

# Licence L3 Informatique

## HLIN503 -Réseaux

2015-2016

## 1 Transfert de données

### Exercice 1

1. Calculer la taille d'une image couleur de  $1024 \times 1024$  points (*pixels*), avec un codage de la couleur en 32 bits. Dans la suite, on prend en compte le transfert de 10 images de ce type, sans compression.
2. Quelle est la limite inférieure du temps de transmission de ces 10 images, sur une ligne à  $55600 \text{ bit.s}^{-1}$  (vitesse classique dans les communications par *modems* sur le réseau téléphonique avec des modems non spécialisés) ? à  $1 \text{ Mbit.s}^{-1}$  (modem «adsl») ? à 10 et  $100 \text{ Mbit.s}^{-1}$  (vitesse classique dans les réseaux locaux) ?  
Est-il possible d'atteindre cette limite ?
3. Réflexion sur le contenu des images : Dans quelles conditions peut-on transmettre une vidéo (25 images par seconde), sur un réseau à  $100 \text{ Mbit.s}^{-1}$  ?
4. Quels sont les différents moyens que vous connaissez pour transférer des fichiers d'une machine à une autre, site local ou distant ? Tenir compte de tous les types de fichiers. Indiquer pour chaque catégorie les transformations à effectuer avant de transférer.  
Comment peut-on transférer des sous-arborescences ?

## 2 Internet : Affectation de noms et adresses

### Exercice 2

On a vu en cours que le nommage utilisé dans l'*Internet* était un nommage par domaine utilisant le caractère . (point) comme séparateur. À chaque hôte était associé un nom sous la forme  
`nom_simple.sous_domaine....domaine.domaine_racine` .

Le domaine racine correspond souvent à un pays ou à un domaine d'activité, géré par une entité de gestion. Le nom du domaine est déposé par chaque organisation cherchant à adhérer à ce réseau. Ensuite, toute organisation peut décider librement de ses sous-domaines.

En fait, cette démarche dans le nommage n'est pas spécifique à l'*Internet*. Toute organisation de réseau passe par une étape de mise en place d'un système d'identification, donc de nommage. Le nommage par domaines avec le point comme séparateur est spécifique de l'*Internet*.

Le nommage constitue donc un arbre, où les feuilles représentent des hôtes (les dénominations hôte, machine ou ordinateur sont équivalentes).

1. pourquoi n'utilise-t-on pas ces noms au niveau des paquets de la couche *réseau* plutôt que les adresses, qui sont ici des entiers de 32 bits ?
2. pour quelles raisons doit-on accepter que des hôtes aient plusieurs noms ?
3. pour quelles raisons doit-on accepter que des hôtes aient plusieurs adresses pour un même nom ?
4. finalement, peut-on déduire que les hôtes peuvent avoir  $n$  noms et  $a$  adresses, sans liaison entre  $n$  et  $a$  ?

Dans un plan d'adressage on associe un nom et une adresse (au moins) à chaque hôte. En fait, une adresse désigne une connexion à un réseau plutôt qu'une machine (cf. ouvrage de D. Comer, dans la bibliographie du cours). Justifier ceci en utilisant ce qui vient d'être vu ci-dessus ? Dans cette citation, le terme réseau désigne à votre avis un réseau physique (vu de la couche dite *liaison de données*) ou logique ("virtuel", vu de la couche dite *réseau*) ?

### Exercice 3

Deux hôtes,  $M_1$  et  $M_2$  sont sur un même réseau local et ont une seule connexion au réseau chacune. On suppose que c'est un réseau de type *ethernet* (quelle couche ?). Ce réseau est relié à l'*Internet*.

Les numéros IP (Internet Protocol) respectifs sont 197.25.26.27 et 197.25.26.100. On rappelle que ce sont des adresses codées sur 32 bits. Les adresses relatives à la couche liaison (*ethernet* dans cet exercice) sont 8:4:CF:20:36:AB et 7:20:FE:10:20:48. Cette représentation d'adresses comportant 48 bits est celle utilisée dans le protocole *ethernet*. Elle consiste à représenter les octets sous une forme hexadécimale, et à les séparer par le caractère ":". Ceci permet de distinguer entièrement des formats d'adresses différentes.

1. Transformer les adresses ci-dessus en entiers, exprimés sous une forme hexadécimale, binaire et décimale. Peut-on déduire que la représentation choisie est plutôt commode ? Sinon, proposer une représentation plus agréable.
2. Décrire rapidement la forme générale d'un paquet au niveau de la couche dite *réseau*. On veut s'intéresser surtout aux adresses sous toutes leurs formes dans la suite. Un paquet  $p_1$  part de  $M_1$  à destination de  $M_2$ . Décrire le contenu de ce paquet dans la couche dite *réseau* avant qu'il ne quitte cette couche sur  $M_1$ .
3. Même question dans la couche dite *liaison de données* sur  $M_1$ . Par abus de langage on appellera les adresses relatives à cette couche *adresses physiques*.
4. Décrire ce qui se passe sur  $M_2$  lors de la réception du paquet.
5.  $M_3$  est une autre machine sur ce même réseau local. Son adresse IP est 197.25.26.129. Décrire ce qui se passe sur  $M_3$  lors de la réception de ce paquet.
6. Si **par erreur**  $M_3$  se trouve avoir la même adresse physique que  $M_2$ , que se passe-t-il ?

### Exercice 4

On suppose maintenant que  $M_1$  et  $M_2$  sont sur deux réseaux locaux distincts, reliés par un intermédiaire, appelé *routeur*  $M_r$ . Prendre pour adresse IP de  $M_2$  195.2.4.8 et ignorer la référence précédente. Le routeur  $M_r$  a pour adresses IP respectives 197.25.26.47 et 195.2.4.54. Ses adresses physiques sont 8:25:aa:bb:cc:ee et a0:37:gg:ab:cd:ef.

1. Faire un schéma (rapide) de ces connexions, en associant des noms aux diverses machines. Attribuez-vous un ou plusieurs noms à  $M_r$  ?
2. Corriger les adresses incorrectes si ce n'est pas encore fait.
3. Décrire le cheminement d'un paquet  $p_2$  cheminant de  $M_1$  vers  $M_2$ ,
4. On reconsidère  $M_3$ , située sur le même réseau physique que  $M_1$ . Quelle est sa vision de ce paquet  $P_2$ , i.e. le reçoit-elle et si oui qu'en fait-elle ?

Construire un exemple complet où deux routeurs  $M_p$  et  $M_q$  séparent les hôtes  $M_1$  et  $M_2$ , et décrire le cheminement d'un paquet de  $M_1$  vers  $M_2$ , puis de  $M_2$  vers  $M_1$ .

Si  $M_p$  tombe en panne, que se passe-t-il ? Il faut considérer ici tous les cas possibles :

1. le paquet est parti de  $M_1$  et  $M_p$  tombe en panne,
2.  $M_p$  a reçu le paquet mais ne l'a pas expédié à  $M_q$ ,
3. etc.

### 3 Internet : correspondance nom et adresses d'hôtes

#### Exercice 5

Après avoir vu en TP la correspondance entre le nom et l'adresse d'un hôte, on a constaté qu'on pouvait se connecter à un hôte local alors qu'il ne figure pas dans le serveur de noms.

1. N'y a-t-il pas plusieurs bases d'informations différentes consultées lors de la recherche de cette correspondance et quelles sont ces bases de recherche d'information ?

#### Question d'examen: Exercice 6

Une application sur un hôte lance une recherche DNS avec un nom inexistant. Décrire ce qui se passe en envisageant plusieurs erreurs possibles :

1. Le domaine de premier niveau est inexistant (par exemple, l'adresse se termine avec .aargh);
2. le domaine de premier niveau existe, le premier sous-domaine aussi, et l'erreur est ensuite dans le sous-domaine en 2<sup>ème</sup> position ;
3. seul le nom d'hôte contient une erreur ;
4. profiter de cette analyse pour généraliser.

**Conseil :** Essayer en TP ces diverses erreurs.

### 4 Transport de paquets : acheminement, ordre ,fiabilité et duplication

#### Question d'examen: Exercice 7

Est-ce que deux paquets UDP suivent le même chemin pour un acheminement d'une même source à une même destination ? Deux paquets TCP ? Deux paquets IP ?

#### Question d'examen: Exercice 8

On a vu en cours que le protocole de transport *tcp* se chargeait de remettre les paquets qu'il reçoit dans l'ordre, afin de délivrer à l'application destinataire les données dans le même ordre que celui de départ. S'il y a besoin de rétablir l'ordre, c'est qu'il peut être perturbé.

- Montrer comment deux paquets *tcp*,  $p_1$  et  $p_2$  expédiés dans un ordre donné peuvent arriver en ordre inverse dans la couche *tcp* de destination. Donner deux exemples amenant à ce type de «désordre».
- Décrire une configuration de réseau (et/ou un cas) dans lesquels l'ordre de réception est forcément identique à celui d'expédition.
- Est-ce que cette situation peut changer dans le cas d'un protocole de transport comme *udp* ?

#### Question d'examen: Exercice 9

Une application a besoin d'un protocole orienté message, sans connexion, fiable. Elle est destinée à tourner dans le monde *Internet* et adopte le protocole *UDP*. Comme dans le cas ci-dessus, on suppose que l'application est implantée directement sur cette couche de transport. Elle doit donc mettre en place les éléments permettant d'assurer la fiabilité. Un système d'accusé de réception est ainsi mis en place. On suppose de plus qu'une primitive *y-a-un-message* lui permet de savoir si un message est disponible.

**Note :** on ne veut pas s'occuper pour l'instant de la possibilité de duplication : on admet donc qu'un message peut être dupliqué et que ça ne gêne pas l'application.

1. A-t-elle besoin de la primitive *y-a-un-message* pour la mise en place de la fiabilité ?
2. On suppose qu'un accusé de réception est envoyé à chaque message. Quel est le défaut le plus important à votre avis de ce choix ?

3. Que se passe-t-il si après plusieurs envois on n'a toujours pas d'accusé de réception ? Que faut-il faire ? La solution consistant à prendre une paire de ciseaux pour tester la solidité du réseau doit être écartée en tant que prématurée.
4. Finalement, si l'on veut éviter la duplication que faut-il mettre en place ?

### Question d'examen: Exercice 10

On a vu en cours que la **duplication** de paquets était gérée par le protocole `tcp` et non gérée par `udp`.

Soient  $A_1$  et  $A_2$  deux applications. Construire un exemple montrant comment une **duplication** de paquet peut se produire dans chacun des cas ci-dessous :

1.  $A_1$  et  $A_2$  communiquent par `tcp`. Indiquer ce qui se passe au niveau des applications même.
2.  $A_1$  et  $A_2$  communiquent par `udp`.

### Question d'examen: Exercice 11

Bien qu'un protocole comme `udp` soit non fiable, dans le cas d'un réseau local un paquet ne peut se perdre que dans quelques cas précis. On admettra ici qu'il ne peut se perdre à cause d'un défaut électronique (*pas de disparition due au matériel*).

1. Illustrer deux exemples de perte de paquets `udp` dans un réseau local composé d'un ensemble d'hôtes connectés et ne nécessitant pas de routeur. On admettra que les applications qui communiquent fonctionnent correctement
2. Si on ajoute un routeur, illustrer un nouvel exemple de perte de paquet.
3. Reprendre les trois exemples des questions ci-dessus et expliquer dans chaque cas ce qui se serait passé si le protocole était `tcp`.