**Couche de Liaison**

**Objectifs du laboratoire**

Appliquer les connaissances acquises dans la partie théorique à la conception d’un ensemble de protocoles de la couche de liaison.

**Introduction**

Dans ce laboratoire, vous devrez concevoir certaines fonctionnalités d’une couche de Liaison pour satisfaire le cahier des charges suivant (les protocoles que nous avons vus en classe doivent en général être modifiés pour les rendre plus adaptées et efficaces dans notre cas) :

1. Offrir à la couche 3 un service de transmission de PDUs. Le personnel enseignant jouera le rôle de cette couche (la couche réseaux donc) lors de l’évaluation de votre protocole en vous passant une primitive de service Data.Request.

2. Le personnel enseignant aura aussi le contrôle de la couche physique en ce sens qu’il pourra y introduire des interruptions ou des erreurs en tout moment.

3. La primitive contiendra la donnée à transmettre et l’identifiant du destinataire.

4. La trame doit pouvoir être adressée à un seul système (unicast), à tous les systèmes (broadcast) ou à un sous-ensemble de systèmes (multicast).

5. Le nombre maximum de systèmes, y compris le vôtre, sera 7. Au moins 5 groupes multicast doivent pouvoir être définis.

6.La cargaison (payload) de votre protocole (pas celui des trames Ethernet) aura une longueur fixe de 56 bits. Votre protocole doit pouvoir corriger toutes les erreurs sur un bit dans ce payload.

7. Votre protocole doit également pouvoir détecter toutes les erreurs sur deux bits dans le payload.

8. Votre protocole doit également pouvoir détecter toutes les erreurs sur trois bits dans le payload.

9. Votre protocole doit pouvoir détecter toutes les rafales d’erreurs de longueur 8 bits dans le payload.

10. Votre protocole doit pouvoir détecter toutes les erreurs simples dans chacun des champs dans l’entête.

11. Vous devez développer un script pour transmettre une trame :

**Pour la transmission**

Le script aura deux arguments :

1. L’adresse de destination

2. Le payload

Pour ce labo, vous pourrez calculez le payload à la main.

La commande aura donc, par exemple, la suivante :

SendFrame(<adresse de destination>, <payload>)

12. Vous devez développer un script pour recevoir une trame :

**Pour la réception**

Le script devra afficher ou stocker dans un fichier les informations suivantes :

1. L’adresse de destination

2. Le payload

Vous devrez ensuite pouvoir calculer à la main les données transportées.

La forme de la commande sera donc, par exemple, la suivante :

ReceiveFrame()

et la sortie sera : <adresse> et <payload>

13. Dans ces deux derniers pas, vous devez automatiser la préparation de la trame pour a transmission et l’extraction des données pour la réception.

Pour la transmission, votre script pourra donc être de la forme :

SendFrameAutom(<adresse de destination>, <données>)

Pour la réception, votre script pourra donc être de la forme :

ReceiveFrameAutom()

La sortie doit être <adresse de destination> et <données>

Le diagramme à la figure 1 représente schématiquement la situation que vous allez rencontrer physiquement au laboratoire le jour de la présentation finale de votre couche 2.



Fig. 1. Réseau sur lequel votre couche de Liaison doit fonctionner.

L’architecture en couches est illustrée à la figure 2.



Fig. 2. Architecture en couches du protocole. A gauche, modèle OSI. A droite, modèle correspondant à votre système.

Vous allez utiliser les services de la couche physique de 802.3 (Ethernet) pour transmettre vos trames. Pour que les trames soient envoyées par cette couche physique, il faudra les encapsuler dans une trame MAC 802.3. Cette encapsulation est la responsabilité de la sous-couche d’adaptation montrée à la figure 2.

La sous-couche d’adaptation doit encapsuler les trames de votre protocole dans des trames MAC 802.3 ayant comme adresse de destination l’adresse broadcast 802.3 (ff:ff:ff:ff:ff:ff). Vous pouvez définir le type de la trame. Le type dans les trames Ethernet se trouve dans les deux octets qui suivent l’adresse de destination MAC.

Une fois la couche de Liaison conçue et implémentée, le réseau doit fonctionner comme suit :

1.- La couche de réseaux, exécutée par le personnel enseignant, vous donnera une primitive de service Data. Request contenant des données et l’identifiant du système destinataire selon votre propre système d’adressage.

2. Vous devrez alors construire une trame suivant le protocole de votre couche de Liaison.

3. La trame doit ensuite être encapsulée dans une trame MAC 802.3 et injectée au niveau de la couche physique pour transmission par le câble 802.3 vers le HUB/Switch (voir figure 1).

La trame sera reçue par les autres systèmes dans le réseau, où elle atteindra les entités de votre couche de Liaison. Vous devrez alors la traiter pour détecter et corriger les erreurs éventuelles. Une fois satisfait-e-s de l’intégrité de la trame, vous passerez les données à la couche de réseau dans une primitive de service Data.Indication.

Step 1: In the first step, we try to write a code which will get some bits of data from a user and automatically send it to another machine. The other machine should be able to receive the sent packet using WireShark.

Step 2: The addressing scheme explained in the Introduction section, items 4 and 5, must be implemented.

Step 3: Follow the steps that you have taken before and prepare an algorithm and code which would be able to do the tasks described in the introduction (points 7 to 10).

Step 4: Follow the steps that you have taken before and prepare an algorithm and code which would be able to do the tasks described in the introduction (points 11 and 12).

Step 5: Follow the stepss that you have taken before and prepare an algorithm and code which would be able to do the tasks described in the introduction (points13).