents of the file "/etc/bind/named.conf.local". The configuration includes comments about local configuration and RFC1918 zones, and defines two zones: "groupe6.tpII.ifi" and "6.16.172.in-addr.arpa". The nano editor interface at the bottom shows various keyboard shortcuts for navigation and editing.

```
//  
// Do any local configuration here  
  
// Consider adding the 1918 zones here, if they are not used in your  
// organization  
//include "/etc/bind/zones.rfc1918";  
  
zone "groupe6.tpII.ifi" {  
    type master;  
    file "/etc/bind/db.groupe6.tpII.ifi";  
    forwarders{};  
};  
  
zone "6.16.172.in-addr.arpa" {  
    type master;  
    file "/etc/bind/db.groupe6.tpII.ifi.inv";  
    forwarders{};  
};
```

FIGURE 5 – Configuration du fichier '/etc/bind/named.conf.local.conf'

The screenshot shows a terminal window with the command "ls /etc/bind" run by root. The output lists several files and directories: bind.keys, db.127, db.empty, db.groupe6.tpII.ifi.inv, named.conf, named.conf.local, rndc.key, db.0, db.255, db.groupe6.tpII.ifi, db.local, named.conf.default-zones, named.conf.options, and zones.rfc1918.

```
root@serveur:~# ls /etc/bind  
bind.keys  db.127  db.empty      db.groupe6.tpII.ifi.inv  named.conf      named.conf.local  rndc.key  
db.0        db.255  db.groupe6.tpII.ifi  db.local          named.conf.default-zones  named.conf.options  zones.rfc1918  
root@serveur:~#
```

FIGURE 6 – Zones créées

(ii) Ensuite, il faudra créer et remplir le fichier de la zone directe en ajoutant les postes du réseau.

The screenshot shows a terminal window titled "user1@serveur: ~". The title bar also displays "File Edit View Search Terminal Help" and "GNU nano 2.7.4". The main area of the terminal shows the content of the file "/etc/bind/db.groupe6.tpII.ifi". The file contains BIND configuration records for a local loopback interface and several hosts (serveur, poste1, poste2, poste3, poste4, mail). It includes SOA, A, and NS records with various TTL values. At the bottom of the screen, there is a menu bar with options like "Read 20 lines", "Get Help", "Write Out", "Where Is", "Cut Text", "Justify", "Cur Pos", "Exit", "Read File", "Replace", "Uncut Text", "To Spell", and "Go To Line".

```

;
; BIND data file for local loopback interface
;
$TTL    604800
@      IN      SOA    serveur.groupe6.tpII.ifi. root.groupe6.tpII.ifi. (
                      2           ; Serial
                      604800      ; Refresh
                      86400       ; Retry
                     2419200     ; Expire
                      604800 )     ; Negative Cache TTL
;
@      IN      NS    serveur.groupe6.tpII.ifi.
serveur IN      A      172.16.6.2
poste1  IN      A      172.16.6.3
poste2  IN      A      172.16.6.4
poste3  IN      A      172.16.6.5
poste4  IN      A      172.16.6.6
groupe6.tpII.ifi.   IN      A      172.16.6.2
mail    IN      A      172.16.6.2

```

FIGURE 7 – Création du fichier de la zone directe

#### Interprétation :

- *\$TTL* permet de définir en secondes et dans un intervalle qui va de 0 à 2147483647 le délai maximum pendant lequel un enregistrement pourra être gardé en cache. Avec 86400, le cache sera vidé, et les fichiers relus, toutes les 24 heures.
- *Refresh*, *Retry*, et *Expire* sont des délais, exprimés en secondes, qui vont piloter le comportement des serveurs esclaves. A l'expiration du délai *Refresh*, l'esclave va entrer en contact avec le maître ; s'il ne le trouve pas, il essaiera de nouveau à la fin du délai *Retry*. Et si, au bout du délai *expire*, il n'est pas parvenu à ses fins, il considérera que le serveur maître a été retiré du service.

- (iii) L'étape suivante consiste à la configuration de la zone inverse. Il s'agira de créer le fichier `/etc/bind/fd.groupe6.tpII.ifi` pour nous permettre de retrouver les noms des machines à partir de leur adresse IP.

```

user1@serveur: ~
File Edit View Search Terminal Help
GNU nano 2.7.4      File: /etc/bind/db.groupe6.tpII.ifi.inv

; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA    serveur.groupe6.tpII.ifi. root.groupe6.tpII.ifi. (
                      1           ; Serial
                      604800      ; Refresh
                      86400       ; Retry
                     2419200     ; Expire
                     604800 )    ; Negative Cache TTL
;
@       IN      NS     serveur.groupe6.tpII.ifi.
2       IN      PTR    serveur.groupe6.tpII.ifi.
3       IN      PTR    poste1.groupe6.tpII.ifi.
4       IN      PTR    poste2.groupe6.tpII.ifi.
5       IN      PTR    poste3.groupe6.tpII.ifi.
6       IN      PTR    poste4.groupe6.tpII.ifi.

[ Read 17 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text ^T To Spell ^_ Go To Line

```

FIGURE 8 – Création du fichier de la zone inverse

Nous avons ensuite redémarré le service avec **/etc/init.d/bind9 restart** pour que les configurations puissent être prises en compte.

#### 1.3.4 Test du serveur de nom avec *dig*, *nslookup*, *host*

Pour vérifier le bon fonctionnement des configurations, nous allons faire des tests pour trouver l'adresse IP à partir du nom et vice versa.

- (i) Pour retrouver l'adresse IP à partir du nom du serveur avec la commande suivante : **dig serveur.groupe6.tpII.ifi**

```

root@igore:/home/igore# dig serveur.groupe6.tpII.ifi
; <>> DiG 9.11.3-1ubuntu1.8-Ubuntu <>> serveur.groupe6.tpII.ifi
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 59134
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;serveur.groupe6.tpII.ifi.      IN      A

;; ANSWER SECTION:
serveur.groupe6.tpII.ifi. 604800 IN A 172.16.6.2

;; AUTHORITY SECTION:
groupe6.tpII.ifi. 604800 IN NS serveur.groupe6.tpII.ifi.

;; Query time: 0 msec
;; SERVER: 172.16.6.2#53(172.16.6.2)
;; WHEN: Sun Aug 11 12:36:34 +07 2019
;; MSG SIZE rcvd: 83

```

FIGURE 9 – Résultat de la commande dig

- (ii) Pour retrouver le nom du serveur à partir de l'adresse IP avec la commande suivante : **nslookup**  
**172.16.6.2 , host 172.16.6.2**

```

root@igore:/home/igore# nslookup 172.16.6.2
2.6.16.172.in-addr.arpa name = serveur.groupe6.tpII.ifi.

```

FIGURE 10 – Résultat de la commande nslookup

```

root@igore:/home/igore# host 172.16.6.2
2.6.16.172.in-addr.arpa domain name pointer serveur.groupe6.tpII.ifi.

```

FIGURE 11 – Résultat de la commande host

### 1.3.5 Installation du serveur ssh et test de connexion à distance

Dans cette partie il va nous falloir se connecter à distance au serveur ayant le nom serveur.groupe6.tpII.ifi ou (172.16.3.1), l'utilisation principale d'un serveur se fait à distance en accédant aux comptes qui y sont créés dans le serveur. Pour ce faire, il faut installer le serveur ssh en utilisant la commande « apt-get install openssh-server » et d'installer du côté client en utilisant la commande « apt-get install openssh-client ». Une fois l'installation terminée, nous pouvons accéder aux comptes qui sont sur le serveur en exécutant la commande : « ssh login@serveur.groupe6.tpII.ifi ».

Pour accéder au serveur à distance, nous avons besoin du serveur **ssh**. Au niveau des postes clients, nous avons installé également. Après l'avoir installé, nous pouvons accéder aux comptes qui y sont créés.

- (i) Installation :

Serveur : **aptitude install openssh-server**

Client : **aptitude install openssh-client**

## (ii) Test : ssh login@serveur.groupe6.tpII.ifi

Nous avons créé un compte **user1** avec un mot de passe avec lequel nous allons pouvoir avoir accès au serveur à distance.

```
morin@morin-MacBookPro:~$ ssh -l user1 172.16.6.2
user1@172.16.6.2's password:
Linux serveur 4.9.0-6-amd64 #1 SMP Debian 4.9.82-1+deb9u3 (2018-03-02) 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: Mon Aug 19 14:23:50 2019 from 172.16.6.3
user1@serveur:~$
```

FIGURE 12 – Résultat du test de la connexion avec ssh

## 1.4 Installation et configuration du service de courrier sur le serveur

Pour cette partie, nous avons d'abord :

- désinstallé **exim4** : **apt-get remove exim4**
- installé **postfix** : **apt-get install postfix postfix-doc**
- généré une paire certificat/clé auto-signée pour l'authentification **TLS/SSL**

### 1.4.1 Installation de postfix

Dans cette rubrique, nous aurons besoin pas seulement de Postfix, mais aussi de **dovecot** et de l'authentification **TLS/SSL**.

Avant l'installation, nous vérifions le fichier de configuration des interfaces "*/etc/network/interfaces*" ainsi que les autres adresses et le mode de connexion à *auto*. Les configurations du serveur DNS que nous avons déjà effectuées seront également importantes pour cette partie.

**Installation de dovecot** : Dovecot fournit des fonctionnalités imap avancées. Pour l'installer, on fait : **apt-get install dovecot-common dovecot-imapd**.

### 1.4.2 Configuration du service de courrier

Vu que nous utilisons TLS/SSL, nous avons besoin de certificat et de clé privée. La génération des clés s'est faite avec la commande **openssl**, à qui on demande la génération de clé **rsa**. Après la génération des clés et des certificats, nous pouvons passer à la configuration de postfix proprement dite. Il faudra donc éditer le fichier **/ect/postfix/main.cf**.

```

user1@serveur: ~
File Edit View Search Terminal Help
GNU nano 2.7.4                               File: /etc/postfix/main.cf

# Debian specific: Specifying a file name will cause the first
# line of that file to be used as the name. The Debian default
# is /etc/mailname.
#myorigin = /etc/mailname

smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no

# appending .domain is the MUA's job.
append_dot_mydomain = no

# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h

readme_directory = no

# See http://www.postfix.org/COMPATIBILITY_README.html -- default to 2 on
# fresh installs.
compatibility_level = 2

# TLS parameters
smtpd_tls_cert_file = /etc/ssl/certs/groupe6.crt
smtpd_tls_key_file = /etc/ssl/private/groupe6.key
smtpd_use_tls = yes
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc package for
# information on enabling SSL in the smtp client.

smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_destination

^G Get Help      ^O Write Out     ^W Where Is     ^K Cut Text      ^J Justify      ^C Cur Pos      ^Y Prev Page    M-\ First Line
^X Exit         ^R Read File     ^\ Replace      ^U Uncut Text   ^T To Spell     ^_ Go To Line   ^V Next Page    M-/ Last Line

```

FIGURE 13 – Configuration postfix 1/2

```

myhostname = serveur.groupe6.tpII.ifi
mydomain = groupe6.tpII.ifi
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = $mydomain
mydestination = $myhostname, groupe6.tpII.ifi, serveur.groupe6, localhost.groupe6, localhost
relayhost =
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
inet_protocols = all
home_mailbox = Maildir/
mailbox_command =
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
smtpd_tls_auth_only = no

^G Get Help      ^O Write Out     ^W Where Is     ^K Cut Text      ^J Justify      ^C Cur Pos      ^Y Prev Page    M-\ First Line
^X Exit         ^R Read File     ^\ Replace      ^U Uncut Text   ^T To Spell     ^_ Go To Line   ^V Next Page    M-/ Last Line

```

FIGURE 14 – Configuration postfix 2/2

Dans ce fichier, il faudra spécifier :

- mon domaine : ***mydomain = groupe6.tpII.ifi***
- mon origine : ***myorigin = \$mydomain***
- le répertoire des mail : ***home\_mailbox = maildir/***
- ***mailbox\_command =***, on le laisse à vide
- ***smtp\_sasl\_type = dovecot***, pour le type de serveur à utiliser
- ***smtp\_sasl\_path = private/auth***, pour le chemin vers les fichiers d'authentification

- *smtp\_sasl\_auth\_enable = yes*
- *smtp\_tls\_auth\_only = no*
- *smtp\_use\_tls = yes*, pour dire que *tls* est également utilisé pour l'authentification
- *smtp\_tls\_note\_starttls\_offer = yes*
- *smtp\_tls\_CAfile = /etc/ssl/certs/cacert.pem*
- *smtp\_tls\_loglevel = 1*, pour la lecture des logs
- *smtp\_tls\_received\_header = yes*
- *smtp\_tls\_session\_cache\_timeout = 3600s*

Lors de l'installation, il est créé plusieurs fichiers de configuration dont */etc/dovecot/conf.d* qui comporte toutes les configurations faites concernant le SSL.

Les autres fichiers modifiés sont mis en annexe, à l'instar de :

- */etc/postfix/main.cf*
- */etc/dovecot/conf.d/10-master.conf*
- */etc/dovecot/conf.d/10-auth.conf*
- */etc/dovecot/conf.d/10-mail.conf*
- */etc/dovecot/conf.d/10-ssl.conf*

A la suite nous redémarrons les services *postfix* et *dovecot* pour que les modifications faites soient prises en compte.

#### 1.4.3 Test d'échange de courrier à l'intérieur de notre domaine :

Pour vérifier l'exactitude des configurations, nous avons testé l'échange de mail à l'intérieur de notre réseau :

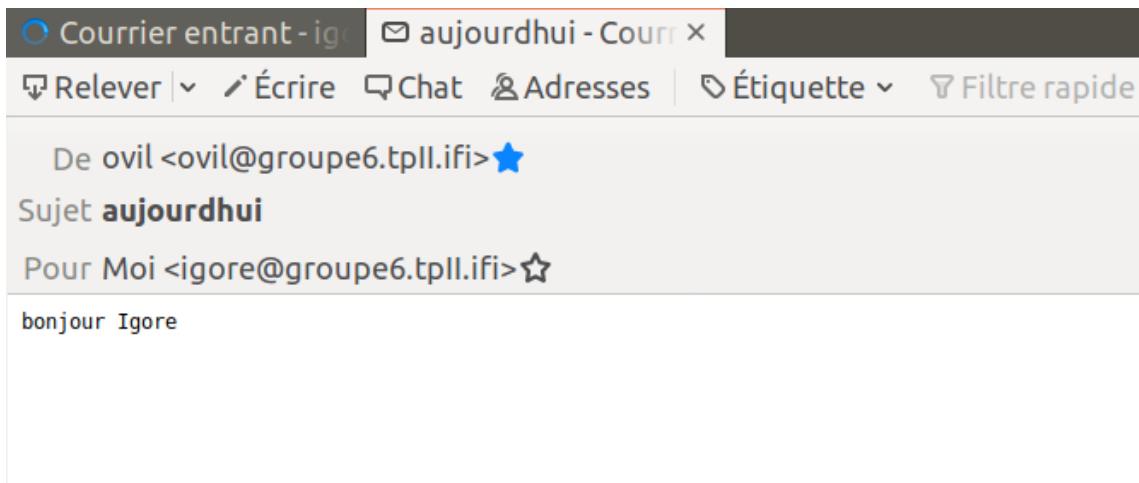


FIGURE 15 – Test d'envoi de mail de OVIL vers IGORE 1/2

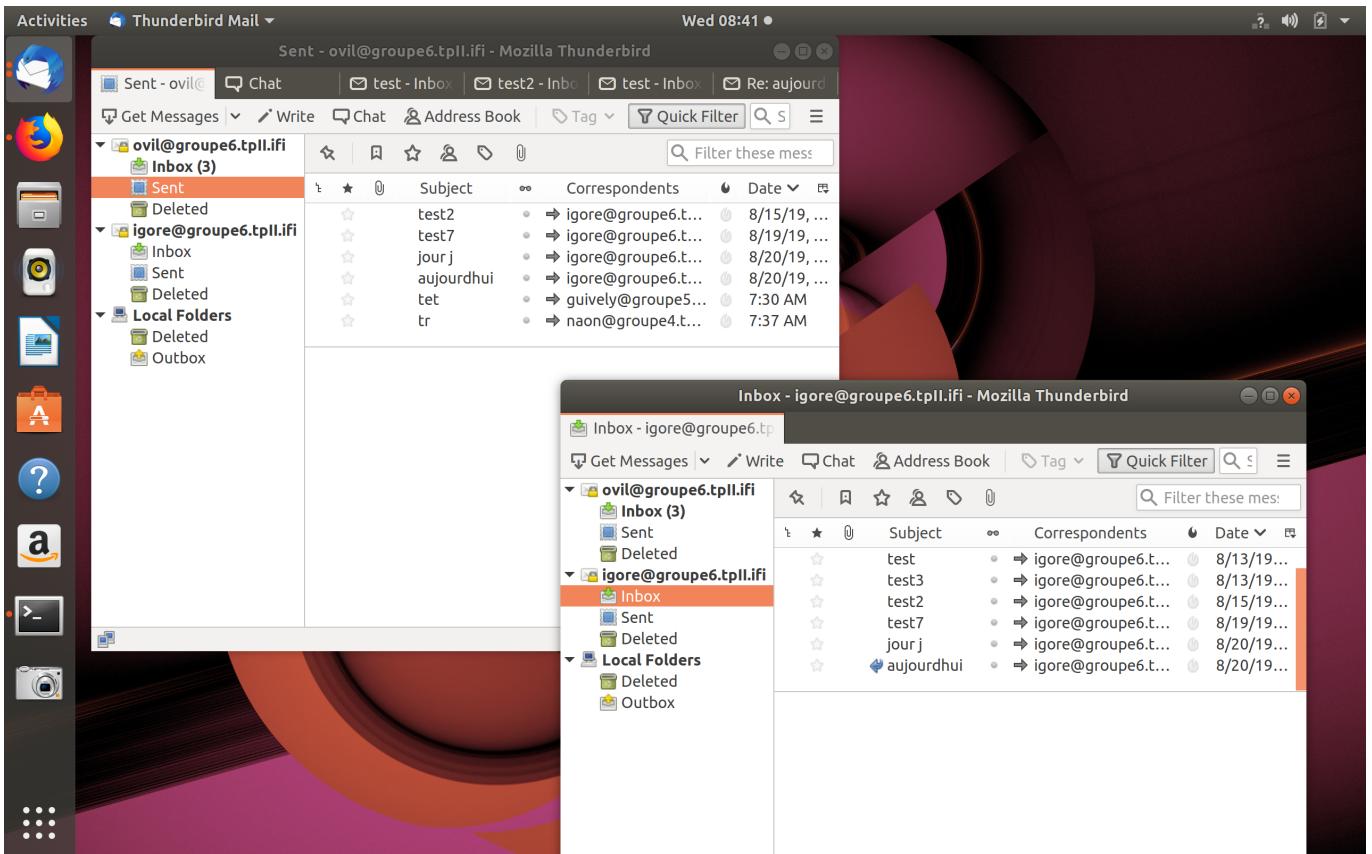


FIGURE 16 – Test d’envoi de mail de OVIL vers IGORE 2/2

## 1.5 Installation et configuration du serveur web

Ici, nous avons installé le serveur **Apache**. Nous avons ensuite modifié le fichier **/var/www/index.html**, pour qu'il affiche les noms des membres du groupe 6.

### 1.5.1 Installation d'Apache

Pour ce faire, nous utilisons la commande : **aptitude install apache2 apache2-doc**.

Nous avons ensuite configuré l'hôte virtuel, en indiquant :

- le nom du serveur
- l'adresse à contacter en cas de problème
- le dossier qui héberge notre page *index.html*

The screenshot shows a terminal window titled "user1@serveur: ~". It has two tabs: "root@morin-MacBookPro: /etc/apache2/..." and "user1@serveur: ~". The current tab displays the contents of the file "groupe6.tppII.ifi.conf" using the "GNU nano 2.7.4" editor. The configuration file contains the following code:

```
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port t$
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.

    ServerName groupe6.tppII.ifi
    ServerAdmin admin@groupe6.tppII.ifi
    DocumentRoot /var/www/groupe6

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    [ Read 31 lines ]
```

At the bottom of the terminal window, there is a menu of keyboard shortcuts:

- ^G Get Help
- ^O Write Out
- ^W Where Is
- ^K Cut Text
- ^J Justify
- ^C Cur Pos
- ^X Exit
- ^R Read File
- ^\\ Replace
- ^U Uncut Text
- ^T To Spell
- ^ Go To Line

FIGURE 17 – Crédit et configuration de l'hôte virtuel

### 1.5.2 Affichage de la liste du groupe sur la page d'accueil

Nous avons testé le fonctionnement de nos configurations en saisissant dans un navigateur l'adresse <http://www.groupe6.tppii.ifi/> depuis l'une des machines clientes.



FIGURE 18 – Liste des membres du groupe affichée depuis un navigateur

## 1.6 Installation et configuration du serveur snmp

Pour la supervision de toutes nos ressources réseaux, il nous est demandé d'utiliser l'outil **mrtg**. Ceci nécessite d'abord l'installation des serveurs « snmp ».

### 1.6.1 Installation de snmp

La commande **aptitude install snmpd snmpd** est celle utilisée.

### 1.6.2 Installation et configuration de l'outil mrtg

Nous avons ensuite installée l'outil **mtrg** pour analyser le traffic sur le réseau.

Pour l'installation, nous avons fait : **apt-get install mrtg**. Nous avons procédé à quelques configurations. Dans le fichier **/etc/snmp/snmpd.conf**, nous avons juste dé-commenté la ligne concernant le **rocommunity public localhost**. Ceci donne autorisation d'accès au localhost afin qu'on puisse visualiser les données.

```

Mon 22:31 ●
user1@serveur: ~

File Edit View Search Terminal Tabs Help
root@morin-MacBookPro: ~
user1@serveur: ~

# createUser authOnlyUser MD5 "remember to change this password"
# createUser authPrivUser SHA "remember to change this one too" DES
# createUser internalUser MD5 "this is only ever used internally, but still change the password"

# If you also change the usernames (which might be sensible),
# then remember to update the other occurrences in this example config file to match.

#####
#
# ACCESS CONTROL
#
view systemonly included .1.3.6.1.2.1.1
view systemonly included .1.3.6.1.2.1.25.1

rocommunity public localhost
rocommunity public default -V systemonly
rocommunity6 public default -V systemonly

#rocommunity secret 10.0.0.0/16

rouser authOnlyUser
# Full read-only access for SNMPv3
# Full write access for encrypted requests
# Remember to activate the 'createUser' lines above

#rwuser authPrivUser priv

# It's no longer typically necessary to use the full 'com2sec/group/access' configuration

```

FIGURE 19 – Configuration du mrtg

Ensuite, nous avons fait :

- **sudo cfgmaker public@localhost > /etc/mrtg.cfg** pour générer la configuration et la page html.
- **sudo indexmaker /etc/mrtg.cfg > /var/www/groupe6/mrtg.html** pour copier l'ensemble des configurations dans notre répertoire groupe6.

Son utilisation nous a permis d'avoir le résultat suivant :

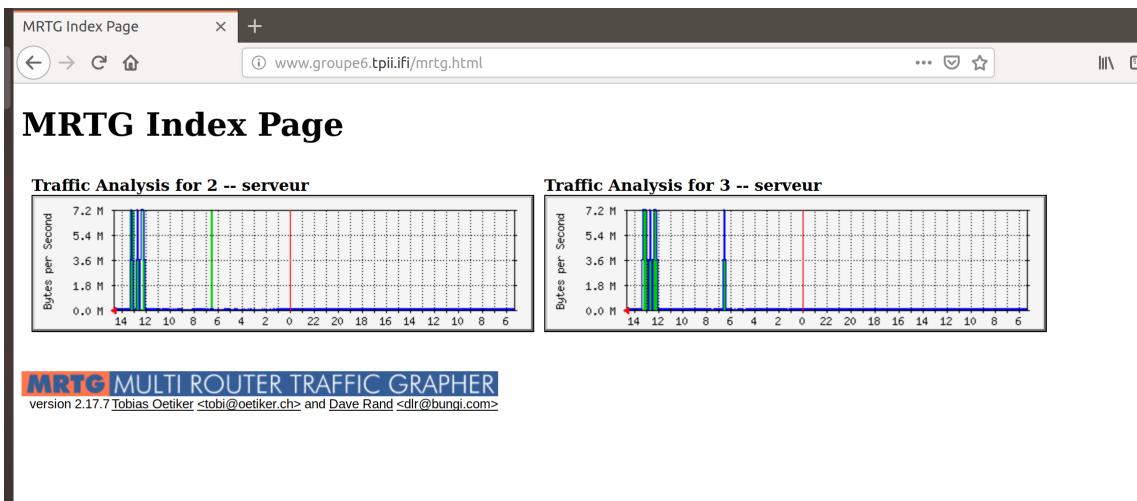


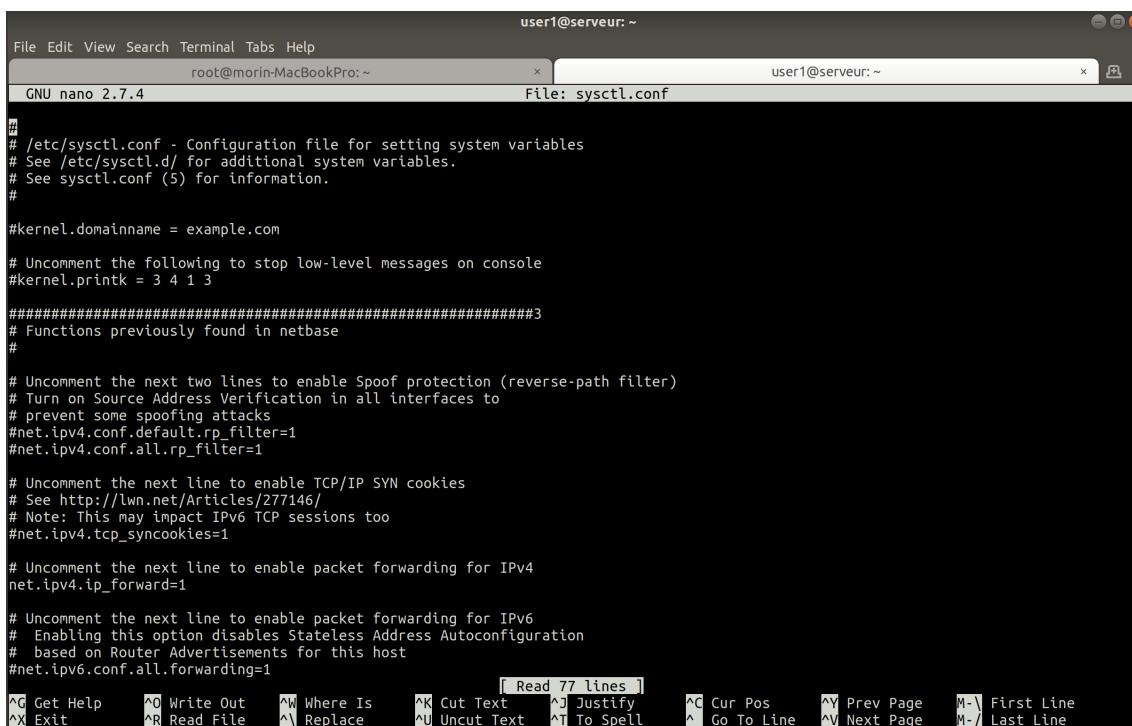
FIGURE 20 – Affichage du traffic

## 2 Deuxième partie : Réseaux locaux interconnectés

### 2.1 Installation et/ou configuration de l'interface externe du serveur

Depuis les postes clients, il n'est pas possible de pinguer par adresse IP des postes qui se trouvent sur le réseau d'un autre groupe parce que le serveur n'est pas converti en routeur. Ceci est dû au fait que le fichier **/proc/sys/net/ipv4/ip\_forward** est par défaut à **0** et que les postes clients ne savent pas quelle route emprunter pour atteindre les autres réseaux.

Dans le fichier **/etc/sysctl.conf**, nous avons décommenté la ligne **net.ipv4.conf.default.forwarding=1** pour rendre cette option permanente.



```
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#
#kernel.domainname = example.com
# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3
#####
# Functions previously found in netbase
#
# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1
#
# Uncomment the next line to enable TCP/IP SYN cookies
# See http://Lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1
#
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
#
# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1
```

FIGURE 21 – Modification du fichier `/etc/sysctl.conf`

### 2.2 Configuration du routage sur le serveur vers les réseaux des autres groupes

Nous définissons des routes pour pouvoir accéder aux autres réseaux en exécutant ce script :

The screenshot shows a terminal window titled "user1@serveur: ~". It has two tabs: "root@morin-MacBookPro: ~" and "File: route.sh". The "route.sh" tab contains the following script:

```

#!/bin/bash
echo 1 > /proc/sys/net/ipv4/ip_forward

route add -net 172.16.1.0 netmask 255.255.255.0 gw 192.168.19.246
route add -net 172.16.2.0 netmask 255.255.255.0 gw 192.168.19.246
route add -net 172.16.3.0 netmask 255.255.255.0 gw 192.168.19.246
route add -net 172.16.4.0 netmask 255.255.255.0 gw 192.168.19.246
route add -net 172.16.5.0 netmask 255.255.255.0 gw 192.168.19.246

```

The terminal window includes a menu bar with File, Edit, View, Search, Terminal, Tabs, Help, and a toolbar at the bottom with various keyboard shortcuts.

FIGURE 22 – Définition des routes

Pour qu'un poste client dans notre réseau puisse joindre un autre poste d'autre réseau, on doit activer le routage sur le serveur. Pour ce faire, nous avons configuré la traduction d'adresse NAT sur un système Linux avec des règles iptables.

Nous réécrivons donc les adresses (source,destination) des paquets IP lorsqu'elles traversent le système NAT.

The screenshot shows a terminal window titled "user1@serveur: ~". It has two tabs: "root@morin-MacBookPro: ~" and "File: iptab.sh". The "iptab.sh" tab contains the following script:

```

#!/bin/bash
#/etc/rc.local

iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -i enp2s0 -j ACCEPT
iptables -A INPUT -i enp3s0 -j ACCEPT

iptables -A FORWARD -i enp2s0 -o enp3s0 -j ACCEPT
iptables -A FORWARD -i enp3s0 -o enp2s0 -j ACCEPT

iptables -t nat -A POSTROUTING -o enp3s0 -j MASQUERADE
exit 0

```

The terminal window includes a menu bar with File, Edit, View, Search, Terminal, Tabs, Help, and a toolbar at the bottom with various keyboard shortcuts.

FIGURE 23 – Configuration de la traduction

## 2.3 Test de connectivité avec les adresses

Après ces configurations, on devrait avoir un résultat positif avec les tests de connectivité avec les autres réseaux. Mais les résultats ne sont pas satisfaisants. Nous obtenons un résultat **Destination Host Unreachable** comme le montre la figure suivante.

```
root@poste2:~# route
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref  Use Iface
default        _gateway        0.0.0.0         UG    0      0        0 enxa0cec8c16bdb
link-local     0.0.0.0         255.255.0.0     U      1000   0        0 enxa0cec8c16bdb
172.16.6.0     0.0.0.0         255.255.255.0   U      0      0        0 enxa0cec8c16bdb
root@poste2:~# ping 172.16.5.1
PING 172.16.5.1 (172.16.5.1) 56(84) bytes of data.
From 172.16.6.4 icmp_seq=1 Destination Host Unreachable
From 172.16.6.4 icmp_seq=2 Destination Host Unreachable
^C
--- 172.16.5.1 ping statistics ---
4 packets transmitted, 0 received, +2 errors, 100% packet loss, time 3016ms
pipe 2
root@poste2:~#
```

FIGURE 24 – Résultat obtenu après les configurations de routage

## METHODE DE RÉSOLUTION DES PROBLÈMES

Pour pallier à ce problème, nous avons ajouté dans la table de routage du poste client, les lignes **1** et **3**; qui désignent les routes vers le groupe 5 et le routeur.

Après un *ping*, nous obtenons une connectivité avec le **groupe 5**, comme il est montré au point **3**.

Au point **4**, on atteint le routeur et les autres serveurs du réseau, mais on ne parvient pas à joindre Internet.

```
root@poste2:~# route
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref  Use Iface
default        _gateway        0.0.0.0         UG    0      0        0 enxa0cec8c16bdb
link-local     0.0.0.0         255.255.0.0     U      1000   0        0 enxa0cec8c16bdb
172.16.5.0     serveur.groupe6 255.255.255.0 1   UG    0      0        0 enxa0cec8c16bdb
172.16.6.0     0.0.0.0         255.255.255.0 2   UG    0      0        0 enxa0cec8c16bdb
192.168.19.0   serveur.groupe6 255.255.255.0 2   UG    0      0        0 enxa0cec8c16bdb
root@poste2:~# ping 172.16.5.1
PING 172.16.5.1 (172.16.5.1) 56(84) bytes of data. 3
64 bytes from 172.16.5.1: icmp_seq=1 ttl=63 time=0.534 ms
64 bytes from 172.16.5.1: icmp_seq=2 ttl=63 time=0.545 ms
64 bytes from 172.16.5.1: icmp_seq=3 ttl=63 time=0.527 ms
64 bytes from 172.16.5.1: icmp_seq=4 ttl=63 time=0.523 ms
^X64 bytes from 172.16.5.1: icmp_seq=5 ttl=63 time=0.540 ms
^C
--- 172.16.5.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4073ms
rtt min/avg/max/mdev = 0.523/0.533/0.545/0.030 ms
root@poste2:~# ping 192.168.19.1 4
PING 192.168.19.1 (192.168.19.1) 56(84) bytes of data.
64 bytes from 192.168.19.1: icmp_seq=1 ttl=63 time=0.910 ms
64 bytes from 192.168.19.1: icmp_seq=2 ttl=63 time=0.672 ms
64 bytes from 192.168.19.1: icmp_seq=3 ttl=63 time=7.37 ms
^C
--- 192.168.19.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2008ms
rtt min/avg/max/mdev = 0.672/2.985/7.375/3.105 ms
root@poste2:~#
```

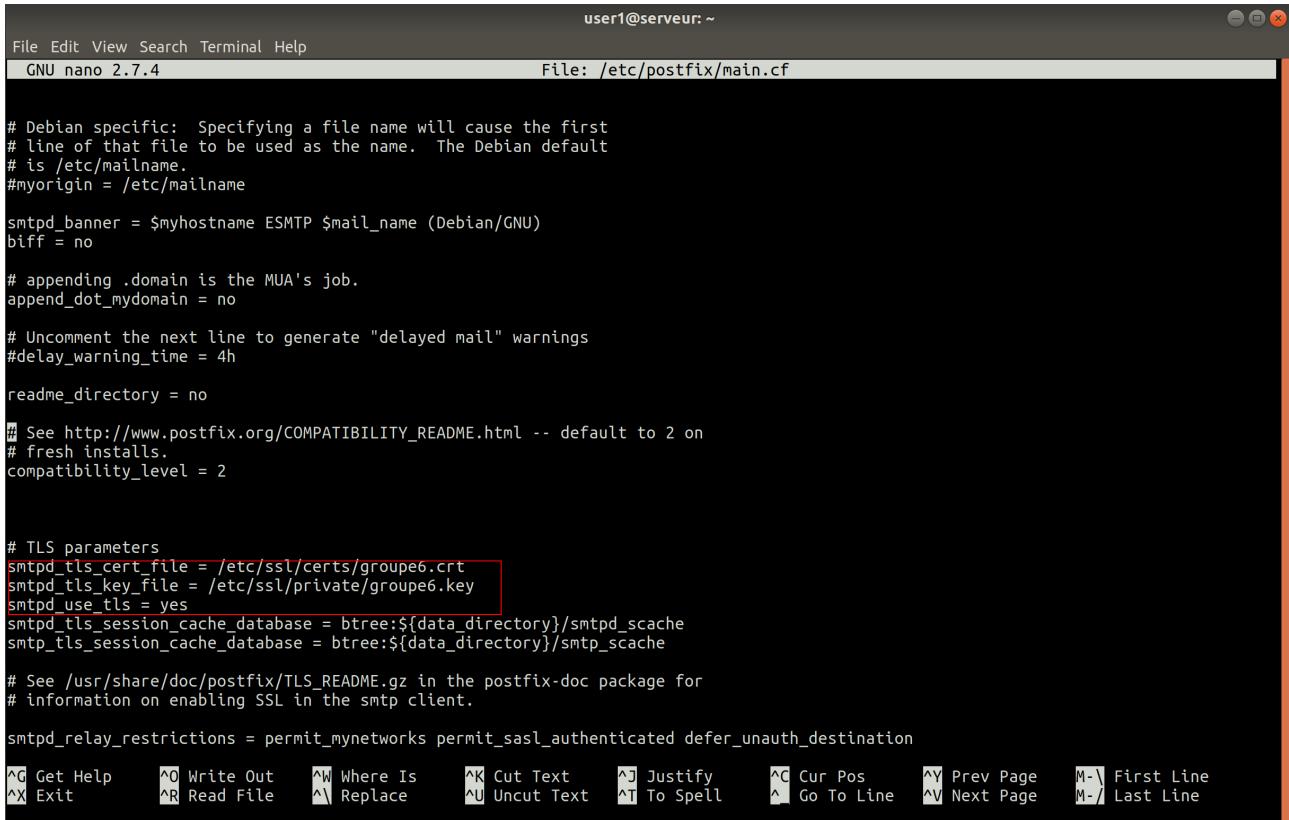
FIGURE 25 – Résultat des tests après correction des problèmes

## Conclusion

Dans ce rapport nous avons décrit comment nous avons installé et configuré les services et serveurs réseaux sou Linux ; à travers l'installation d'un serveur **Linux Debian 9**, la construction de notre réseau local et autonome. Nous avons également réalisé l'interconnexion aux réseaux d'autres groupes. Ce qui vraisemblablement fonctionne. Ce TP nous aura permis de prendre en main la mise en place d'un réseau avec les services de bases nécessaires. Nous avons ainsi par le biais de ce TP, pu déployer un serveur de nom de domaine, un serveur de messagerie, un serveur web, un serveur « snmp ». Nous pouvons également souligner qu'au cours de ce TP, nous avons rencontré quelques difficultés mais en fin de compte on a trouvé des solutions par des recherches ; ce qui nous a le plus forgé.

## ANNEXES

Nous présentons ici, certains fichiers modifiés avec les parties modifiées encadrées de couleur rouge.



```
user1@serveur: ~
File Edit View Search Terminal Help
GNU nano 2.7.4
File: /etc/postfix/main.cf

# Debian specific: Specifying a file name will cause the first
# line of that file to be used as the name. The Debian default
# is /etc/mailname.
#myorigin = /etc/mailname

smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no

# appending .domain is the MUA's job.
append_dot_mydomain = no

# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h

readme_directory = no

# See http://www.postfix.org/COMPATIBILITY_README.html -- default to 2 on
# fresh installs.
compatibility_level = 2

# TLS parameters
smtpd_tls_cert_file = /etc/ssl/certs/groupe6.crt
smtpd_tls_key_file = /etc/ssl/private/groupe6.key
smtpd_use_tls = yes
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc package for
# information on enabling SSL in the smtp client.

smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_destination

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify     ^C Cur Pos     ^Y Prev Page   M-\ First Line
^X Exit         ^R Read File    ^A Replace      ^U Uncut Text   ^T To Spell    ^L Go To Line   ^V Next Page   M-/ Last Line
```

FIGURE 26 – Modification du fichier main.cf 1/2

```

user1@serveur: ~
File Edit View Search Terminal Help
GNU nano 2.7.4
File: /etc/postfix/main.cf

# Debian specific: Specifying a file name will cause the first
# line of that file to be used as the name. The Debian default
# is /etc/mailname.
#myorigin = /etc/mailname

smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no

# appending .domain is the MUA's job.
append_dot_mydomain = no

# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h

readme_directory = no

# See http://www.postfix.org/COMPATIBILITY_README.html -- default to 2 on
# fresh installs.
compatibility_level = 2

# TLS parameters
smtpd_tls_cert_file = /etc/ssl/certs/groupe6.crt
smtpd_tls_key_file = /etc/ssl/private/groupe6.key
smtpd_use_tls = yes
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc package for
# information on enabling SSL in the smtp client.

smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_destination

^G Get Help      ^O Write Out     ^W Where Is      ^K Cut Text      ^J Justify      ^C Cur Pos      ^Y Prev Page    M-\ First Line
^X Exit         ^R Read File     ^A Replace       ^U Uncut Text    ^T To Spell     ^G Go To Line   ^V Next Page    M-/ Last Line

```

FIGURE 27 – Modification du fichier main.cf 2/2

```

#user - 
#group =
}

# Postfix smtp-auth
unix_listener /var/spool/postfix/private/auth {
  mode = 0666
  user = postfix
  group = postfix
}

```

FIGURE 28 – Modification du fichier master.conf

```

## 
## Authentication processes
## 

# Disable LOGIN command and all other plaintext authentications unless
# SSL/TLS is used (LOGINDISABLED capability). Note that if the remote IP
# matches the local IP (ie. you're connecting from the same computer), the
# connection is considered secure and plaintext authentication is allowed.
# See also ssl=required setting.
disable_plaintext_auth = yes

# Authentication cache size (e.g. 10M). 0 means it's disabled. Note that
# bsdauth, PAM and vpopmail require cache_key to be set for caching to be used.
#auth_cache_size = 0
# Time to live for cached data. After TTL expires the cached record is no
# longer used, *except* if the main database lookup returns internal failure.

```

FIGURE 29 – Modification du fichier auth.conf

```

##  

## SSL settings  

##  

# SSL/TLS support: yes, no, required. <doc/wiki/SSL.txt>  

ssl = required  

# PEM encoded X.509 SSL/TLS certificate and private key. They're opened before  

# dropping root privileges, so keep the key file unreadable by anyone but

```

FIGURE 30 – Modification du fichier ssl.conf

```

# Prefix required to access this namespace. This needs to be different for
# all namespaces. For example "Public/".
#prefix =  

# Physical location of the mailbox. This is in same format as
# mail_location, which is also the default for it.
#location =  

# There can be only one INBOX, and this setting defines which namespace
# has it.
inbox = yes  

# If namespace is hidden, it's not advertised to clients via NAMESPACE
# extension. You'll most likely also want to set list=no. This is mostly
# useful when converting from another server with different namespaces which
# you want to deprecate but still keep working. For example you can create
# hidden namespaces with prefixes "~/mail/", "~%u/mail/" and "mail/".
#hidden = no

```

FIGURE 31 – Modification du fichier mail.conf 1/2

```

# Group to enable temporarily for privileged operations. Currently this is
# used only with INBOX when either its initial creation or dotlocking fails.
# Typically this is set to "mail" to give access to /var/mail.
mail_privileged_group = mail  

# Grant access to these supplementary groups for mail processes. Typically
# these are used to set up access to shared mailboxes. Note that it may be
# dangerous to set these if users can create symlinks (e.g. if "mail" group is
# set here, ln -s /var/mail ~/mail/var could allow a user to delete others'
# mailboxes, or ln -s /secret/shared/box ~/mail/mybox would allow reading it).
#mail_access_groups =

```

FIGURE 32 – Modification du fichier mail.conf 2/2

## Références

- Cours de Conception et Architecture des Réseaux, M. Nguyen Hong Quang
- Craig Hunt TCP/IP Network Administration, 3rd Edition , O'Reilly, 2002
- Paul Albitz, Cricket Liu DNS and BIND, O'Reilly, 2001.
- Un guide sur Debian sur le site andesi.org : <http://www.andesi.org/installation>
- Un tutoriel sur les serveurs sous Linux : <http://www.linux-france.org/prj/edu/archinet/systeme/index.html>
- <http://jeyg.info/un-serveur-mail-debian-avec-postfix-et-dovecot/>
- [https://www.howtoforge.com/nat\\_iptablesstepbystep-configuration-of-nat-with-iptables](https://www.howtoforge.com/nat_iptablesstepbystep-configuration-of-nat-with-iptables)
- <https://www.google.com/amp/s/www.howtoforge.com/tutorial/how-to-install-and-configure-mrtg-on-ubuntu-1804/amp/>