

Réseaux

Nicolas Bourras

September 13, 2020

Contents

0.1	Premier cours	2
0.1.1	Informations utiles	2
0.1.2	Introduction & rappels	2

0.1 Premier cours

9 septembre 2020, 10h

0.1.1 Informations utiles

Répartition horaire :

- 12h de cours
- 12h td
- 14h tp (avec une partie projet)

Le professeur en CM est Djamal Habet, et le responsable des TPs est Emmanuel Godard.

Evaluation :

- Session 1: $\frac{2}{3}ET + \frac{1}{3}TP$
- Session 2: $Max(ET, \frac{2}{3}ET + \frac{1}{3}TP)$

Les scéances de TP contiendront un projet.

0.1.2 Introduction & rappels

L'UE réseau de la L3 se concentrait sur le modèle application, celle de cette année va plus s'intéresser au fonctionnement du réseau dans sa globalité.

Contenu du cours :

- Introduction & notions de base
- Modèle OSI
- Pile TCP/IP
- Applications

Note : Un graphe permet de représenter facilement un réseau.

Taille des réseaux

Les réseaux ont une taille (géographique) :

- un bus est à une échelle très petite (eg: carte mère)
- un LAN ne va que de quelques m à quelques km
- un MAN (Metropolitan Area Network) est à l'échelle d'une ville

par exemple un LAN est local seulement qui va de quelques mètres à quelques km, un PAN est un réseau personnel, un WAN (Wide Area Network) est plus large (pays/monde), un bus est très petit (sur une carte mère par exemple), un MAN (Metropolitan Area Network) est à l'échelle d'une ville, ...

Topologie

Très détaillé sur le transparent.

On a plusieurs topologies pour les réseaux, comme : - en diffusion (eg: bus, anneau, satellite), comme pour la radio; C'est l'adresse spécifique placée dans le message qui permet de déterminer qui va le recevoir. Ce mode est très utilisé pour les réseaux locaux (eg: en salle de TP). On peut représenter ce réseau par un arbre. - en point à point (eg: étoile, boucle simple, boucle double, maillage régulier, maillage irrégulier)

On va donc utiliser des algorithmes comme dijkstra pour router les messages.

Communication

Avec connexion Comme pour le téléphone, l'émetteur demande l'établissement d'une connexion par l'envoi d'un bloc de données spécial. Si la connexion est acceptée, elle est établie par la mise en place d'un circuit virtuel.

Exemple: TCP

Sans connexion On appelle les blocs de données datagrammes. Ils sont transmis sans vérifier qu'un circuit existe, il suffit juste d'une adresse de destination.

Exemple: UDP

Commutation

Commutation de circuits Consiste à créer dans le réseau un circuit entre l'émetteur et le récepteur avant que ceux-ci commencent à échanger des informations.

C'est ce que les opératrices au tout début du téléphone faisaient.

Commutation de messages Consiste à envoyer un message de l'émetteur jusqu'au récepteur en passant de noeud de commutation en noeud de commutation.

On risque d'avoir des problèmes avec les messages longs. -> taux d'erreurs résiduel

Commutation de paquets Consiste à découper un message en paquets avant de l'envoyer. On évite ainsi les erreurs avec les messages longs.

Problème: il faut que le récepteur final sache reconstituer le message, et donc il faut un protocole particulier.

On va devoir avoir plusieurs champs dans chaque paquet: - un ID de message - un nombre pour l'ordre - un booléen pour savoir si c'est le dernier paquet

Commutation des cellules Pareil que la commutation de paquets, avec une taille fixée de paquet de 53 octets (5 d'en-tête et 48 de données).

La faible taille, permet de limiter les erreurs.

La normalisation

-> distinction norme/standard, une norme est établie par un organisme donc c'est officiellement le rôle alors qu'un standard est rédigé par une entité non reconnue -> rôle des organismes de normalisation: définir un cadre, souvent nommé modèle, et aussi garantir la complétude & l'intégrité des spécifications
-> organismes les plus connus: ISO, UTI (ex CCITT), IEEE

Le modèle OSI => Operated Open System

Architecture en couche Double objectif: - avoir à remplacer seulement le ou les modules nécessaires - se procurer les modules chez différents fournisseurs
insérer schéma en U du modèle OSI

à l'envoi d'un message, on le fait passer par toutes les couches. Chaque couche ajoute son en-tête (la couche liaison ajoute aussi à la fin du message).

à la réception, on fait l'inverse.

On a 7 couches: - application 7 - présentation 6 - session 5 - transport 4 - réseau 3 - liaison 2 - physique 1

Les 4 premières couches sont orientées réseau (en général faites par l'OS), les 4 dernières sont orientées application.

Attention: ce modèle est **théorique**. On peut se permettre de changer les choses un peu.

Par exemple, les couches physique et les couches liaison sont un peu liées (pas totalement séparées).

La couche physique Elle fournit les moyens mécaniques, électriques, fonctionnels et procédurax nécessaires à l'activation des connexions physiques destinées à la transmission de bits entre deux entités de liaison de données.

=> On transmet du binaire.

Sens de transmission

-> mode simplex (eg: radio) -> mode semi-duplex (eg: talkie walkie) -> mode duplex (eg: websocket)

Transmission parallèle

Les bits d'un même caractère sont envoyés en même temps chacun sur un fil distinct.

Pose des problèmes de synchronisation et n'est utilisé que sur des courtes distances (eg: bus).

Transmission en série

Les bits sont envoyés les uns derrière les autres de manière synchrone ou asynchrone.

Synchrone -> l'émetteur et le récepteur se mettent d'accord sur une base de temps régulier. Asynchrone -> pas de négociations, chaque caractère est précédé d'un bit de start et suivi d'un bit de stop. de nos jours, un seul bit de start suffit plus

Débits Quel que soit le mode de transmission, on peut compter le débit de la ligne en Bauds : le nombre de bits d'horloge en une seconde.

Le nombre de bauds correspond au nombre d'émissions autorisés, donc: Bauds
= Bits/S
Si le signal prend 2^n valeurs distinctes, on dit que sa valence est de n.