**TD analyse de trame Ethernet**

# Contexte de travail

# Sur un réseau local d’architecture Ethernet à 10 Mbps, un analyseur de trames (de type Wireshark) a capturé la trame présentée en annexe 3. Il s’agit d’une trame Ethernet II encapsulant un datagramme IP, lui-même encapsulant un segment TCP.

# L’analyse du segment TCP permet de trouver les éléments d’une communication entre un navigateur et un serveur HTTP.

# Travail à réaliser

# En vous aidant des indications portées en annexes 1 et 2, compléter la structure des unités de données présentée dans les tableaux de l’annexe 4.

# ANNEXE 1 : Analyse d’une trame ETHERNET

Dans un réseau Ethernet, l’unité de données utilisée au niveau LIAISON est appelée trame. La structure de la trame utilisée dans l’exemple qui suit est de type Ethernet II. Une telle trame contient d’abord l’adresse de l’émetteur et du destinataire des données contenues dans la trame. Ces adresses sont celles des cartes réseau des postes émetteur et destinataire (ou adresses MAC). La trame contient ensuite l’identification du protocole de niveau RESEAU utilisé dans la partie *Données* de la trame, ou type de paquet (comme le nom de l’unité de données le plus courant au niveau RESEAU). Le type de paquet est une valeur hexadécimale. La valeur 0800 correspond au protocole IP. Cette information est destinée à aiguiller les données qui suivent vers le logiciel de gestion du protocole adapté.

La partie *Données* de la trame contient les informations gérées par la couche RESEAU, soit l’unité de données de niveau 3. Dans le cas du protocole IP, l’unité de données est appelée datagramme.

Le datagramme débute par l’identification du protocole utilisé au niveau TRANSPORT puis suivent des informations de gestion du datagramme IP comme le temps de vie (TTL : Time To Live) et les adresses IP source et destination. Puis suit l'identification du protocole de niveau TRANSPORT. Ainsi dans l’exemple, la valeur du type de protocole est 6, ce qui signifie que le datagramme IP contient une unité de données de type TCP. Dans le cas d’une communication UDP, la valeur aurait été 17. Ensuite le datagramme contient l’unité de données de niveau TRANSPORT. Il s’agit donc dans notre exemple, d’un segment TCP.

Le segment TCP débute par l’indication du type d’application émetteur et destinataire des données, ce sont les numéros de port TCP utilisés. Dans notre cas, la station émettrice ouvre son port 1035 pour communiquer avec le port 80 de la station destinataire. Le port 80 correspond généralement au serveur HTTP.

Les autres informations du segment TCP sont donc destinées à l’application qui gère le port 80. Elles répondent à un format défini qui est HTTP. C’est donc à ce niveau que les informations de l’application (dans notre cas, le navigateur) vont être échangées. On peut y voir en clair la version du client qui est utilisé (MSIE 4.01 sous Windows 98), ainsi que l’adresse IP du serveur qui est en communication.

# ANNEXE 2 : Le Protocole IP

*(Extrait du site internet commentcamarche.net)*

Le **protocole IP** fait partie de la couche Internet de la suite de protocoles TCP/IP. C'est un des protocoles les plus importants d'Internet car il permet l'élaboration et le transport des datagrammes IP (les paquets de données), sans toutefois en assurer la « livraison ». En réalité, le protocole IP traite les datagrammes IP indépendamment les uns des autres en définissant leur représentation, leur routage et leur expédition.

Le protocole IP détermine le destinataire du message grâce à 3 champs :

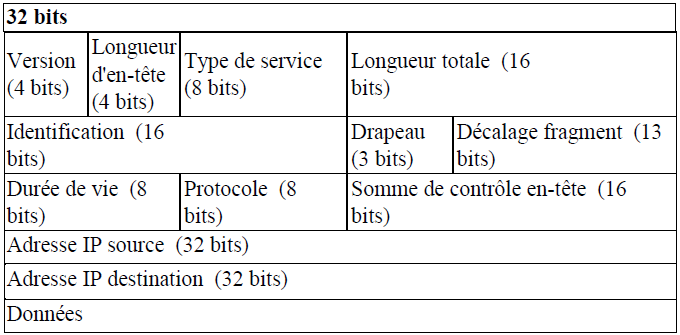
* Le champ adresse IP : adresse de la machine
* Le champ masque de sous-réseau : un masque de sous-réseau permet au protocole IP de déterminer la partie de l'adresse IP qui concerne le réseau
* Le champ passerelle par défaut : Permet au protocole Internet de savoir à quelle machine remettre le datagramme si jamais la machine de destination n'est pas sur le réseau local

**Les datagrammes**

Les données circulent sur Internet sous forme de datagrammes (on parle aussi de paquets). Les datagrammes sont des données encapsulées, c'est-à-dire des données auxquelles on a ajouté des en-têtes correspondant à des informations sur leur transport (telles que l'adresse IP de destination).

Les données contenues dans les datagrammes sont analysées (et éventuellement modifiées) par les routeurs permettant leur transit.

**Schéma d’un datagramme (IP – couche 3):**



**Description des différents champs :**

* **Version** (4 bits) : il s'agit de la version du protocole IP que l'on utilise (actuellement on utilise la version 4 *IPv4*) afin de vérifier la validité du datagramme. Elle est codée sur 4 bits.
* **Longueur d'en-tête**, ou IHL pour Internet Header Length (4 bits) : il s'agit du nombre de mots de 32 bits constituant l'en-tête (nota : la valeur minimale est 5). Ce champ est codé sur 4 bits.
* **Type de service** (8 bits) : il indique la façon selon laquelle le datagramme doit être traité.
* **Longueur totale** (16 bits): il indique la taille totale du datagramme en octets. La taille de ce champ étant de 2 octets, la taille totale du datagramme ne peut dépasser 65536 octets. Utilisé conjointement avec la taille de l'en-tête, ce champ permet de déterminer où sont situées les données.
* **Identification, drapeaux (flags) et déplacement de fragment** sont des champs qui permettent la fragmentation des datagrammes, ils sont expliqués plus bas.
* **Durée de vie appelée** aussi **TTL**, pour **T**ime **T**o **L**ive (8 bits) : ce champ indique le nombre maximal de routeurs à travers lesquels le datagramme peut passer. Ainsi ce champ est décrémenté à chaque passage dans un routeur, lorsque celui-ci atteint la valeur critique de 0, le routeur détruit le datagramme. Cela évite l'encombrement du réseau par les datagrammes perdus.
* **Protocole** (8 bits) : ce champ, en notation décimale, permet de savoir de quel protocole est issu le datagramme o ICMP : 1 IGMP : 2 o TCP : 6 UDP : 17
* **Somme de contrôle de l'en-tête**, ou en anglais **header checksum** (16 bits) : ce champ contient une valeur codée sur 16 bits qui permet de contrôler l'intégrité de l'entête afin de déterminer si celui-ci n'a pas été altéré pendant la transmission. La somme de contrôle est le complément à un de tous les mots de 16 bits de l'en-tête (champ somme de contrôle exclu). Celle-ci est en fait telle que lorsque l'on fait la somme des champs de l'en-tête (somme de contrôle incluse), on obtient un nombre avec tous les bits positionnés à 1
* **Adresse IP source (**32 bits) : Ce champ représente l'adresse IP de la machine émettrice, il permet au destinataire de répondre
* **Adresse IP destination** (32 bits) : adresse IP du destinataire du message

# ANNEXE 3 : Exemple de Trame Ethernet

*Contenu et analyse d’une trame.*

*La trame TCP/IP est constituée d’une suite de bits. Il convient de décomposer son contenu selon les couches du modèle OSI ou DoD.*

# 

# ANNEXE 4 : Structure des unités de données

*Cette annexe décrit partiellement les informations contenues dans la trame Ethernet et dans le datagramme IP encapsulé.*

# *Trame Ethernet*

|  |  |  |
| --- | --- | --- |
| **INFORMATION** | **TAILLE** | **Valeur hexadécimale correspondante sur le document de l’annexe 1** |
| **Préambule** de synchronisation | 7 octets | *non présenté dans ce document* |
| **Délimiteur** de début de trame | 1 octet | *non présenté dans ce document* |
| Adresse **MAC de l’émetteur** | 6 octets | 00 00 C0 B2 AB BF |
| Adresse **MAC du destinataire** | 6 octets | 00 60 08 5C 61 6E |
| **Type du paquet** (IP, AppleTalk...) | 2 octets | 0800 |
| **Données** | 293 octets | Contenu du datagramme IP.  *Possibilité de compléter le champ d'une séquence de bourrage si le nombre d'octets est inférieur à 46.* |
| **Champ de contrôle** calculé sur les données et l’en tête | 4 octets | **5E A9 B7 B1** *(dernière partie en noir)* |
| **TOTAL** | 319 |  |

# *Datagramme IP*

|  |  |  |
| --- | --- | --- |
| **INFORMATION** | **TAILLE** | **Valeur correspondante sur le document de l'annexe 1** |
| **Version** du protocole IP et longueur de l'en-tête | 8 bits | 45 |
| **Type de service** | 8 bits | 00 |
| **Taille** du paquet | 16 bits | 01 25 |
| Numéro **du paquet** | 16 bits | 89 00 |
| Drapeau indiquant si le paquet est fragmenté  + le numéro du fragment | 16 bits | 40 00 |
| Durée de vie | 8 bits | 80 |
| Protocole | 8 bits | 06 |
| Somme de controle en tête | 16 bits | FA 84 |
| Adresse ip source | 32 bits | C0 A8 FA FA |
| Adresse ip destination | 32 bits | C0 A8 FA 01 |
| TOTAL | 160 bits |  |

# *Segment TCP (facultatif)*

|  |  |  |
| --- | --- | --- |
| **INFORMATION** | **TAILLE** | **Valeur correspondante sur le document de l'annexe 1** |
| Numéro de port émetteur | 16 bits | 04 0B |
| Numéro de port récepteur | 16 bits | 00 50 |
| Numéro de séquence | 32 bits | 00 67 7C 92 |
| Numéro d'acquittement | 32 bits | 1E D1 1C 30 |
| **Taille d’en-tête** | 4 bits | 0101 (bin) |
| **Réservé** | 6 bits | 000000 (bin) |
| Code indiquant le **type du paquet** (urgent...) | 6 bits | 011000 (bin) |
| **Quantité de bits** que peut recevoir le destinataire | 16 bits | 22 38 |
| **Calcul d'erreur** sur les données et l'en-tête | 16 bits | 8D 22 |
| Pointeur pour données urgentes | 16 bits | 00 00 |
| Données | 253 |  |
| TOTAL | 273 |  |