

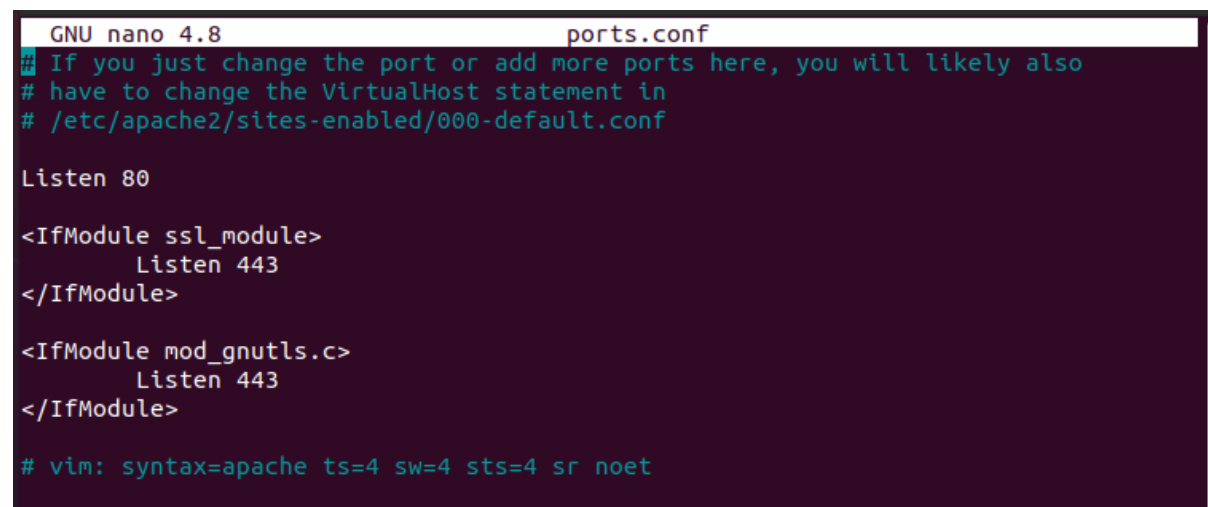
SAE Installation de services réseaux

Configurer un serveur web Apache local

Dans cette première partie nous allons voir comment configurer un serveur web Apache local sur le système d'exploitation Ubuntu.

Pour commencer tout les fichiers concernant la gestion de votre serveur web Apache se trouvent dans le répertoire `/etc/apache2`

Dans ce répertoire, ouvrons le fichier `ports.conf` pour y définir le port du site en modifiant la valeur après la directive `Listen`. En exemple nous mettrons le serveur web sur le port 80, dans notre cas nous écrirons donc '`Listen 80`'. (Si le fichier n'est pas modifiable se servir de la commande '`sudo`')



```
GNU nano 4.8                                ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Ensuite toujours dans le répertoire `/etc/apache2` allons dans le répertoire '`/etc/apache2/sites-available`' où nous créerons un nouveau fichier avec l'extension `.conf`.

Dans ce fichier y taper :

```
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
```

```
ServerName nom_domaine
ServerAlias www.nom_domaine
DocumentRoot /var/www/nom_fichier
ErrorLog ${APACHE_LOG_DIR}/error.log
</VirtualHost>
```

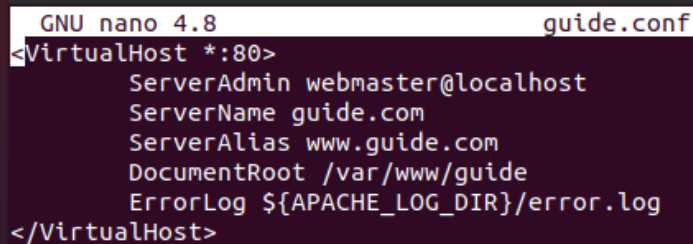
ServerAdmin : Adresse de contact à inclure dans les messages d'erreur que le serveur renvoie au client.

ServerName : Nom du serveur.

ServerAlias : Alias du serveur.

DocumentRoot : Répertoire dans la configuration à partir duquel Apache2 recherche et sert les fichiers Web de l'URL demandée à la racine du document.

ErrorLog : Répertoire dans la configuration dans lequel Apache2 enregistre les erreurs du serveur.



```
GNU nano 4.8 guide.conf
<VirtualHost *:80>
  ServerAdmin webmaster@localhost
  ServerName guide.com
  ServerAlias www.guide.com
  DocumentRoot /var/www/guide
  ErrorLog ${APACHE_LOG_DIR}/error.log
</VirtualHost>
```

Maintenons actifs le site avec l'outil a2ensite

```
sudo a2ensite guide.conf
```

Ensuite désactivons le site par défaut

```
sudo apache2ctl configtest
```

Suite à cela vérifions la présence d'éventuels erreurs

```
sudo a2dissite 000-default.conf
```

Enfin redémarrons apache2 pour appliquer les changements

```
sudo systemctl restart apache2
```

Voilà ! Votre serveur web est désormais configuré.

En tapant désormais localhost ou votre adresse IP dans la barre de recherche d'un navigateur votre site devrait être affiché

Capture de trame avec Wireshark

Cette deuxième partie à quant à elle pour but de présenter les différentes trames qui circulent lors de l'affichage d'un site web. Pour cela nous capturerons les différentes trames avec Wireshark.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	127.0.0.1	127.0.0.1	TCP	74	33428 → 80 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 ...
2	0.000017313	127.0.0.1	127.0.0.1	TCP	74	80 → 33428 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 S...
3	0.000037160	127.0.0.1	127.0.0.1	TCP	66	33428 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=2788314480...
4	0.000192763	127.0.0.1	127.0.0.1	HTTP	692	GET /FAQ/FAQ.html HTTP/1.1
5	0.000209875	127.0.0.1	127.0.0.1	TCP	66	80 → 33428 [ACK] Seq=1 Ack=627 Win=64896 Len=0 TSval=27883144...
6	0.003004670	127.0.0.1	127.0.0.1	HTTP	1483	HTTP/1.1 200 OK (text/html)
7	0.003034856	127.0.0.1	127.0.0.1	TCP	66	33428 → 80 [ACK] Seq=627 Ack=1418 Win=64256 Len=0 TSval=27883...
8	0.047607391	127.0.0.1	127.0.0.1	HTTP	565	GET /Image/logo-lyon1.png HTTP/1.1
9	0.047628571	127.0.0.1	127.0.0.1	TCP	66	80 → 33428 [ACK] Seq=1418 Ack=1126 Win=65152 Len=0 TSval=2788...
10	0.048729954	127.0.0.1	127.0.0.1	HTTP	247	HTTP/1.1 304 Not Modified
11	0.048760923	127.0.0.1	127.0.0.1	TCP	66	33428 → 80 [ACK] Seq=1126 Ack=1599 Win=65408 Len=0 TSval=2788...
12	0.065881137	127.0.0.1	127.0.0.53	DNS	101	Standard query 0xd651 A tutorat-sciences.univ-lyon1.fr OPT
13	0.065896606	127.0.0.1	127.0.0.53	DNS	101	Standard query 0xd651 AAAA tutorat-sciences.univ-lyon1.fr OPT
14	0.082743204	127.0.0.53	127.0.0.1	DNS	136	Standard query response 0xd651 A tutorat-sciences.univ-lyon1...
15	0.092691219	127.0.0.53	127.0.0.1	DNS	129	Standard query response 0xd651 AAAA tutorat-sciences.univ-lyo...
16	5.007716543	127.0.0.1	127.0.0.1	TCP	66	80 → 33428 [FIN, ACK] Seq=1599 Ack=1126 Win=65536 Len=0 TSval...
17	5.007893827	127.0.0.1	127.0.0.1	TCP	66	33428 → 80 [FIN, ACK] Seq=1126 Ack=1600 Win=65536 Len=0 TSval...
18	5.007918203	127.0.0.1	127.0.0.1	TCP	66	80 → 33428 [ACK] Seq=1600 Ack=1127 Win=65536 Len=0 TSval=2788...

Le client commence par faire un scan TCP SYN sur le port 80 du serveur (No.1), comme ce port est ouvert sur le serveur il envoie en retour un paquet TCP SYN/ACK (No.2) permettant de valider la SYN du client et d'initialiser la connexion TCP et enfin le client renvoie un paquet TCP ACK au serveur (No.3).

Le transfert de données (images, textes, etc...) s'effectue ensuite avec des requêtes HTTP, le client renvoie aussi un paquet TCP ACK au serveur après chaque transfert (No.4 à No.11).

Après avoir fini de transférer toutes les données un paquet TCP FIN/ACK est envoyé pour signifier la fin du transfert (No.16 et No.17).

Le paquet TCP SYN (No.1) permet de démarrer la connexion entre le client et le serveur, il a son flag Synchronisation à 1

```

2020 ... = Header Length: 40 bytes (10)
Flags: 0x002 (SYN)
000. .... = Reserved: Not set
...0 .... = Nonce: Not set
.... 0... = Congestion Window Reduced (CWR): Not set
.... .0.. = ECN-Echo: Not set
.... ..0. = Urgent: Not set
.... ...0 = Acknowledgment: Not set
.... .... 0... = Push: Not set
.... ..... 0.. = Reset: Not set
.... .... .1. = Syn: Set
  [Expert Info (Chat/Sequence): Connection establish request (SYN): server port 80]
.... .... 0... = Fin: Not set
[TCP Flags: .....S.]

```

Le paquet TCP SYN/ACK (No.2) permet de montrer au client que le serveur a bien reçu les paquets, il a ses flags Acknowledgment et Synchronisation à 1

```

2020 ... = Header Length: 40 bytes (10)
Flags: 0x012 (SYN, ACK)
000. .... = Reserved: Not set
...0 .... = Nonce: Not set
.... 0... = Congestion Window Reduced (CWR): Not set
.... .0.. = ECN-Echo: Not set
.... ..0. = Urgent: Not set
.... ...1 = Acknowledgment: Set
.... .... 0... = Push: Not set
.... ..... 0.. = Reset: Not set
.... .... .1. = Syn: Set
  [Expert Info (Chat/Sequence): Connection establish acknowledge (SYN+ACK): server port 80]
.... .... 0... = Fin: Not set
[TCP Flags: .....A..S.]

```

Le paquet TCP ACK (No.3) permet de montrer au serveur que le client a bien reçu les paquets, il a son flag Acknowledgment à 1

```

Flags: 0x010 (ACK)
000. .... = Reserved: Not set
...0 .... = Nonce: Not set
.... 0... = Congestion Window Reduced (CWR): Not set
.... .0.. = ECN-Echo: Not set
.... ..0. = Urgent: Not set
.... ...1 = Acknowledgment: Set
.... .... 0... = Push: Not set
.... ..... 0.. = Reset: Not set
.... .... ..0. = Syn: Not set
.... .... ...0 = Fin: Not set
[TCP Flags: .....A....]

```

Les requêtes HTTP GET (No.4 et No.8) permettent au client de recevoir des données du serveur

```

Transmission Control Protocol, Src Port: 46864, Dst Port: 80, Seq: 1, ACK: 1, Len: 562
Hypertext Transfer Protocol
  GET /FAQ/FAQ.html HTTP/1.1\r\n
    [Expert Info (Chat/Sequence): GET /FAQ/FAQ.html HTTP/1.1\r\n]
    Request Method: GET
    Request URI: /FAQ/FAQ.html
    Request Version: HTTP/1.1
    Host: localhost\r\n

```

```

Hypertext Transfer Protocol
  GET /Image/logo-lyon1.png HTTP/1.1\r\n
    [Expert Info (Chat/Sequence): GET /Image/logo-lyon1.png HTTP/1.1\r\n]
    [GET /Image/logo-lyon1.png HTTP/1.1\r\n]
    [Severity level: Chat]
    [Group: Sequence]
    Request Method: GET
    Request URI: /Image/logo-lyon1.png
    Request Version: HTTP/1.1
    Host: localhost\r\n

```

Les requêtes HTTP/1.1 (No.6 et No.10) sont des réponses du client au serveur sur l'état de la requête HTTP GET lancée précédemment. Le statut de la requête est donné par un code par exemple 200 (OK) ou 304 (Not Modified)

```

Transmission Control Protocol, Src Port: 80, Dst Port: 40004, Seq: 1, ACK: 303, Len: 141
Hypertext Transfer Protocol
  HTTP/1.1 200 OK\r\n
    [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
    Response Version: HTTP/1.1
    Status Code: 200
    [Status Code Description: OK]
    Response Phrase: OK

Transmission Control Protocol, Src Port: 80, Dst Port: 40004, Seq: 1410, ACK: 1002, Len: 101
Hypertext Transfer Protocol
  HTTP/1.1 304 Not Modified\r\n
    [Expert Info (Chat/Sequence): HTTP/1.1 304 Not Modified\r\n]
    Response Version: HTTP/1.1
    Status Code: 304
    [Status Code Description: Not Modified]
    Response Phrase: Not Modified

```

Les paquets TCP FIN/ACK (No.16 et No.17) permettent au client et au serveur d'indiquer qu'il n'y a plus de donnée à transférer et donc de terminer la connection, ils ont leurs flags Acknowledgment et Fin à 1

```

... ..
Flags: 0x011 (FIN, ACK)
  000. .... = Reserved: Not set
  ...0 .... = Nonce: Not set
  .... 0... = Congestion Window Reduced (CWR): Not set
  .... .0.. = ECN-Echo: Not set
  .... ..0. = Urgent: Not set
  .... ...1 = Acknowledgment: Set
  .... .... 0... = Push: Not set
  .... .... .0.. = Reset: Not set
  .... .... ..0. = Syn: Not set
  .... .... ...1 = Fin: Set
    [Expert Info (Chat/Sequence): Connection finish (FIN)]
    [TCP Flags: .....A...F]

```

19	50.560087447	127.0.0.1	127.0.0.1	TCP	74	35442 → 8080 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=...
20	50.560127282	127.0.0.1	127.0.0.1	TCP	74	8080 → 35442 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495...
21	50.560139004	127.0.0.1	127.0.0.1	TCP	66	35442 → 8080 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=27883650...
22	50.560964646	127.0.0.1	127.0.0.1	HTTP	620	GET / HTTP/1.1
23	50.560988902	127.0.0.1	127.0.0.1	TCP	66	8080 → 35442 [ACK] Seq=1 Ack=555 Win=65024 Len=0 TSval=278836...
24	50.562300557	127.0.0.1	127.0.0.1	HTTP	503	HTTP/1.1 200 OK (text/html)
25	50.562323420	127.0.0.1	127.0.0.1	TCP	66	35442 → 8080 [ACK] Seq=555 Ack=438 Win=65152 Len=0 TSval=2788...
26	55.571090046	127.0.0.1	127.0.0.1	TCP	66	8080 → 35442 [FIN, ACK] Seq=438 Ack=555 Win=65536 Len=0 TSval...
27	55.571275956	127.0.0.1	127.0.0.1	TCP	66	35442 → 8080 [FIN, ACK] Seq=555 Ack=439 Win=65536 Len=0 TSval...
28	55.571287037	127.0.0.1	127.0.0.1	TCP	66	8080 → 35442 [ACK] Seq=439 Ack=556 Win=65536 Len=0 TSval=2788...
29	60.481297406	127.0.0.1	127.0.0.53	DNS	100	Standard query 0x99a8 AAAA connectivity-check.ubuntu.com OPT
30	60.497253794	127.0.0.53	127.0.0.1	DNS	100	Standard query response 0x99a8 AAAA connectivity-check.ubuntu...
31	60.497361807	127.0.0.1	127.0.0.53	DNS	112	Standard query 0x4bf2 AAAA connectivity-check.ubuntu.com.local...
32	60.517950881	127.0.0.53	127.0.0.1	DNS	112	Standard query response 0x4bf2 No such name AAAA connectivity...

En changeant le port du serveur ,ici le port 8080, on remarque que si le port est ouvert il se passe la même chose que lorsqu'on était sur le port 80.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	127.0.0.1	127.0.0.1	TCP	74	35450 → 8080 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=...
2	0.000000959	127.0.0.1	127.0.0.1	TCP	54	8080 → 35450 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
3	0.000072698	:::1	:::1	TCP	94	46294 → 8080 [SYN] Seq=0 Win=65476 Len=0 MSS=65476 SACK_PERM=...
4	0.000077977	:::1	:::1	TCP	74	8080 → 46294 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
5	0.000489096	127.0.0.1	127.0.0.1	TCP	74	35452 → 8080 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=...
6	0.000493784	127.0.0.1	127.0.0.1	TCP	54	8080 → 35452 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
7	0.000542025	:::1	:::1	TCP	94	46296 → 8080 [SYN] Seq=0 Win=65476 Len=0 MSS=65476 SACK_PERM=...
8	0.000546574	:::1	:::1	TCP	74	8080 → 46296 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
9	2.043101212	127.0.0.1	127.0.0.53	DNS	90	Standard query 0x5bb1 A support.mozilla.org OPT
10	2.043581178	127.0.0.1	127.0.0.53	DNS	90	Standard query 0xd3bd AAAA support.mozilla.org OPT
11	2.058731379	127.0.0.53	127.0.0.1	DNS	160	Standard query response 0x5bb1 A support.mozilla.org CNAME pr...
12	2.069843999	127.0.0.53	127.0.0.1	DNS	128	Standard query response 0xd3bd AAAA support.mozilla.org CNAME...


Cependant si le port est fermé, la réponse du serveur au paquet TCP SYN est un paquet TCP RST/ACK.

Le paquet TCP RST/ACK est utilisé pour terminer une session TCP. Le paquet accuse la réception du paquet précédent dans le flux, puis ferme cette même session avec un paquet RST envoyé au client pour lui faire savoir que la connexion est fermée, il a ses flags Acknowledgment et Reset à 1

0101	header Length: 20 bytes (0)
▼	Flags: 0x014 (RST, ACK)	
	000. = Reserved: Not set
	...0 = Nonce: Not set
 0... = Congestion Window Reduced (CWR): Not set
0.. = ECN-Echo: Not set
0. = Urgent: Not set
1 = Acknowledgment: Set
 0... = Push: Not set
▼1.. = Reset: Set
	▶	[Expert Info (Warning/Sequence): Connection reset (RST)]
0. = Syn: Not set
0 = Fin: Not set
		[TCP Flags:A.R..]

Voilà, nous espérons que cela vous aidera dans votre configuration de serveur web !

Partie 2



Juin 2022

SAE : INSTALLATION DE SERVICES RÉSEAU

SAE S2.03, DÉCOUVERTE PARCOURS B

Camille L.
Aro R.

Introduction

Travail réalisé par le binôme Aro - Camille et fais en collaboration avec les binômes Aurelyen - Noah et Jérémie - Zoubayr

Ce document regroupe le plan d'adressage de notre réseau, la configuration sur le routeur et tout autre chose nécessaire à la configuration de notre réseau. Cette configuration se fera sur le système d'exploitation Ubuntu Linux.

Plan d'adressage du réseau :

Le réseau est composé d'un routeur, deux commutateurs et de trois ordinateurs. L'ordinateur PC 1 est sur le réseau 172.16.14.0/24 et les deux autres ordinateurs PC 2 et PC 3 sont sur le réseau 10.0.69.0/24.

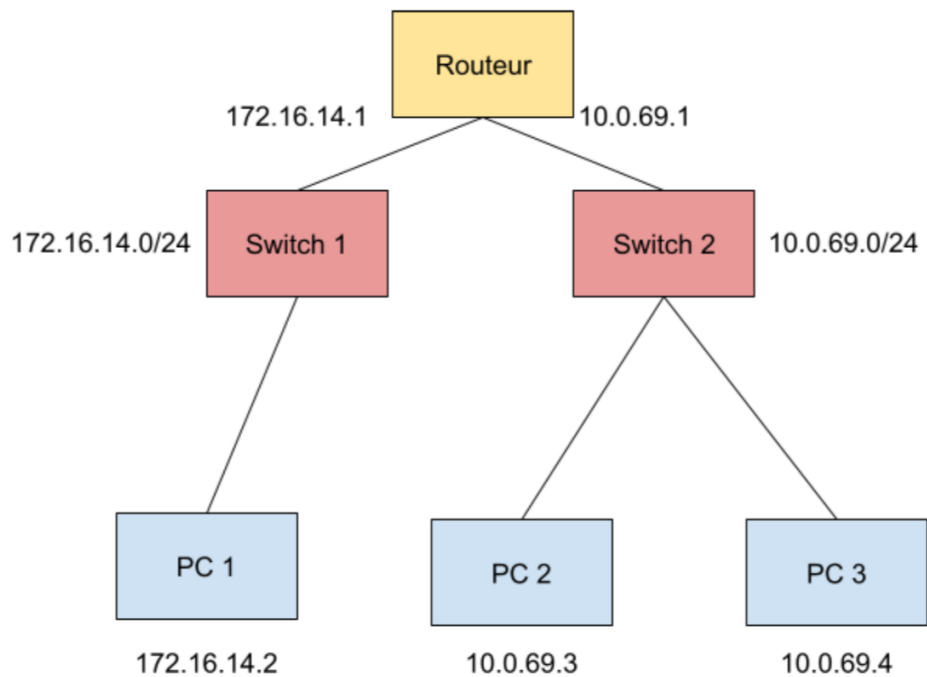


Table de routage des différentes machines du réseau :

Nom machine	réseau	masque	prochain saut	interface
routeur	172.16.14.0	255.255.255.0	DC	172.16.14.1
	0.0.0.0	0.0.0.0	DC	10.0.69.1
PC 1	172.16.14.0	255.255.255.0	DC	172.16.14.2
	0.0.0.0	0.0.0.0	172.16.14.1	172.16.14.2
PC 2	10.0.69.0	255.255.255.0	DC	10.0.69.3
	0.0.0.0	0.0.0.0	10.0.69.1	10.0.69.3
PC 3	10.0.69.0	255.255.255.0	DC	10.0.69.4
	0.0.0.0	0.0.0.0	10.0.69.1	10.0.69.4



Commandes configuration routeur:

Afin de configurer le routeur reliant les deux différents réseaux, nous nous servons du logiciel XTerm où nous entrerons les commandes suivantes :

```
enable      (Passe en mode superviseur)
configure terminal  (Passe en mode config)
ip routing   (Active le routage)
interface fastethernet 0/0  (Sélectionne l'interface connecté au
premier réseau)
ip address 10.69.0.1 255.255.255.0  (Renseigne l'@IP et le masque de
sous-réseau de l'interface)
no shutdown  (active l'interface)
exit  (sort du mode config-if)
interface fastethernet 0/1  (Sélectionne l'interface connecté au
deuxième réseau)
ip address 172.16.14.1 255.255.255.0  (Renseigne l'@IP et le masque
de sous-réseau de l'interface)
no shutdown  (active l'interface)
exit  (sort du mode config-if)
exit  (sort du mode config)
disable     (Passe en mode normal)

show ip interface brief  (debrief des interfaces connectées)
ping 10.0.69.3           (Différents ping envoyés aux
ping 10.0.69.4           ordinateurs possédant
ping 172.16.14.2        l'adresse IP correspondante)
                        (Vous pouvez repasser en mode superviseur à
                        tout moment ! On peut aussi raccourcir les
                        commandes comme ici avec "conf t"

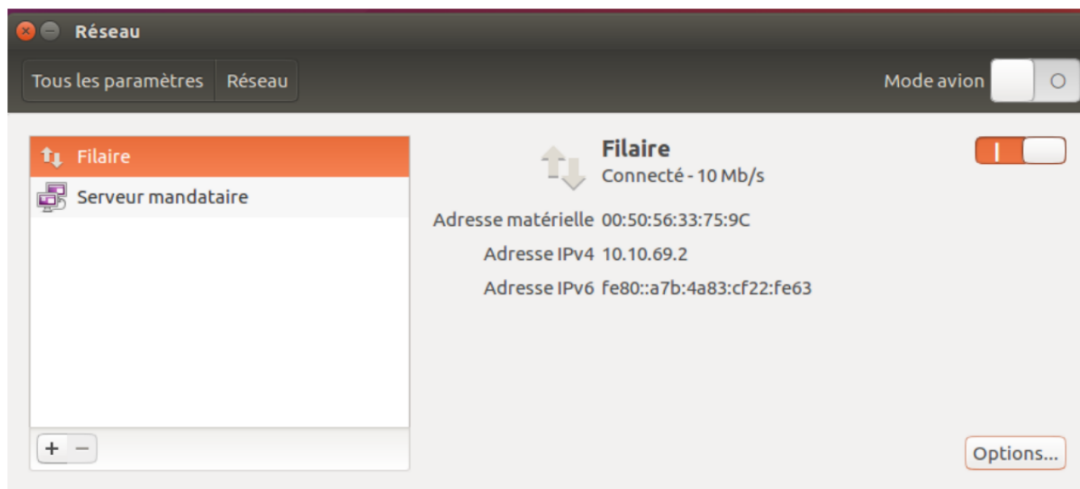
enable
configure terminal
ip route 10.0.69.0 255.255.255.0 172.16.14.0 | Configuration de la route par
exit                                          défaut (adresse réseau
disable                                     recherchée, masque de sous-
show ip route Debrief des routes par défaut | réseau et port par défaut)
```



Configuration réseau :

On va désormais configurer les adresses IPv4 de chaque ordinateur.

Allez dans les paramètres de votre machine, tapez "Réseau" dans la barre de recherche et lancez le programme de même nom :



On pourra donc appuyer sur le bouton "options" en bas à droite de la fenêtre afin d'obtenir plus d'informations concernant les paramètres réseau.



Voici la configuration des 3 ordinateurs du réseau :

Pour le PC 1 :

Nom de la connexion : Connexion SAE

Général Ethernet Sécurité 802.1x DCB Paramètres IPv4 Paramètres IPv6

Méthode : Manuel

Adresses

Adresse	Masque de réseau	Passerelle	
172.16.14.2	24	172.16.14.1	Ajouter
			Supprimer

Serveurs DNS :

Domaines de recherche :

ID de client DHCP :

☒ Requiert un adressage IPv4 pour que cette connexion fonctionne

Routes... Annuler Enregistrer

Pour le PC 2 :

Nom de la connexion : Connexion SAE

Général Ethernet Sécurité 802.1x DCB Paramètres IPv4 Paramètres IPv6

Méthode : Manuel

Adresses

Adresse	Masque de réseau	Passerelle	
10.0.69.3	24	10.0.69.1	Ajouter
			Supprimer

Serveurs DNS :

Domaines de recherche :

ID de client DHCP :

☒ Requiert un adressage IPv4 pour que cette connexion fonctionne

Routes... Annuler Enregistrer

Pour le PC 3 :

Nom de la connexion : Connexion SAE

Général Ethernet Sécurité 802.1x DCB Paramètres IPv4 Paramètres IPv6

Méthode : Manuel

Adresses

Adresse	Masque de réseau	Passerelle	
10.0.69.4	24	10.0.69.1	Ajouter
			Supprimer

Serveurs DNS :

Domaines de recherche :

ID de client DHCP :

☒ Requiert un adressage IPv4 pour que cette connexion fonctionne

Routes... Annuler Enregistrer

Configuration DNS:

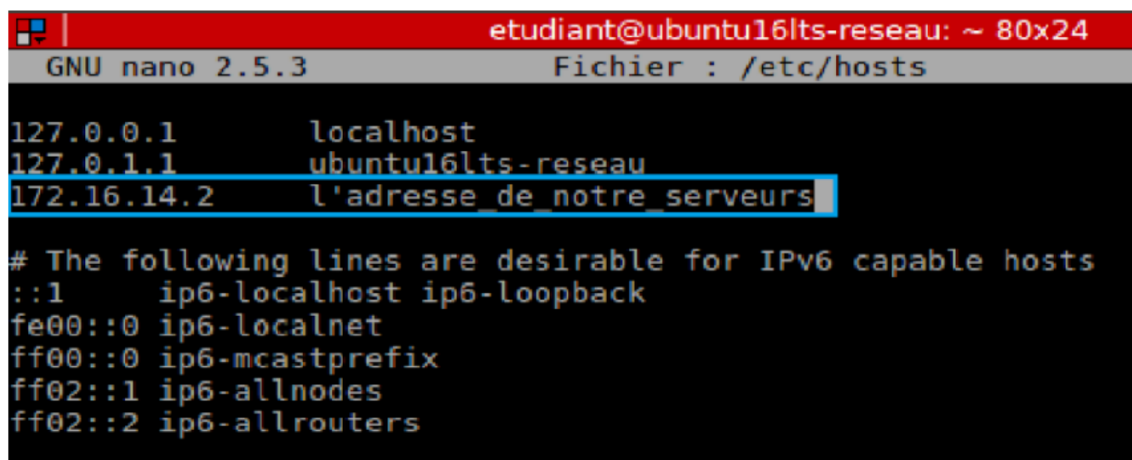
Pour lier les adresse ip des serveur à un nom de domaine il faut modifier le fichier "hosts" sur linux, celui-ci se trouve dans /etc. Il porte le même nom sur windows mais il est situé dans un autre emplacement.

Pour modifier ce fichier il faut faire cette commande. Celle-ci demande les droits administrateur, c'est pour cela que nous rajoutons sudo au début de la commande.

```
etudiant@ubuntu16lts-reseau:~$ sudo nano /etc/hosts
```

Ensuite il faut rajouter une ligne avant le commentaire dans le fichier. Sur cette ligne il faut mettre en premier votre adresse ip(V4) ("172.16.14.2" dans notre exemple) , puis le nom de domaine("l'adresse _de_ notre serveurs" dans notre exemple).

Puis on sauvegarde le fichier avec CTRL S et on quitte avec CTRL X.

A screenshot of a terminal window with a red title bar. The title bar contains a window icon, the text 'etudiant@ubuntu16lts-reseau: ~ 80x24', and a close button. Below the title bar is a grey bar with 'GNU nano 2.5.3' on the left and 'Fichier : /etc/hosts' on the right. The main area is black with white text. It shows the contents of the /etc/hosts file. The first three lines are: '127.0.0.1 localhost', '127.0.1.1 ubuntu16lts-reseau', and '172.16.14.2 l'adresse_de_notre_serveurs'. The last line is highlighted with a blue border. Below these are several lines of IPv6 addresses and their corresponding hostnames, starting with a comment: '# The following lines are desirable for IPv6 capable hosts'.

```
127.0.0.1      localhost
127.0.1.1      ubuntu16lts-reseau
172.16.14.2    l'adresse_de_notre_serveurs

# The following lines are desirable for IPv6 capable hosts
::1           ip6-localhost ip6-loopback
fe00::0       ip6-localnet
ff00::0       ip6-mcastprefix
ff02::1       ip6-allnodes
ff02::2       ip6-allrouters
```

