

# TD 9: Transmettre des données

## Module ArcSys

### Objectifs pédagogiques:

- Découvrir les deux méthodes de commutation au sein d'un réseau (E1, E2);
- Comprendre comment TCP évite les congestions (E3, E4);
- Découvrir quelques limitations de TCP (E5);
- Savoir comment détecter des erreurs de transmission (E6).

### ★ Exercice 1: Commutation de circuits et commutation par paquets

▷ **Question 1:** Rappelez le principe et l'utilité de la commutation de circuits et de la commutation par paquets.

Un délai de commutation sur un système à commutation de paquets en mode différé (store-and-forward) est le temps nécessaire à un commutateur pour recevoir complètement un paquet et le retransmettre.

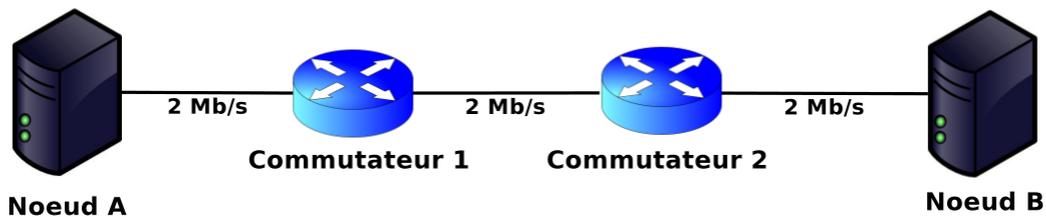
▷ **Question 2:** Un délai de commutation de  $10 \mu s$  affecterait-il beaucoup le temps de réponse sur un système client-serveur dont le client se trouve à New-York et le serveur en Californie, en admettant que la vitesse de propagation sur cuivre ou fibre soit égale à  $2/3$  de la vitesse de la lumière dans le vide ?

Supposons que différents utilisateurs se partagent une liaison à  $1 \text{ Mb/s}$ , que chaque utilisateur utilise  $100 \text{ kb/s}$  au moment de transmettre des données, mais qu'il ne soit actif que  $10\%$  du temps.

▷ **Question 3:** Combien d'utilisateurs peut-il y avoir en commutation de circuits ?

### ★ Exercice 2: Commutation par paquets avec segmentation de messages

Soit le réseau décrit par la figure ci-dessous. On considère que le réseau ne présente pas d'encombrement et que le nœud A veut envoyer à B un message d'une longueur de  $7,5 \text{ Mbits}$ .



▷ **Question 1:** Combien de temps faut-il au message pour rejoindre B si A ne segmente pas le message (commutation de message) ?

▷ **Question 2:** Combien de temps faut-il au message pour rejoindre B si A segmente le message en  $5000$  paquets de  $1500$  bits chacun ?

### ★ Exercice 3: Slow start

▷ **Question 1:** Rappelez ce qu'est le contrôle de flux de TCP et le principe du *slow start*.

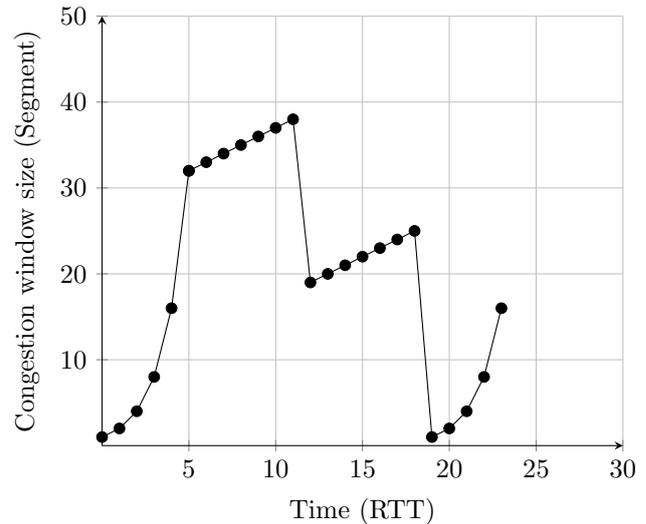
Rappelons que le Maximum Segment Size (MSS) est la quantité maximale d'octets pouvant être envoyés par un segment (équivalent au MTU moins la taille des en-têtes IP et TCP) et que le Round-Trip Time (RTT) est le délai de transmission pour envoyer un paquet plus le délai pour recevoir le ACK.

▷ **Question 2:** Considérons une machine *A* transférant un fichier de longueur  $L$  à une machine *B* en utilisant le protocole TCP (avec *slow start*). Soit  $R$  le RTT entre *A* et *B*,  $W$  la taille de la fenêtre de réception annoncée par *B* avec  $W = 4 \times \text{MSS}$ , et  $L = 10 \times \text{MSS}$ . En ignorant l'initialisation de la connexion, sa terminaison, et en considérant qu'il n'y a aucune erreur ou perte de paquet, combien de temps est nécessaire pour envoyer le fichier et s'assurer qu'il soit reçu par *B* ? Pour cette question il est intéressant de dessiner un diagramme de séquence montrant l'envoi de chaque paquet et l'envoi de chaque ACK.

### ★ Exercice 4: Congestion avoidance

▷ **Question 1:** Rappelez le principe du *congestion avoidance*.

▷ **Question 2:** Considérons une machine  $A$  transférant un fichier de longueur  $L$  à une machine  $B$  en utilisant le protocole TCP (avec *slow start* et *congestion avoidance*). Soit  $R$  le RTT entre  $A$  et  $B$ ,  $W$  la taille de la fenêtre de réception annoncée par  $B$  avec  $W = 8 \times \text{MSS}$ ,  $L = 30 \times \text{MSS}$  et  $ssthresh = W$ . En ignorant l'initialisation de la connexion et sa terminaison, combien de temps est nécessaire pour envoyer le fichier et s'assurer qu'il soit reçu par  $B$  en sachant que le 22ème paquet (numéro de séquence  $21 \times \text{MSS}$ ) a été perdu ? Pour cette question il est intéressant de dessiner un diagramme de séquence montrant l'envoi de chaque paquet et l'envoi de chaque ACK.



▷ **Question 3:** Identifiez les intervalles où *slow start* et *congestion avoidance* sont en action.

### ★ Exercice 5: Mais tout n'est pas parfait...

▷ **Question 1:** Supposons une connexion TCP avec une liaison fournissant une bande passante de  $B = 1 \text{ Gb/s}$  avec un RTT  $R = 100 \text{ ms}$ . Quel est l'effet indésirable se produisant ?

▷ **Question 2:** Supposons une connexion TCP qui utilise une liaison (e.g. Wi-Fi) où des erreurs surviennent aléatoirement. Que va-t-il se passer ?

▷ **Question 3:** Supposons un modèle client/serveur où le client envoie 1 Ko de données au serveur. Le client a utilisé les appels systèmes `socket`, `connect`, `send` et `close`, aucun n'a retourné d'erreurs et il a utilisé les paramètres nécessaires pour utiliser le protocole TCP. Expliquez pourquoi :

1. Le serveur pourrait ne pas avoir reçu les données
2. Le serveur pourrait avoir reçu des données corrompues

▷ **Question 4:** Quelle est la taille maximale de données transférées dans une session TCP ? Que ce passe-t-il si on veut transférer plus ? Pouvez-vous imaginer un problème avec cette méthode ?

### ★ Exercice 6: Codes détecteurs/correcteurs d'erreur (optionnel).

Pour détecter les erreurs de transmission, on peut utiliser une parité à deux dimensions ( $n$  lignes de  $k$  bits auxquels on ajoute un bit de parité à chaque ligne et chaque colonne).

▷ **Question 1:** Rappelez dans quel cas un seul bit de parité pour tout un message ne permet pas de détecter les erreurs ?

▷ **Question 2:** Ce dispositif permet-il de détecter toutes les erreurs simples ? Doubles ? Triples ? Donner des exemples.

Un flux binaire 10011101 est transmis avec la méthode CRC. Le générateur polynomial est  $x^3 + 1$ .

▷ **Question 3:** Quelle est la chaîne binaire réellement transmise ? Si le troisième bit à partir de la gauche est inversé durant la transmission, est-il détecté par le destinataire ?

▷ **Question 4:** Pourquoi les protocoles de liaison de données placent-ils toujours le CRC en fin de trame et jamais au début ?