

TP de Réseaux n°1

Assemblage d'un réseau - Observations

1. INTRODUCTION

Si ce n'est déjà fait, il est recommandé de commencer par la lecture de la documentation qui vous a été fournie sur le matériel (Présentation de la plate-forme).

1.1 Notion de débit :

L'une des caractéristiques importantes des réseaux est le nombre de bits qui peuvent être transférés par seconde entre deux machines. C'est ce qu'on appelle le débit d'une liaison.

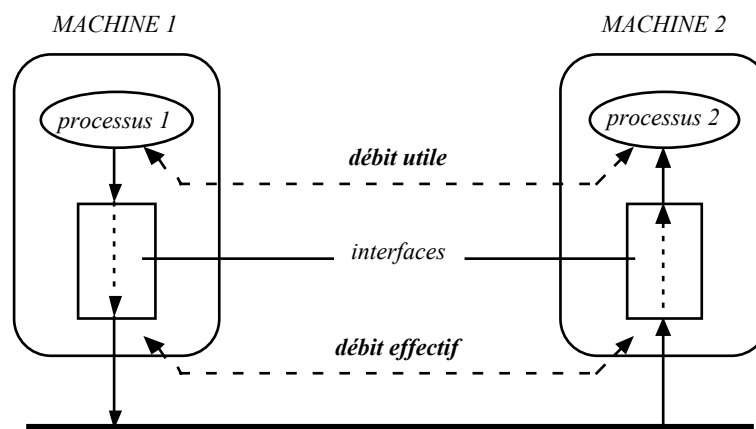


Figure 1 : Débit utile et débit effectif

Nous pouvons distinguer deux types de débits:

- **Le débit effectif.**

C'est le nombre maximum de bits qui peuvent être transférés en une seconde sur le support physique. Cette valeur est liée aux caractéristiques physiques du médium.

- **Le débit réel (ou utile).**

Les données échangées par les applications passent à travers un certain nombre d'interfaces (différentes couches OSI 2, 3, 4 ...) qui pour le besoin de la communication ajoutent des informations supplémentaires aux données proprement dites et dont l'efficacité peut varier suivant certains paramètres (charge du processeur de la machine, taille des

paquets...). Le débit réel est le nombre de bits de données échangées par seconde au niveau des applications (voir figure 1).

Le débit effectif est naturellement supérieur au débit réel puisqu'il est le maximum que celui-ci peut atteindre.

Rappel: le débit effectif du réseau Ethernet peut être 10, 100 ou 1000 Mégabits/s.

1.2. Adresse Ethernet et adresse Internet:

Chaque interface Ethernet possède une adresse dites « physique » (ou Ethernet) qui est fixée au moment de sa fabrication. Cette interface peut se situer sur la carte mère de l'ordinateur ou être ajoutée par la suite sous forme de carte. Le protocole Ethernet est réalisé par « hard » (contrairement à TCP/UDP et IP).

L'adresse Ethernet tient sur six octets. La notation habituelle pour ces adresses Ethernet consiste à écrire les six octets hexadécimal et à les séparer par ":". Par exemple: **8:0:20:4:69:d6**

Il existe d'autres types de réseaux au sens « réseau physique » et protocole de niveau 2.

Ces réseaux utilisent d'autres types d'adresse. Il est donc nécessaire d'attribuer une adresse « logique » à chaque machine qui permet de faire abstraction de la nature des réseaux sous-jacents. Dans le monde Internet cela est fait au niveau de la couche IP (Internet Protocol : niveau « réseau » dans les couches OSI). Dans notre cas c'est l'**adressage Internet** (version 4) qu'on utilisera.

Internet est un réseau de réseau (**Inter**connexion **Network**).

L'adresse IP est constituée de manière à identifier le réseau (au sens local) sur laquelle elle est connecté et à la distinguer des autres machines se trouvant aussi sur ce réseau.

Une adresse IP :

- un numéro qui identifie le réseau sur lequel se trouve la machine : on parle d'**adresse réseau**;
- un numéro qui identifie la machine dans ce réseau : on parle d'**adresse machine** sur le réseau.

Cette adresse tient sur quatre octets et est donnée sous la forme **n₁.n₂.n₃.n₄** où n_i est la valeur décimale d'un octet.

Plusieurs **classes d'adresses** Internet existent suivant la taille de l'adresse réseau (un, deux, ou trois octets). Voici en résumé, pour chaque classe, les bits réservés pour le codage de l'adresse du réseau (bits r) et ceux réservés pour le codage de l'adresse de la machine sur le réseau (bits m). La valeur des 3 premiers bits de l'octet de poids fort décide de la classe.

<i>Classe</i>	<i>Format des adresses</i>
A	0 rrrrrrr.mmmmmmmmm.mmmmmmmmm.mmmmmmmmm
B	10 rrrrrrr.rrrrrrrrr.mmmmmmmmm.mmmmmmmmm
C	110 rrrrr.rrrrrrrrr.rrrrrrrrr.mmmmmmmmm

Exemples:

- adresse classe A (en décimal): **55.22.45.12**
- adresse classe B (en décimal): **132.10.155.1**
- adresse classe C (en décimal): **195.1.10.41**

Adresse sans classe :

Pour des raisons de pénurie et donc d'économie, les adresses IP peuvent maintenant être attribuées sans tenir compte des classes. Il suffit de préciser le nombre de bit de la partie « réseau » de l'adresse.

Notation : 192.0.0.193/26

Le /26 indique que 26 bits de poids fort sont réservés pour la partie réseau. La partie machine fait ici 6 bits.

ATTENTION certaines applications ou commandes système tiennent compte encore des classes d'adresses. Par exemple ifconfig attribue par défaut le netmask associé à la classe de l'adresse.

2. DEROULEMENT DU TP:

2.1. Mise en place du réseau

2.1.1.Raccordement du matériel:

Vous utiliserez dans les Tps de la **paire torsadée** et des **Hubs**. (**Attention à ne pas utiliser ici de commutateurs à la place des hubs**).

Connectez physiquement les 4 PCs sur un même réseau.

Inspirez vous du schéma donné dans la figure 2. Vous pouvez utiliser au choix une des deux interfaces Ethernet se trouvant à l'arrière des machines.

2.1.2.Configuration des machines

Votre réseau est prêt; il faut maintenant configurer les machines au niveau « système », afin qu'elles se reconnaissent et qu'elles puissent dialoguer.

Vous pouvez rendre votre réseau opérationnel de deux façons:

1. Soit en modifiant des fichiers de configurations précis et en « rebootant » les machines pour que celles-ci prennent en compte vos modifications; Plus d'intervention n'est alors ensuite nécessaire pour la configuration réseau.
2. Soit en lançant manuellement les commandes qui permettent de configurer tout de suite vos machines.

2.1.2.1.Choix des adresses INTERNET des machines

Référez vous à l'introduction pour plus d'informations sur les adresses.

Choisissez une classe d'adresses. Dans cette classe, choisissez une adresse pour votre réseau. Enfin, choisissez une adresse pour chaque machine.

Dans le schéma de la figure 2, notez les différentes adresses et les noms des interfaces choisies (plan d'adressage).

Remarque: Si nous vous laissons choisir librement l'adresse des machines, c'est uniquement parce que celles-ci ne seront pas raccordées au réseau international. Normalement, il faut formuler une demande auprès d'un organisme international (**NIC:.....**) qui distribue de façon unique les adresses INTERNET dans le monde entier.

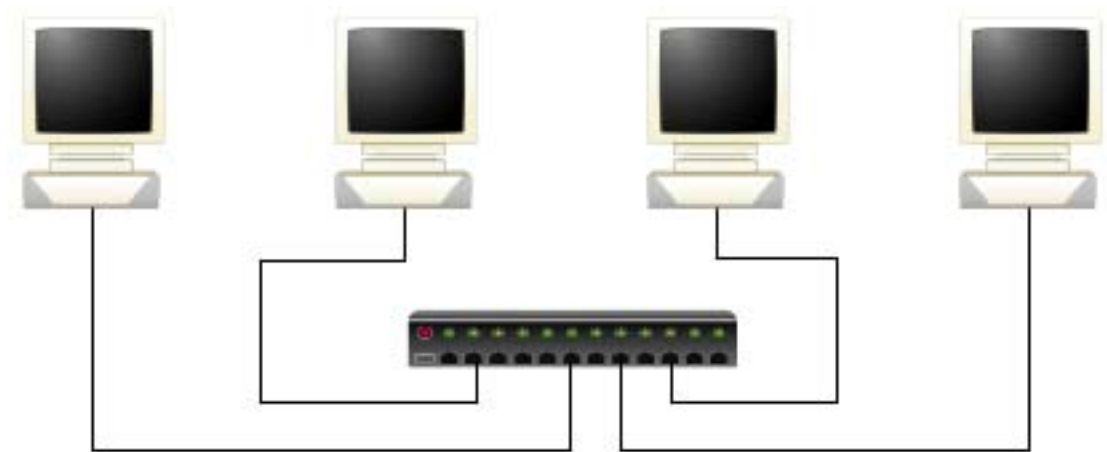


Figure 2: Réseau à réaliser

2.1.2.2. Configuration manuelle des machines

La commande utilisée pour configurer les interfaces Ethernet s'appelle **"ifconfig"** ("InterFace CONFIGuration").

Configurer une interface consiste à l'initialiser, lui associer un certain nombre d'informations (l'adresse INTERNET de la machine entre autre), et enfin la déclarer en "état de marche".

Avant de configurer l'interface d'une machine, il faut que vous connaissiez son nom « système ». Les noms d'interface sont toujours de la forme: **<nom><numéro>** (**x10, x11, ep0...**). On peut connaître le nom de l'ensemble des interfaces de la machine en tapant **ifconfig**.

lo0 est une interface virtuelle servant au rebouclage (loopback) sur la machine, son adresse est toujours 127.0.0.1. Il ne faut pas la modifier.

- Configuration de l'interface

La configuration avec ifconfig est de la forme suivante:

ifconfig <nom_interface> <adresse INTERNET>

Configurez l'interface des quatre machines.

- Marquage de l'interface à l'état "marche" (UP)

Maintenant que l'interface est initialisée et configurée, vous pouvez la marquer "prête à l'emploi" !

La commande à lancer est: **ifconfig <nom_interface> up**

On peut aussi taper en une seule fois :

ifconfig <nom_interface> <adresse INTERNET> up

Désormais, votre machine peut dialoguer sur le réseau Ethernet.

Remarque: Vous n'avez pas besoin de rebooter les machines pour que votre réseau soit opérationnel. Mais dès que vous les éteindrez, elles perdront leur configuration, et vous aurez à ré-exécuter les mêmes commandes quand vous les rallumerez.

- Contrôle de l'état des interfaces

A tous moment vous pouvez contrôler l'état de vos interfaces par la commande :

ifconfig <nom_interface>

Vous obtiendrez quelque chose comme:

```
en0:  
flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST>  
mtu 1500  
  
inet6 fe80::20a:95ff:fea2:686c%en0 prefixlen 64 scopeid 0x5  
  
inet 129.88.38.229 netmask 0xfffffe00 broadcast 129.88.39.255  
  
inet6 2001:660:5301:26:20a:95ff:fea2:686c prefixlen 64 autoconf  
  
ether 00:0a:95:a2:68:6c  
  
media: autoselect (100baseTX <full-duplex>)  
  
status: active
```

Les informations qui nous intéressent dans le résultat de cette commande sont éventuellement de voir si l'interface (de nom **en0**) est en marche (**UP**), de connaître l'adresse Internet associée à l'interface (**inet ...**) ainsi que l'adresse Ethernet de la machine (**ether ...**). Le **status** permet de savoir si la carte est branchée à un réseau.

2.1.2.3. Configuration par modification des fichiers de configuration

Dans une situation réelle, on ne configure pas les machines "à la main" (sauf exceptionnellement): on modifie des fichiers de configurations que les machines lisent automatiquement au moment du boot. La configuration se fait alors de façon automatique.

Pour lui indiquer qu'une interface doit être configurée, vous devez modifier le fichier: **/etc/rc.conf**

Par exemple, si vous désirez que l'interface **xl0** soit configurée au moment du boot, vous devez rajouter la ligne : **ifconfig_xl0="inet 192.168.1.1 netmask 255.255.255.0"**

2.1.2.4.Association adresse / nom symbolique

On peut associer un (ou plusieurs) nom symbolique aux adresses Internet. Cette association doit être défini dans le fichier **/etc/hosts** . Les lignes du fichier **/etc/hosts** sont de la forme suivante:

<adresse INTERNET> <nom officiel de la machine>

On peut associer plusieurs noms à une même adresse. Remplissez les fichiers **/etc/hosts** sur chaque machine avec les 4 adresses et noms associés.

Il n'y a pas besoin de reboot pour que les « alias » du fichier **hosts** soient vus par les différentes applications utilisant des adresses Internet. On remarquera que les associations sont locales à une machine, on peut donc choisir des alias différents sur les machines mais bien sûr ce n'est pas recommandé.

A titre d'exercice, vous pouvez éditer le fichier **/etc/hosts** sur le serveur **ufrima** et analyser son contenu.

2.1.3.Contrôle du réseau

2.1.3.1.Utilisation de ping

Il convient maintenant de vérifier que les machines sont bien connectées et bien configurées. Il existe un outil standard pour cela: **ping**. Par défaut ping permet de vérifier qu'une machine distante répond bien quand on lui envoie un paquet.

Sur une machine, lancez la commande suivante:

ping <adresse_internet_de_machine_distante>

Si la « connexion réseau » est possible avec la machine distante l'application ping affiche à l'écran une ligne donnant le temps d'aller/retour. Il recommence toutes les secondes jusqu'à que l'on tape **ctrl-C**.

Remarque : L'utilisation des adresses Internet en décimale au niveau utilisateur n'est pas souple, il est commode d'utiliser plutôt le nom donné dans le fichier **hosts**.

2.1.3.2.Procédure de login sur une machine distante

On va utiliser pour cette manipulation l'application **telnet** qui permet à un utilisateur de se logger sur une machine distante.

Cette manipulation fait apparaître la machine locale comme étant un terminal relié à la machine distante. (cf man telnet)

Tapez: **telnet** <nom_de_la_machine_distante>

Au prompt « **login** » et "**password:**" donnez le compte et le mot de passe associé à l'utilisateur existant sur les machines **login** : **guest** et **password:** **guest** (telnet sur le compte root n'est pas possible pour des raisons de sécurité).

Une fois votre login accepté, vous pouvez travailler sur la machine distante de la même façon que localement.

L'application telnet crée une connexion entre les deux machines à travers laquelle les commandes tapées sur la machine locale sont transférées pour être exécutées sur la machine distante. Les résultats obtenus sur celle-ci seront également transférés à travers cette même connexion pour être affichés sur l'écran de la machine locale.

2.2.Observation de l'activité du réseau

Rappel: Pour communiquer, les machines échangent des informations sous forme de paquets qui sont l'unité de données échangées sur le réseau.

Après avoir configuré les machines et vérifié au niveau utilisateur que le « réseau » fonctionne correctement, vous allez maintenant "écouter" le câble Éthernet et regarder ce qui se passe quand vous lancez des commandes comme ping, telnet ou talk.

L'outil qui permet d'observer le réseau s'appelle **ethereal** (voir documentation « outils »).

- Observation de la commande ping

- Sur une des quatre machines, lancez ethereal. Une capture peut être lancée en cliquant le bouton **start** du menu capture. Une nouvelle fenêtre apparaît permettant de spécifier des paramètres de la capture; il faut choisir l'interface sur laquelle on veut lancer la capture puis lancez effectivement la capture.

- Sur une deuxième machine, exécutez un ping pour savoir si une troisième machine est en marche: **ping** <nom_d'une_machine_distante>

Sachant que ping utilise des paquets de type **ICMP**, analysez et commentez le fonctionnement de ping.

En plus des paquets concernant directement ping (ICMP), d'autres paquets (de type **ARP** : Address Resolution Protocol) devraient apparaître. Si ce n'est pas le cas, laissez

reposer la machine d'où vous aviez lancé ping pendant au moins une minute, et relancez ping. Vous pouvez aussi taper la commande **arp -d -a** (vidage de la table ARP).

- Analysez et interprétez les paquets de type ARP.

Que contient la table ARP (commande **arp -a**). Pourquoi les paquets ARP n'apparaissent pas systématiquement avant chaque paquet *ICMP request* ?

2.3.Observation du protocole CSMA/CD

Le protocole de la **sous couche MAC (Multiple Acces Carrier)** de la couche 2 appelé **CSMA/CD** (Carrier Sense Multiple Access/Collision Detection) a été normalisé par l'**IEEE** sous le nom de **802.3** et permet de contrôler l'accès au support des réseaux Éthernet. Il est implémenté sur la carte "Ethernet" .

Rappels : Le protocole CSMA/CD se base sur le mécanisme suivant: avant d'émettre une trame, une machine écoute toujours le câble pour vérifier qu'aucune autre machine n'est déjà en train d'émettre des données. Lorsque le médium devient disponible, la machine envoie sa trame.

Il arrive parfois que deux machines connectées sur un même câble Éthernet décident d'envoyer simultanément (ou presque) une trame. Dans ce cas, les signaux électriques s'ajoutent, et aucune des deux trames n'est plus lisible: on dit qu'il se produit une **collision**. Chaque machine est capable de détecter ces collisions pendant qu'elle émet une trame.

Quand une collision est détectée, chaque machine arrête son émission, et attend pendant un laps de temps aléatoire, avant de tenter de re-émettre toute la trame.

Dans le cas des réseaux Ethernet, les trames échangées ont une longueur maximale de **1518 octets** et une longueur minimale de **64 octets**.

- Dans une fenêtre d'une première machine, lancez l'outil **netstat** de manière à ce qu'il affiche à intervalles d'une seconde l'activité sur l'interface utilisée (nombre de paquets émis, reçus et surtout nombre de collisions observées sur cette interface): **netstat -I xl0 1**

Dans une autre fenêtre, utilisez **udpmt** (voir la documentation "Outils utilisés dans les TPs Réseaux") (**udpmt -s <taille_paquet> -p <port-serveur> <serveur>**) pour générer un trafic à destination d'une deuxième machine où vous aurez lancé auparavant **udptarget -p <port-serveur>**.

Ces utilitaires se trouvent dans le répertoire **/ipmt-tools**

Notez le nombre de collisions.

Sur une troisième machine, lancez **udpmt** à destination de la quatrième machine de la plate-forme (où vous aurez lancé auparavant **udptarget**).

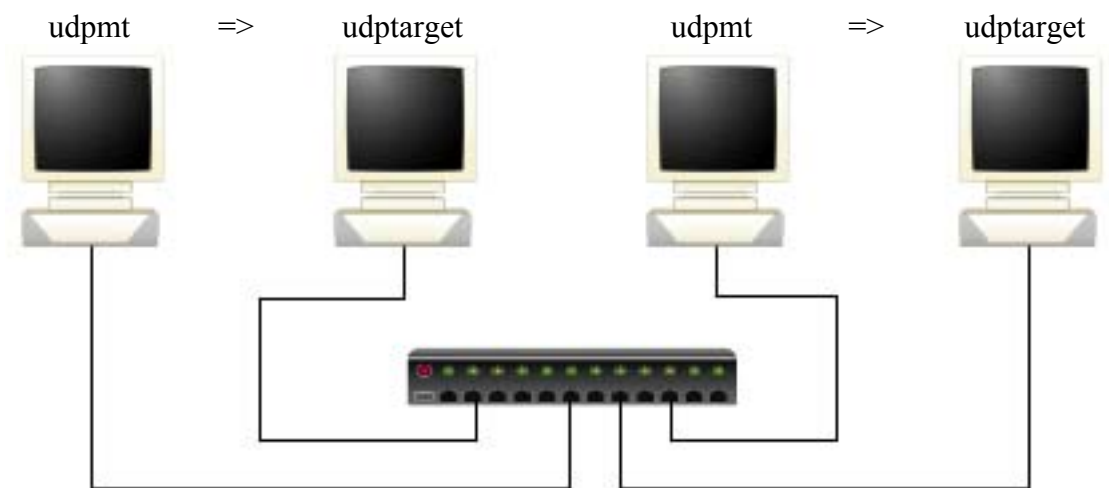


Figure 3 : Génération de trafic sur le réseau

Lancez netstat sur chaque machine.

Notez la variation du nombre de collisions.

- Essayez la manipulation précédente avec des paquets de tailles différentes (vérifiez la taille des paquets circulant sur le réseau).

Expliquez les variations du nombre de collisions en fonction de la taille des paquets (en rappelant le protocole CSMA/CD).

Rappel: une machine ne voit les collisions que lorsqu'elle est en train d'émettre.

- Observer le nombre d'erreurs détectées (affiché par netstat).

2.4. Analyse des performances du réseau

Les utilitaires (**udpmnt** et **tcpmt**) permettent de calculer le débit réel du réseau (niveau applicatif).

Pour calculer ce débit, ils mesurent donc le temps nécessaire pour envoyer des paquets (de taille donnée) sur le réseau soit en utilisant le protocole *udp* soit le protocole *tcp*.

2.4.1. Mesure du débit réel (niveau applicatif)

Sur une des quatre machines, utilisez **udpmnt** pour envoyer des paquets vers une autre machine.

- Notez le compte rendu de *udpmnt* en faisant varier la taille des paquets (de 1, 10, 20, 100, 1000, 1450, 2000, et 3000 octets).

Pour comprendre pourquoi le débit réel n'atteint pas le débit effectif d'Ethernet, lancez sur une troisième machine l'utilitaire **ethereal** et capturez quelques trames échangées pendant l'exécution de **udpmt**.

- Ne pas essayez de comprendre pour l'instant la signification des champs des différentes entêtes, regardez seulement les champs "longueur de paquet" et "longueur d'entête".
- Expliquez précisément les débits que vous constatez. Pour cela calculer le pourcentage nombre d'octets de données/nombre d'octets total des paquets.

Quelques points pour vous aider à comprendre:

- **udpmt** utilise ici le protocole UDP pour émettre ses messages et mesurer le débit utile.
- UDP rajoute des octets "de contrôle" (appelés entête) aux données qui lui sont fournies. Ces paquets UDP (entête + données) sont passés à un autre protocole de niveau réseau (IP) qui rajoute aussi des octets de contrôle. Ce protocole IP passe ensuite le tout au protocole Ethernet qui s'occupe d'envoyer physiquement les octets sur le câble, Ethernet rajoute aussi une entête.
- Dans le calcul de ces octets rajoutés par les protocoles, il ne faudra pas oublier (non visible avec Ethereal) le CRC d'Ethernet (4 octets), le préambule d'Ethernet (8 octets), le silence Inter trame de 9,6 micro secondes.

2.4.2.Mesure du débit dans le cas de plusieurs trafics.

Choisissez deux machines qui serviront à charger le réseau (émission en continu de paquets). Lancez comme précédemment **udpmt** entre ces deux machines.

Lancez à peu près en même temps **udpmt** entre les deux machines restantes comme indiqué dans la figure 3 :

- Notez les débits moyens sur les deux machines génératrices de **udpmt**, que constatez-vous?
- Comparez ces résultats à ceux obtenus en 2.4.1.
- Pour un nombre de station fixe, l'efficacité du protocole Ethernet dépend de la taille des paquets et du temps de propagation. On arrive en approximant à l'équation suivante (voir le cours) :
 - $E = 1/(1+(5,4 \cdot T_{prop}/T_{emis}))$ où T_{prop} est le temps de propagation et T_{emis} est le temps d'émission ($T_{emis} = \text{Taille_paquet} / \text{débit}$).
 - Calculez T_{prop} pour votre réseau. Donnez la courbe de l'efficacité en fonction de la taille des paquets pour le T_{prop} calculé. Conclusions.

- Pourquoi le partage est-il « à peu près équitable » ? Quel protocole gère cette équité ?
- Mesurez le temps de propagation (application à application) en utilisant l'utilitaire *ping* entre deux machines du réseau.

2.4.2. Mesures en utilisant des commutateurs

- Remplacer le hub de votre réseau par un commutateur (ou switch).
- Refaire les mesures de débit pour différent cas de flux : 2 flux indépendants (figure 3) et deux ou trois flux vers une même machine.. Faites varier la taille des paquets (une petite, une grande suffiront). On observera les débits en émission et en réception (donnés par *udptarget*)
- Conclure sur les débits observés en rappelant le fonctionnement d'un commutateur.
- Mesurez le temps de propagation (application à application) en utilisant l'utilitaire *ping*. Expliquez la différence avec celui mesuré dans le cas de l'utilisation d'un HUB.

2.5. Exercice de synthèse

Il existe plusieurs réseaux locaux dans le bâtiment (adresses Internet différentes). L'adresse 195.221.226.0/24 est normalement attribuée aux salles réseaux (101,102,104 et 105).

- Regardez sur le prompt système, le nom associé à votre machine (*knuthxx*). L'adresse correspondant au nom de machine **knuthxx doit être 193.54.236.(xx+150)**.
Exemple : adresse de knuth02 : 193.54.236.152. Le netmask associé (/25) doit être : **255.255.255.128**.

- Affectez cette adresse à la deuxième interface de la machine sur laquelle vous travaillez (ifconfig ...) et branchez cette interface à une prise réseau murale.

- Vérifiez que le routeur est accessible : **ping 193.54.236.129**
- Tapez la commande **route add default 193.54.236.129**.

- Vérifiez que l'accès à un des serveurs de l'UFR est possible : **ping 195.220.82.136** pour le serveur *hopper*. Vous pouvez maintenant envoyer des fichiers sur votre compte à l'aide de l'application *sftp* (ftp sécurisé) : **sftp moncompte@195.220.82.136**. Il existe aussi l'application *scp* (**scp nomfichier compte@adresse:nomfichier**).

- Vous pouvez aussi utiliser l'application **ssh** pour travailler à distance sur une autre machine. Pour des raisons de sécurité, les serveurs des applications ftp et telnet ne tournent pas sur les serveurs de l'UFR.

- Que se passe-t-il si plusieurs étudiants utilisent la même adresse Internet sur plusieurs machines ?