

## TD 1 : Modèles en couches (OSI et TCP/IP)

### Exercice 1

- 1) Qu'est ce que l'encapsulation ?
- 2) Décodez la trame Ethernet suivante en vous servant des formats joints en annexe (ne donner queles champs en gras) :

```

0000 00 04 76 f0 fb b5 00 06 5b c2 f5 9e 08 00 45 00  ..v.....[.....E.
0010 01 4f 06 cf 40 00 40 06 b1 6f c0 a8 00 17 c0 a8  .O..@.@..o.....
0020 00 03 80 09 00 50 85 e6 67 33 03 6c 42 f4 80 18  ....P..g3.lB...
0030 16 d0 78 f1 00 00 01 01 08 0a 00 09 62 11 0b 5a  ..x.....b..Z
0040 6a 43 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 31  jCGET / HTTP/1.1
0050 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 3a 20 4b 65  ..Connection: Ke
0060 65 70 2d 41 6c 69 76 65 0d 0a 55 73 65 72 2d 41  ep-Alive..User-A
0070 67 65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 2f 35 2e  gent: Mozilla/5.
0080 30 20 28 63 6f 6d 70 61 74 69 62 6c 65 3b 20 4b  0 (compatible; K
0090 6f 6e 71 75 65 72 6f 72 2f 32 2e 32 2d 31 31 3b  onqueror/2.2-11;
00a0 20 4c 69 6e 75 78 29 0d 0a 41 63 63 65 70 74 3a  Linux)..Accept:
00b0 20 74 65 78 74 2f 2a 2c 20 69 6d 61 67 65 2f 6a  text/*, image/j
00c0 70 65 67 2c 20 69 6d 61 67 65 2f 70 6e 67 2c 20  peg, image/png,
00d0 69 6d 61 67 65 2f 2a 2c 20 2a 2f 2a 0d 0a 41 63  image/*, /*..Ac
00e0 63 65 70 74 2d 45 6e 63 6f 64 69 6e 67 3a 20 78  cept-Encoding: x
00f0 2d 67 7a 69 70 2c 20 67 7a 69 70 2c 20 69 64 65  -gzip, gzip, ide
0100 6e 74 69 74 79 0d 0a 41 63 63 65 70 74 2d 43 68  ntity..Accept-Ch
0110 61 72 73 65 74 3a 20 41 6e 79 2c 20 75 74 66 2d  arset: Any, utf-
0120 38 2c 20 2a 0d 0a 41 63 63 65 70 74 2d 4c 61 6e  8, *..Accept-Lan
0130 67 75 61 67 65 3a 20 66 72 2c 20 66 72 5f 46 52  guage: fr, fr_FR
0140 40 65 75 72 6f 2c 20 65 6e 0d 0a 48 6f 73 74 3a  @euro, en..Host:
0150 20 73 65 72 76 43 33 30 39 0d 0a 0d 0a  servC309....
    
```

- 3) à votre avis, quel est le protocole transporté à l'intérieur du segment TCP et quelle est l'application qui l'utilise ?

### Exercice 2

Dans un réseau basé sur le modèle de la pile de protocoles TCP/IP, on définit un protocole MDTP (*My Data Transfer Protocol*) de la couche application pour transférer des données entre 2 machines. Lorsqu'une machine veut transférer des données en utilisant ce protocole, elle émet un (ou plusieurs) datagramme contenant les données à transférer à la machine destinatrice. Ainsi, au niveau de MDTP le protocole se résume à un monologue (envoi unidirectionnel de PDU). Ce protocole utilise :

- au niveau de la couche transport le protocole UDP (*User Datagram Protocol*),
- au niveau de la couche internet le protocole IP
- et au niveau de la couche réseau des trames Ethernet.

Quelle est l'efficacité du transfert, en sachant que :

- les données à transférer occupent 3 Ko
- l'entête MDTP a une taille fixe de 60 octets
- l'entête UDP a une taille fixe de 8 octets
- le format des paquets IP et des trames Ethernet vous sont données dans l'annexe le protocole IP n'utilise pas son champ « Options »

## Annexe

### Structures de données des trames Ethernet, paquets IP et ARP, et segments TCP et UDP

Sur un réseau circulent des trames. Ces trames sont des messages interprétables quand on connaît la règle du jeu (le protocole). Chaque trame est composée d'une partie entête et d'une partie information et que cette partie information est un message que l'on peut aussi décoder si on connaît le protocole correspondant.

#### Décodage des trames Ethernet

Adresse destination (6 octets)	Adresse source (6 octets)	Type (2 octets)	Information (46 à 1500 octets)	Code correcteur (4 octets)
-----------------------------------	------------------------------	--------------------	-----------------------------------	-------------------------------

Cette structure est une première peau. À l'intérieur du champ « Information » se trouve aussi une structure représentant une unité de donnée de protocole de réseau NPDU (Network Protocol Data Unit). Lorsqu'il y a moins de 46 octets de données, ce champ est complété par des octets de bourrage pour atteindre sa taille minimale.

Le champ « Type » vous renseigne sur la manière de lire le contenu du champ « Information ». Ce contenu quand c'est un NPDU est un paquet. Ce paquet peut être

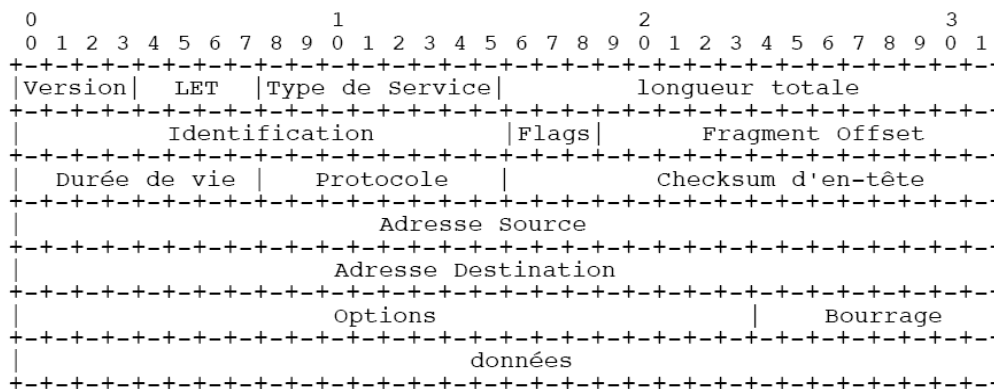
- un paquet IP si « Type » = 08 00
- un paquet ARP si « Type » = 08 06

Souvent le code correcteur n'est pas affiché lors des captures des trames.

#### Décodage de paquets IP

Un paquet IP (*Internet Protocol*) est composé : d'une entête et d'un contenu (données).

##### NPDU (paquet) du protocole IP



Notez que chaque marque indique une position bit.

- **Version** : 4 bits - format d'entête
- **LET** ou Longueur d'En-Tête : 4 bits - taille de l'entête en nombre de mots de 32 bits. La valeur la plus courante est 5. Soit 5\*4 = 20 octets.
- **Type de Service** : 8 bits - qualité de service
- **Longueur Totale** : 16 bits - longueur totale du paquet entête + données. Elle est exprimée en octets
- **Identification** : 16 bits - numéro d'identification servant au réassemblage des paquets
- **Flags** : 3 bits (Bit 0: réservé, doit être laissé à zéro ; Bit 1: (AF) 0 = Fragmentation possible, 1 = Non fractionnable ; Bit 2: (DF) 0 = Dernier fragment, 1 = Fragment intermédiaire)
- **Position relative** : 13 bits - situation du fragment dans le paquet
- **Durée de vie** : 8 bits - temps maximal que le paquet peut rester dans le réseau (si 0, paquet détruit)
- **Protocole** : 8 bits - indique quel protocole de niveau supérieur est utilisé dans la section données du

- paquet qui suit l'entête ci-dessus. Vaut 1 si ICMP, 17 si UDP, 6 si TCP
- Checksum d'en-tête : 16 bits - code de contrôle d'erreur pour l'entête
- **Adresse source** : 32 bits - adresse IP de la machine source
- **Adresse destination** : 32 bits - adresse IP de la machine destination
- Options : variable
- Bourrage : variable - n'existe que pour assurer à l'en-tête une taille totale multiple de 4 octets. Le bourrage se fait par des octets à zéro.

L'information (données) contenue dans le paquet IP est aussi une structure. Cette dernière est définie de la manière suivante : une entête et une information. Mais elle peut être de plusieurs types : c'est une donnée de protocole de transfert (TPDU : Transfert Protocol Data Unit) du protocole TCP (*Transfert Control Protocol*), du protocole UDP (*User Datagram Protocol*) ou d'un autre protocole de niveau 4.

### Décodage de segment TCP

Un segment TCP est également composé d'un entête et d'un contenu (non représenté ci-dessous).

#### TPDU (segment) du protocole TCP



*Notez qu'une case représente une position bit.*

- **Port source** : 16 bits - peut définir le format du contenu du segment (protocole supérieur)
- **Port Destinataire** : 16 bits - peut définir le format du contenu du segment (protocole supérieur)
- Numéro de séquence : 32 bits - numéro du premier octet de données par rapport au début de la transmission (sauf si SYN est marqué). Si SYN est marqué, le numéro de séquence est le numéro de séquence initial (ISN) et le premier octet à pour numéro ISN+1).
- Accusé de réception: 32 bits - si ACK est marqué ce champ contient le numéro de séquence du prochain octet que le récepteur s'attend à recevoir. Une fois la connexion établie, ce champ est toujours renseigné.
- **Data Offset** : 4 bits - longueur entête en multiples de 32 bits
- Réservé : 6 bits
- Bits de contrôle : 6 bits (de gauche à droite):
  - URG: Pointeur de données urgentes significatif
  - ACK: Accusé de réception significatif
  - PSH: Fonction Push
  - RST: Réinitialisation de la connexion
  - SYN: Synchronisation des numéros de séquence
  - FIN: Fin de transmission
- Fenêtre: 16 bits
- Checksum: 16 bits
- Pointeur de données urgentes: 16 bits
- Options: variable
- Bourrage (padding): variable. Les octets de bourrage terminent l'en-tête TCP de sorte que le nombre d'octet de celle-ci soit toujours multiple de 4 octets (32 bits) et de sorte que l'offset de données marqué dans l'en-tête corresponde bien au début des données applicatives.

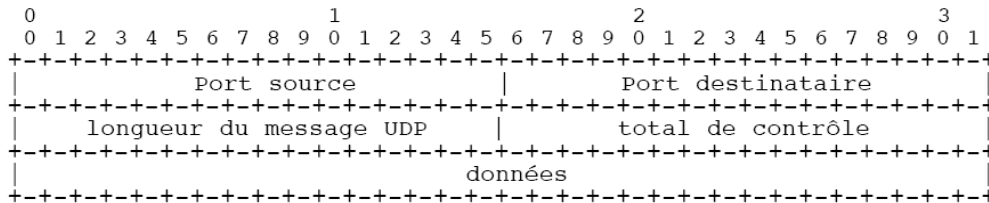
L'information (données) contenue dans le segment TCP peut aussi être une structure avec une entête et des données appartenant à protocole tel que : HTTP, FTP,....

**Autres exemples de NPDU et TPDU**

NPDU ARP (protocole de contrôle) et TPDU UDP (protocole de transport sans garantie)

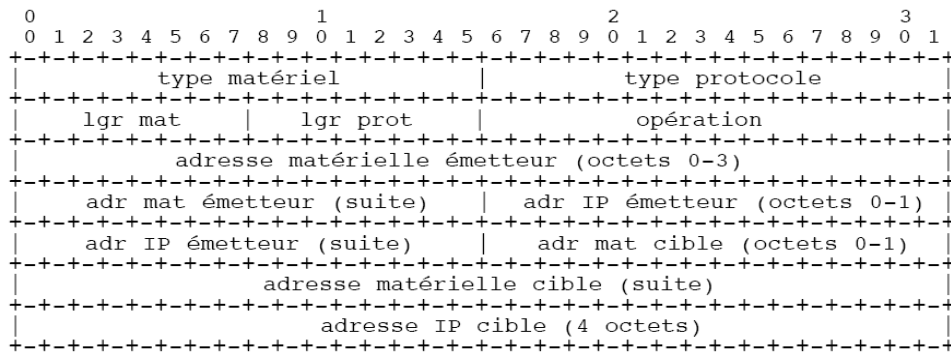
**TPDU (segment) du protocole UDP**

Un segment UDP (*User Datagram Protocol*) est composé d'une entête et d'un contenu qui est l'information à transmettre. On a toujours la même image :



**NPDU (paquet) du protocole ARP**

ARP (*Address Resolution Protocol*) est un protocole de contrôle de niveau 3, il permet de faire la correspondance entre les @IP et les @MAC dans un réseau local.



- type matériel : type du protocole de la couche liaison de données, si 0001 alors Ethernet
- type protocole : type du protocole de la couche réseau, si 0800 alors IP
- lgr mat : longueur des adresses physiques (au niveau liaison de données) en octets
- lgr prot : longueur des adresses de la couche réseau en octets
- opération : 1 demande ARP, 2 réponse ARP, 3 demande RARP, 4 réponse RARP
- adresse matérielle émetteur = @MAC de l'émetteur du paquet ARP
- adresse IP émetteur = @IP de l'émetteur du paquet ARP
- adresse matérielle cible = @MAC du destinataire du paquet ARP
- adresse IP cible = @IP du destinataire du paquet ARP