

Cheick CISSOKO

Gaëtan LAGIER

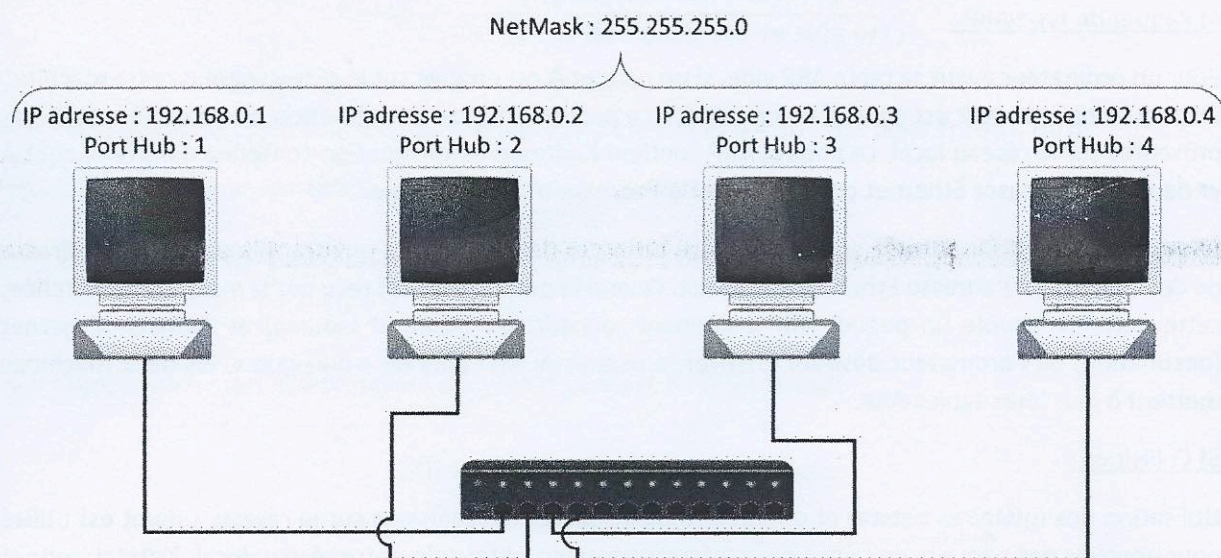
Eric THIERRY

M2 Pro CCI

Très bien

CR réseau TP1

1) Schéma du réseau utilisé au cours de la séance :



2) NetMask et adresse broadcast :

Netmask : FFFFFFF0 (255.255.255.0), c'est le masque de réseau pour des adresses IP de classe C. Ce qui est équivalent à 192.168.0.X/24. Sur un réseau comme nous avons défini, l'adresse broadcast correspond à la dernière adresse possible sur ce réseau soit 192.168.0.255

Pour simplifier l'identification des machines du réseau, nous avons modifié le fichier hosts pour associer des noms aux adresses IP des machines.

192.168.0.1 = pc1

192.168.0.2 = pc2

192.168.0.3 = pc3

192.168.0.4 = pc4

3) Ping et paquet ICMP :

Machines utilisées pour la commande ping :

Machine A :

- Adresse IP : 192.168.0.3 (nom symbolique pc3)

Machine B :

- Adresse IP : 192.168.0.2 (nom symbolique pc2)

Commande ping exécutée sur Machine A : ping pc2

Un paquet de type ICMP provenant de la machine A est envoyé sur le réseau en tant que requête. Quand la machine B (pc2) reçoit le paquet elle en renvoie un en réponse à destination de la machine A. A la réception de ce dernier paquet par la machine A, la commande ping indique si les deux machines peuvent communiquer et donne à titre indicatif le temps d'aller-retour entre les deux machines.

4) Paquet de type ARP :

Pour un ordinateur ayant sa table ARP vide, si un paquet A est envoyé sur le réseau depuis cette machine, un paquet de type ARP est envoyé en broadcast. Le paquet est alors à destination de toutes les machines présentes sur le réseau local. Le paquet ARP contient l'adresse de destination contenue dans le paquet A et demande l'adresse Ethernet correspondant à l'adresse IP de destination.

Le paquet ARP a pour adresse source l'adresse Ethernet de l'ordinateur envoyant le paquet A. L'adresse de destination est l'adresse Ethernet broadcast. Quand le paquet ARP est reçu par la machine recherchée, cette dernière envoie un paquet ARP contenant son adresse Ethernet (source) et l'adresse Ethernet (destination) de l'ordinateur désirant envoyer le paquet A. Au cours du « dialogue », les deux machines mettent à jour leurs tables ARP.

5) Collisions :

Utilisation des utilitaires netstat et udpmt afin de visualiser les collisions sur le réseau. udpmt est utilisé pour envoyer des paquets de tailles diverses entre deux machines de notre réseau local. Netstat permet d'observer le réseau, il est lancé sur la machine envoyant les paquets pour pouvoir détecter les collisions. Seule la machine qui envoie les paquets pourra détecter les collisions.

Nous observons une moyenne de 800 collisions, avec des intervalles de 10 sec pour netstat et des paquets de 1472 octets envoyé par udpmt. Il n'y a pas de détection d'erreur. Avec une taille de 64 octet par paquet, on observe une très légère baisse du nombre de collisions en comparaison des paquets de 1472 octets.

6) Collisions avec plus de trafic sur le réseau :

Il n'y a pas d'augmentation du nombre de collision, quel que soit la taille des paquets utilisés.

7) Variations du nombre de collisions en fonction de la taille des paquets :

Le protocole CSMA/CD permet de réduire le nombre de collision suivant deux principes, émission seulement quand le support est libre (CSMA) et détection des collisions (CD). La détection est possible car la machine écoute le support tout en émettant. Si une différence est observée c'est qu'il y a eu collision.

Seule la machine qui a émis le paquet est capable de détecter la collision. Ainsi pour une détection efficace des collisions, il faut que le temps d'émission soit supérieur ou égale au temps d'aller-retour sur le réseau. Ce qui implique une taille minimale des paquets à envoyer sur le réseau. Pour des paquets de trop petite taille, la machine émettrice ne détecte pas de collision de son paquet avec un autre.

*ce devrait être la
cause ?*

Non

*Ethernet rajoute
des octets de
bourrage pour
arriver aux 64 octets*

Pour les questions 5) et 6), il n'était pas évident de voir une variation du nombre de collisions en fonction de la taille des paquets. Pour faciliter cette observation, il faudrait un temps de propagation plus grand donc une distance de câble plus longue, 200 m serait une taille intéressante.

8) Temps de propagation et efficacité :

Le réseau que nous avons créé, était composé de câbles d'environ 3m entre chaque ordinateur et le hub. Ainsi la longueur du réseau était d'environ 6m. Pour calculer le temps de propagation on utilise l'équation suivante :

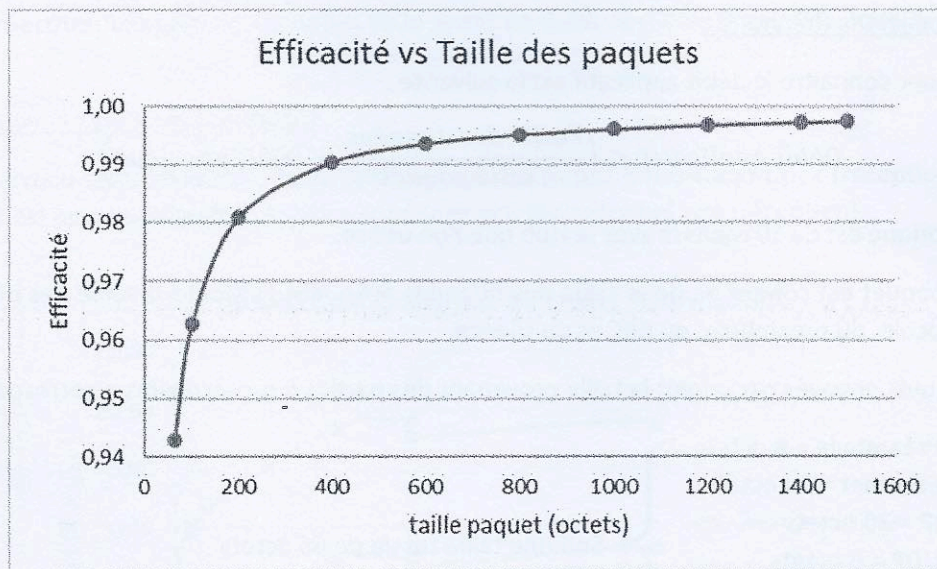
$$T_{prop} = \frac{L \text{ (Longueur du support)}}{v \text{ (vitesse de l'onde sur le support)}}$$

Les caractéristiques de notre réseau Ethernet sont : $L = 6 \text{ m}$ et $v = 10^8 \text{ m/s}$ donc $T_{prop} = 60 \text{ ns}$

L'efficacité du réseau est obtenue grâce à l'équation suivante :

$$E = \frac{1}{\left(1 + \left(5,4 \times \frac{T_{prop}}{T_{émis}}\right)\right)} \quad \left(T_{émis} = \frac{\text{Taille paquet}}{\text{débit}}\right)$$

Sur notre réseau nous avons mesuré un débit maximum de 9571 kbits/s



La taille de notre réseau étant très petite, l'efficacité évolue très peu avec la taille des paquets.

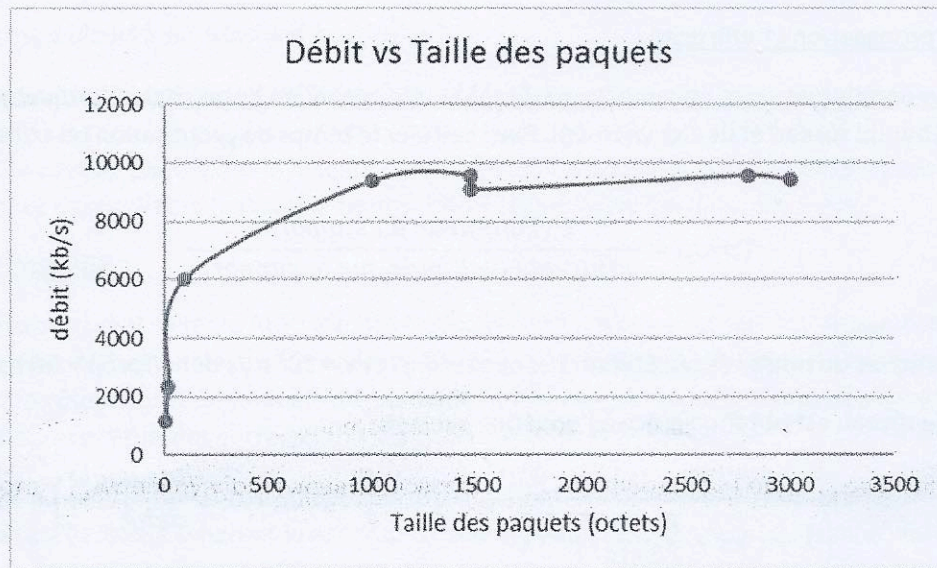
9) Partage du débit :

Le partage du débit est à peu près équitable car les machines parlent sur le réseau seulement s'il est libre, elle dispose ainsi du débit maximal à chaque envoi de paquet. C'est le protocole CSMA/CD qui gère cette équité.

et la ligne aléatoire

10) Débit vs taille des paquets :

Mesure du débit pour des paquets de tailles diverses (10, 20, 100, 1000, 1472, 1473, 2800 et 3000 octets)



11) Débits applicatifs théorique :

L'équation pour connaître le débit applicatif est la suivante :

$$\text{Débit applicatif} \leq \left(\frac{\text{Taille des données}}{\text{Taille paquet}} \times \text{Débit physique} \right)$$

Le débit théorique est de 10 Mbits/s avec le Hub que l'on utilise.

La taille du paquet est composée de la taille des données auxquelles s'ajoute la taille des différents en-tête de protocole, du préambule, du CRC et du silence.

Dans les paquets envoyés par udpmnt la taille provenant des parties « protocolaires » correspond à :

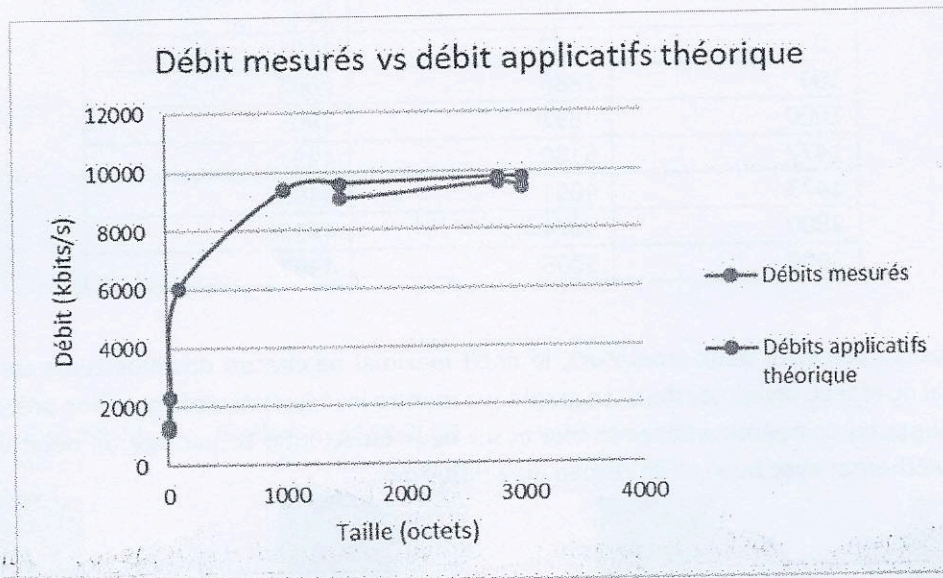
- Préambule = 8 octets
 - Ethernet = 14 octets
 - IP = 20 octets
 - UDP = 8 octets
 - CRC = 4 octets
 - Silence = 12 octets
- } Soit une taille totale de 66 octets

Taille données (octets)	Débit applicatif (kbits/s)	Taille données (octets)	Débit applicatif (kbits/s)
10	1315,8	1472	9570,9
20	2325,6	1473	9571,2
100	6024,1	2800	9769,7
1000	9380,9	3000	9784,7

64 octets min + préambule + silence = 84

$$\frac{10}{84} = 1190$$

2 paquets donc
1473
1472 + 66 + 84

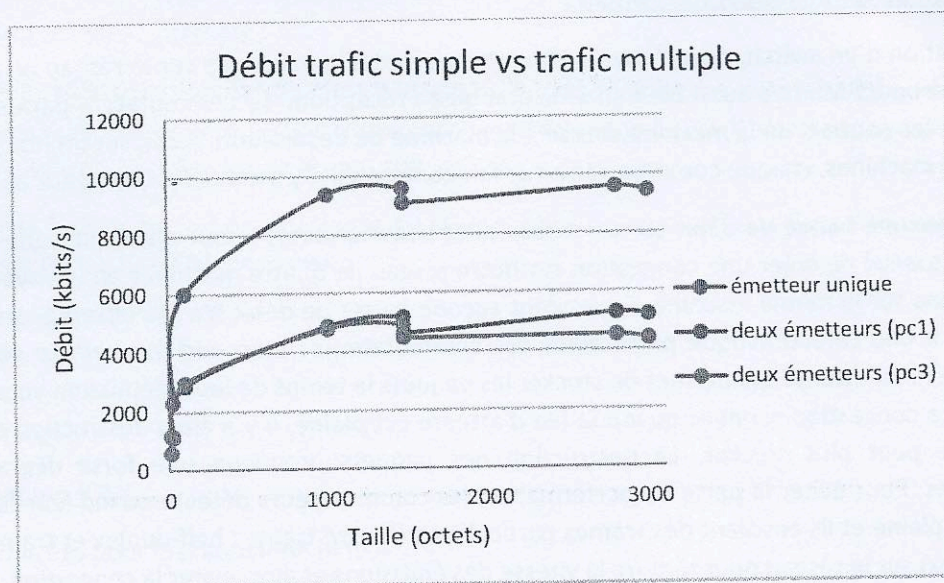


On voit que les résultats sont identiques pour les paquets de 10, 20, 100, 1000 et 1472 octets. Pour les paquets suivant (1473, 2800, 3000) les débits mesurés sont légèrement plus faibles. Ceci s'explique par le fait que pour des paquets de taille supérieure à 1472 octets, il y a fragmentation des paquets par la machine émettrice. Les paquets de petite taille ayant un débit applicatif plus faible, cela diminue le débit moyen.

? vous oubliez le
bonnage à 64 octets

12) Débit moyen avec plusieurs trafics

Pour ce nouveau test, udpmt est lancé entre les machines pc1 (émission) et pc2 (réception). Un second test udpmt est lancé en simultané sur les machines pc3 (émission) et pc4 (réception).



/ Bin

Taille (octets)	Débit pc1 (kbits/s)	Débit pc3 (kbits/s)
10	602	587
20	1133	1191
100	2886	2985
1000	4892	4805
1472	5180	4937
1473	4651	4499
2800	5274	4532
3000	5206	4397

Lorsque le réseau présente deux émetteurs, le débit maximal de chacun des émetteurs correspond au débit maximal du réseau divisé par deux. Le graphique montre les résultats de la question précédente avec émetteur unique en comparaison avec le test ayant deux émetteurs. Le partage du débit vient du fait qu'un réseau Ethernet avec hub est un réseau dit à diffusion.

13) latence

Information obtenues avec la commande ping entre le pc3 et pc2 :

Taille données (octets)	Temps aller-retour (ms)	Latence (ms)	Tprop (ms)	Temps protocole (ms)
64	0,403	0,2015	0,06	0,1415
1000	1,938	0,969	0,06	0,909
1472	2,708	1,354	0,06	1,294

NON. Tprop à 10^{-7} s.

Plus la taille des données est grande plus le temps d'émission est important ce qui implique une augmentation la latence (et donc du temps d'aller-retour).

14) Utilisation de commutateur (ou switch) :

Avec l'utilisation d'un switch, pour l'envoi d'une trame de taille maximale sur le réseau on observe des débits d'environ 95 Mbits/s aussi bien en émission qu'en réception. Le commutateur permet d'envoyer directement les paquets de la machine source à la machine de destination. Ainsi, sur un réseau composé de plusieurs machines, chaque communication se fait comme s'il n'y avait qu'un émetteur et récepteur.

Pour observer une baisse de débit sur un réseau avec commutateur, il faut provoquer une congestion. Nous avons essayé de créer une congestion sur notre réseau de quatre machines en envoyant deux flux d'informations sur la même machine. Cependant aucune baisse de débit n'a été observé lors de ce test. Ceci est dû à une caractéristique particulière des commutateurs : *stop and forward*. Le commutateur possède une file d'attente qui permet de stocker les paquets le temps de leur réémission vers la machine réceptrice. La congestion survient quand la file d'attente est pleine, il y a alors destruction des paquets que l'on ne peut plus stocker. La destruction des paquets provoque une forte dégradation des performances. Pour éviter la perte de performance, les commutateurs détecte quand leur file d'attente est presque pleine et ils envoient des trames particulières (fausse trame : half-duplex et trame « pause » en full-duplex) sur le réseau pour réduire la vitesse des émissions et ainsi éviter la congestion.

Si! mais il faut regarder le débit en réception (divisé par 2): la moitié des paquets de chaque flux est perdue.