
HALAWI Ali

AWADA Ahmad

EL MOUSSAOUI HABIB

TP1: Réseaux

MISE EN PLACE DU RESEAU :

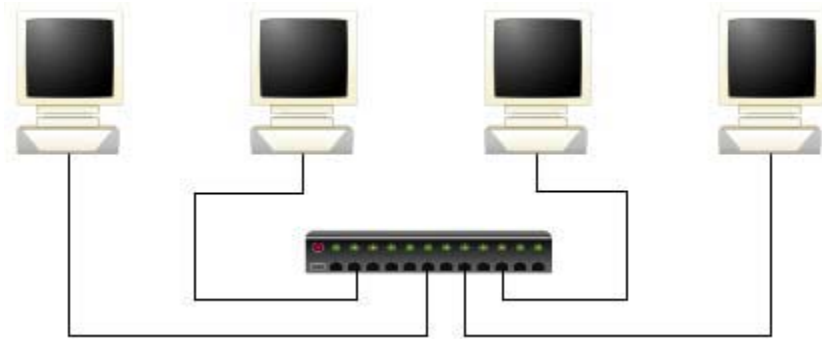


Figure 1 : Réseau réaliser

On a connecté tous les ordinateurs a un hub en utilisant des câbles directes (non croisées) puis on a configurée chaque interface internet de chaque PC en utilisant la commande « ifconfig » :

```
Ifconfig <nom_interface> <adresse INTERNET>
Ex : ifconfig lx0 192.168.0.1
```

Après avoir initialisée et configurée l'interface, on la marquer a l'état « marche » en utilisant la commande

```
Ifconfig <nom_interface> <adresse INTERNET>up
```

Cependant, il existe d'autre méthode pour configurer les machines « a la main » en modifiant le fichier /etc/rc.com.

Utilisation du Ping :

Pour vérifier que les machines sont bien connectées on a utilisées un outil standard : « Ping » qui permet de vérifier qu'une machine distante répond bien quand on lui envoie un paquet. Pour cela on a lancé la commande suivante :

ping 192.168.0.1 (adresse d'une machine distante)

Qui affiche à l'écran une ligne donnant le temps d'aller/retour qui recommence toutes les secondes jusqu'à l'on tape ctrl-C.

Sinon si on veut pas utiliser l'adresse internet décimale on utilise le nom donné dans le fichier **hosts**.

Procédure de login sur une machine distante:

Chaque machine dispose d'un login qui s'appelle guest autre que le root (administrateur) pour cela on a essayé la commande « telnet » qui permet de se logger sur une machine distante.

```
telnet 192.168.0.2  
login : guest  
password : guest
```

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.

Observation de l'activité du réseau :

Après avoir configuré les machines et vérifié au niveau utilisateur que le « réseau » fonctionne correctement, on va maintenant "écouter" le câble Ethernet et regarder ce qui se passe quand on lance des commandes comme ping, telnet.

L'outil qui permet d'observer le réseau s'appelle **ethereal** :

Observation de la commande ping :

On a lancé sur une des machines (la machine d'adresse 192.168.0.1) « ethereal » et sur une autre on a tapé ping 192.168.0.1.

On remarque que le ping utilise des paquets de type ICMP et comme on a cité avant le ping sert à :

Vérifier qu'une machine distante répond bien quand on lui envoie un paquet.

En plus des paquets concernant directement ping (ICMP), d'autres paquets (de type **ARP** : Address Resolution Protocol) n'apparaissent pas au début pour cela on a vidé le tableau grâce à la commande :

arp -d -a

après qu'on a vidé le tableau des paquets apparaissent dans « ethereal » ce qui permet de dire que chaque machine identifie maintenant l'adresse ethernet de l'autre. En plus si on affiche le contenu du tableau par : arp -a on remarque aussi qu'il reconnaît l'adresse internet de chacune.

Donc les paquets ARP n'apparaissent pas à chaque fois parce que une fois les 2 machines sont identifiées, on aura plus besoin de remplir le tableau ARP à nouveau.

Observation du protocole CSMA/CD :

Pour réaliser la manipulation on a lancé un serveur sur une des machine(192.168.0.1) grâce a la commande :

Udptarget -p 80 (une numéro de port au hasard)

Après on a connecté 2 autres machine(192.168.0.2 , 192.168.0.3) sur ce serveur grâce à la commande :

udpmt -s 200 -p 80 192.168.0.1

on a remarqué au début en utilisant netstat (netstat -l lx0 1) que la taille des paquets augmente le nombre de collisions augmente, en plus une machine ne voit les collisions que lorsqu'elle est en train d'émettre.

Au niveau du serveur, on a un nombre d'erreurs qui apparait lors de chaque collision(sequence error : il attendait un numéro de paquets et il reçoit une autre), sur les autres machines clients lors de la collision le paquet n'est pas transmis et ils attendent un temps aléatoire pour renvoyer le paquet.

Analyse des performances du réseau:

Mesure du débit réel (niveau applicatif)

En regardant la trace effectuée par « ethreal » on a constaté que pour un paquet envoyé dont le nombre d'octets est 200 on reçoit sur l'autre machine 242 octets ce qui nous permet d'expliquer la différence entre débit réel et le débit effectif.

Débit réel : 242 octets

Débit effectif : 200 octets

Cette différence de débit correspond au octets rajouté aux données par l'UDP qui sont des octets « de contrôle » appelés entête dont la valeur dans l'exemple précédent est de 42 octets qui se distribuent de la manière suivante :

- 2 octets ajoutés par : type de protocole
- 8 octets ajoutés par un protocole IP.
- 20 octets ajoutés par le protocole ETHERNET qui s'occupe d'envoyer physiquement les octets sur le câble.
- 4 octets pour le CRC d'Ethernet
- 8 octets pour le préambule d'Ethernet

Mesure du débit dans le cas de plusieurs trafics.

On rappelle qu'on utilise bien un concentrateur (hub).

On réalise la même manipulation entre de machine comme étant l'un serveur l'autre client et on lance un ping la machine cliente.

On remarque que les débits est de 7241 Kb/s , or lorsque on a connecté une autre machine sur le serveur le débit a passé à 4264 Kb/s ce qui permet de constater qu' un concentrateur partage les débit en moitié (50 % pour la machine 1 et 50% pour l'autre) ayant un temps d'émission calculé de 12 ms.

Le protocole est à peu près équitable comme les débits sont partagés à moitié, le protocole qui gère cette équité est l'ETHERNET.

Mesures en utilisant des commutateurs.

En remplaçant le hub par un commutateur et en réalisant la même démarche on constate que le Switch est plus intelligent et plus performant qu'un hub on a toujours le même débit moyen sur différentes machines.

Ceci peut être expliqué par le fait que le Switch essaye de faire des circuits à deux ports. En plus en mesurant le temps de propagation en utilisant l'utilitaire ping, on remarque que le temps est divisé par 2 comparé à celle d'un HUB.

En résumé, de nos jours les commutateurs sont devenus plus utilisés vue la performance qui offrent en temps que débit et temps de propagation d'application à application, or ce n'était pas le cas 5 ou 10 ans avant comme les SWITCH coûtaient chers.

Merci pour votre lecture !!