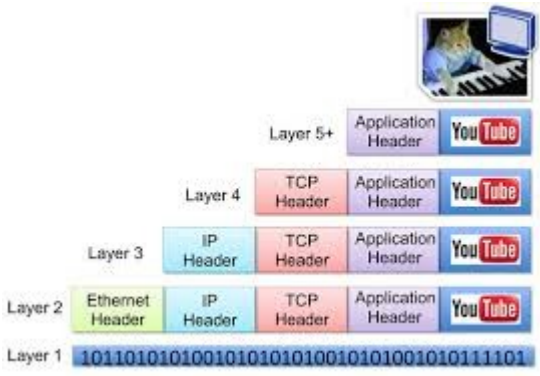
Encapsulation de données

Table des matières

L'encapsulation, en informatique et spécifiquement pour les réseaux informatiques, est un procédé consistant à inclure les données d'un protocole dans un autre protocole.

Lors d'une encapsulation, la couche la plus abstraite est appelée « couche protocole de plus haut niveau » (Upper Layer Protocol - ULP) alors que la couche la plus spécifique est appelée « couche protocole de plus bas niveau » (Lower Layer Protocol - LLP).

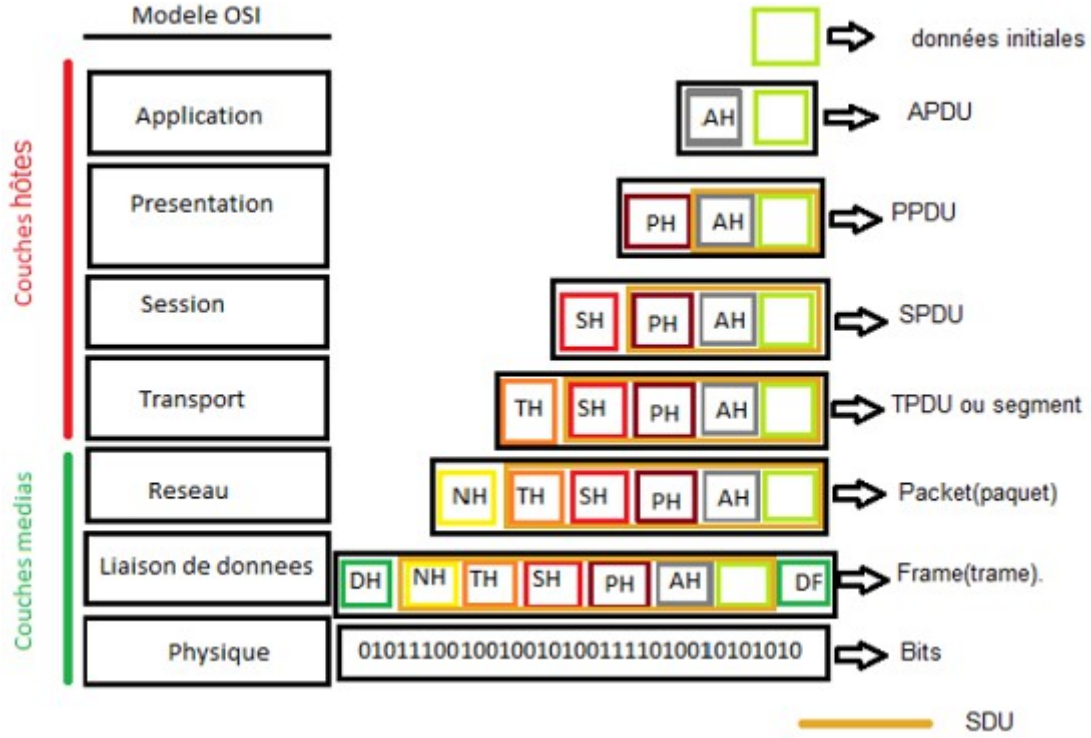


1. Principe

Lors d'une transmission, les données traversent chacune des couches au niveau de la machine émettrice. À chaque couche, une information est ajoutée au paquet de données, il s'agit d'un en-tête, ensemble d'informations qui garanti la transmission. Au niveau de la machine réceptrice, lors du passage dans chaque couche, l'en-tête est lu, puis supprimé. Ainsi à la réception, le message est dans son état originel.

À chaque niveau, le paquet de données change d'aspect, car on lui ajoute un en-tête, ainsi les appellations changent suivant les couches :

- le paquet de données est appelé message au niveau de la couche application
- le message est ensuite encapsulé sous forme de segment dans la couche transport
- le segment une fois encapsulé prend le nom de paquet dans la couche réseau
- enfin on parle de trame au niveau de la couche liaison
- et de signal au niveau de la couche physique



Par exemple, l'Internet est basé sur l'Internet Protocol version 4 et la plupart des applications utilisent aussi bien l'UDP (User Datagram Protocol) que le TCP (Transmission Control Protocol).

Ainsi un fragment de donnée est encapsulé dans un datagramme UDP qui lui-même est encapsulé dans un paquet IP, ce dernier étant alors envoyé via un protocole de la couche de liaison (par exemple Ethernet).

La couche de liaison est responsable de la transmission physique des données ; IP ajoute l'adressage des ordinateurs individuels ; UDP ajoute « l'adressage des applications » (c'est-à-dire le port spécifiant le service comme, par exemple, un service web ou un serveur TFTP).

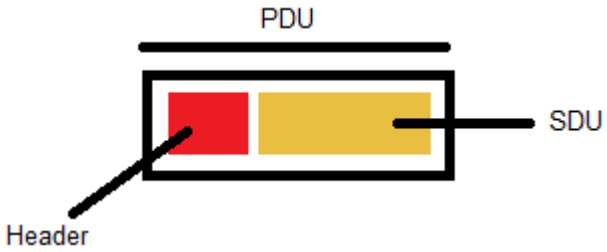
Le modèle OSI et la suite des protocoles Internet utilisent l'encapsulation.

Chaque couche du modèle OSI a une fonction déterminée. Cette corrélation indique bien que certaines informations peuvent se retrouver d'une couche à une autre. Cela n'est possible que grâce au principe d'encapsulation.

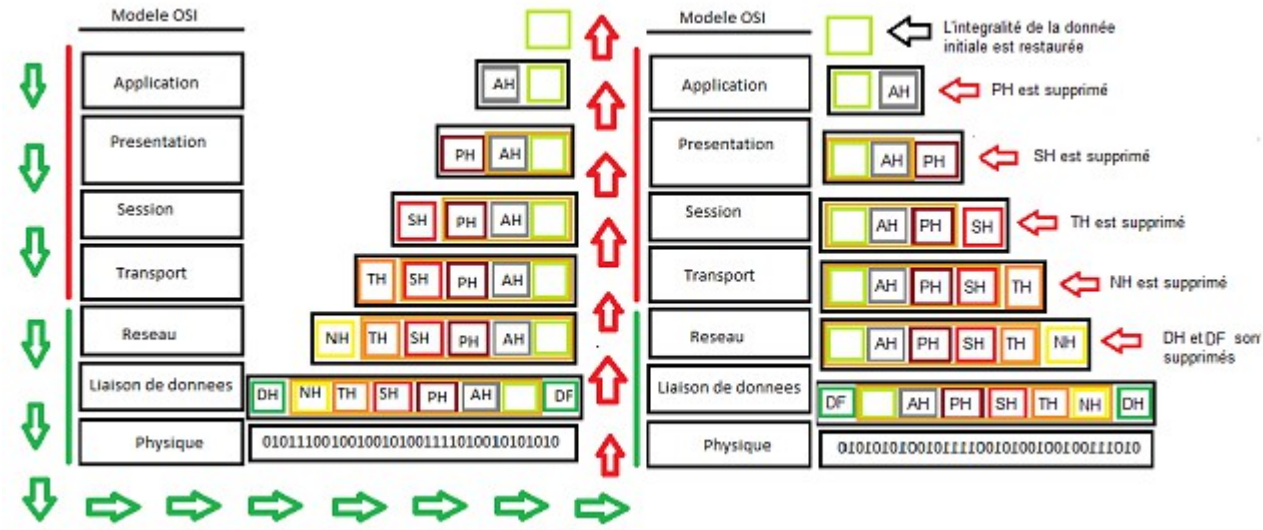
Les données sont enveloppées à chaque couche et portent le nom de PDU (Protocol Data Unit) et contiennent deux choses : la donnée en elle-même et l'en-tête spécifique à cette couche. La partie « donnée » de ce paquet est composée de la donnée initiale, mais aussi des en-têtes des couches qui la précèdent.

Dans une couche N, le PDU est le SDU (Service Data Unit) de la couche N + 1 plus son en-tête (couche N). Ce SDU ne devient un PDU qu'après l'encapsulation. La couche N ajoute des informations dans l'en-tête (header) ou le pied (trailer), voire les deux, du SDU afin de le transformer en un PDU. Ce PDU sera alors le SDU de la couche N - 1. Donc le PDU est un SDU encapsulé avec un en-tête.

Constitution d'un PDU :

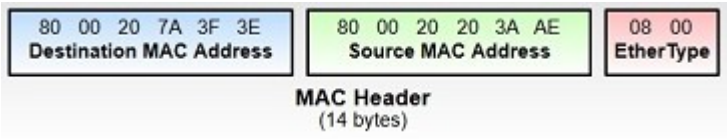


Dans la procédure de réception, chaque en-tête est enlevé lorsque le message « remonte » les couches, tel qu'illustré par le schéma ci-dessous. Cette « suppression » d'en-tête, c'est la décapsulation.



2. Couche liaison de données

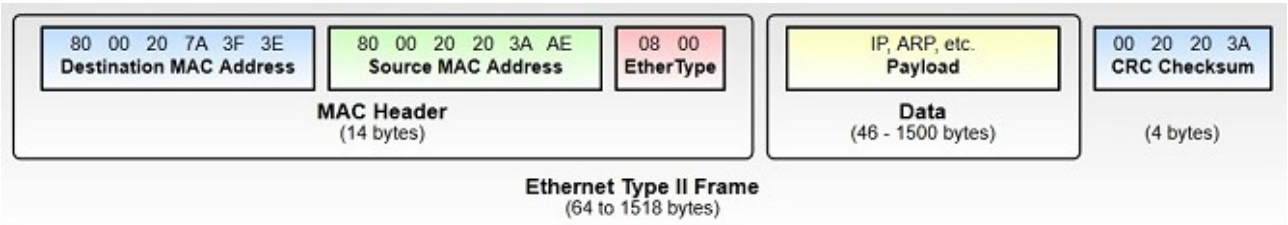
2.1. En-tête Ethernet



Le champ Ether Type peut prendre par exemple les valeurs suivantes :

Type	**utilisation**
0x0800	IPv4, DoD Internet
0x0801	X.75 Internet
0x0802	NBS Internet
0x0803	ECMA Internet
0x0804	ChaosNet
0x0805	X.25 niveau 3
0x0806	ARP
0x0807	XNS
0x86DD	IPv6
0x0806	ARP
0x8035	RARP
0x809B	AppleTalk
0x88CD	SERCOS III
0x0600	XNS
0x8100	VLAN

2.2. Trame Ethernet



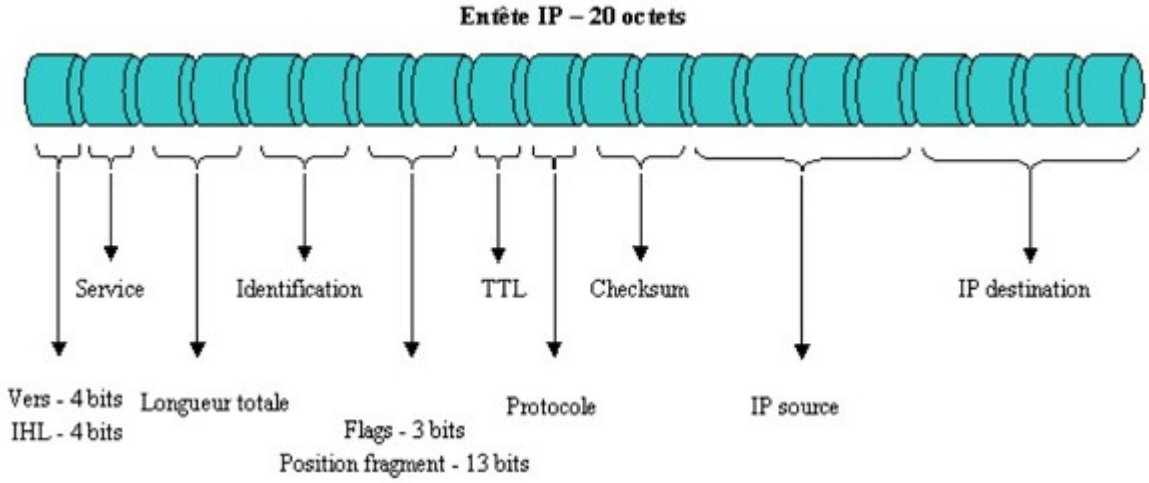
Remarques :

- comme expliqué ci-dessus, si le champ type de protocole possède une valeur hexadécimale inférieure à 0x05DC alors la trame est une trame Ethernet 802.3 et ce champ indique la longueur du champ données ;
- on notera la présence parfois d'un préambule de 64 bits de synchronisation, alternance de 1 et 0 avec les deux derniers bits à 1 (non représenté sur la trame) ;

- l'adresse de broadcast (diffusion) Ethernet a tous ses bits à 1 ;
- la taille minimale des données est de 46 octets (RFC 894 - Frame Format).
- si nécessaire, pour atteindre les 46 octets de données, un bourrage est effectué, et celui-ci est transparent au niveau utilisateur3.

3. Couche réseau

3.1. En-tête IP



3.2. Paquet IP

0 1 2 3	4 5 6 7	8 9 10	11 12 13 14 15	16 17 18	19 20 21 22 23 24 25 26 27 28 29 30 31
Version d'IP	Longueur de l'en-tête	Type de service		Longueur totale	
Identification				Indicateur	Fragment offset
Durée de vie		Protocole		Somme de contrôle de l'en-tête	
Adresse source					
Adresse destination					
Option(s) + remplissage					

Signification des champs :

- Version (4 bits) : Version d'IP utilisée. Ici, 4.
- Longueur de l'en-tête ou IHL (pour Internet Header Length) (4 bits) : Nombre de mots de 32 bits, soit 4 octets (ou nombre de lignes du schéma). La valeur est comprise entre 5 et 15, car il y a 20 octets minimum et on ne peut dépasser 40 octets d'option (soit en tout, 60 octets).
- Type de service ou ToS (pour Type of Service) (8 bits) : Ce champ permet de distinguer différentes qualités de service différenciant la manière dont les paquets sont traités. Composé de 3 bits de priorité (donc 8 niveaux) et trois indicateurs permettant de différencier le débit, le délai ou la fiabilité.

- Longueur totale en octets ou Total Length (16 bits) : Nombre total d'octets du datagramme, en-tête IP comprise. Donc, la valeur maximale est 2^{16} - 1 octets.
- Identification (16 bits) : Numéro permettant d'identifier les fragments d'un même paquet.
- Indicateurs ou Flags (3 bits) :
  - (Premier bit) actuellement inutilisé.
  - (Deuxième bit) DF (Don't Fragment) : lorsque ce bit est positionné à 1, il indique que le paquet ne peut pas être fragmenté. Si le routeur ne peut acheminer ce paquet (taille du paquet supérieure à la MTU), il est alors rejeté.
  - (Troisième bit) MF (More Fragments) : quand ce bit est positionné à 1, on sait que ce paquet est un fragment de données et que d'autres doivent suivre. Quand il est à 0, soit le fragment est le dernier, soit le paquet n'a pas été fragmenté.
- Fragment offset (13 bits) : Position du fragment par rapport au paquet de départ, en nombre de mots de 8 octets.
- Durée de vie ou TTL (pour Time To Live) (8 bits) : Initialisé par l'émetteur, ce champ est décrémenté d'une unité généralement à chaque saut de routeur. Quand TTL = 0, le paquet est abandonné et un message ICMP est envoyé à l'émetteur pour information.
- Protocole (8 bits) : Numéro du protocole au-dessus de la couche réseau : TCP = 6, UDP = 17, ICMP = 1.

  Ce champ permet d'identifier le protocole utilisé par le niveau supérieur :
  - Internet Control Message Protocol ou ICMP est repéré par les bits 00000001, qu'on écrit souvent en hexadécimal avec 01
  - Transmission Control Protocol ou TCP par les bits 00000110, soit 06
  - User Datagram Protocol ou UDP par les bits 00010001, soit 17 en décimal

Code	**Abréviation**	**Nom du protocole**
0	Reserved	
1	ICMP	Internet Control Message
2	IGMP	Internet Group Management
3	GGP	Gateway-to-Gateway
4	IP	IP in IP (encasulation)
5	ST	Stream
6	TCP	Transmission Control
7	UCL	UCL
8	EGP	Exterior Gateway Protocol
9	IGP	any private interior gateway
10	BBN-RCC-MON	BBN RCC Monitoring
11	NVP-II	Network Voice Protocol
12	PUP	PUP

13	ARGUS	ARGUS
14	EMCON	EMCON
15	XNET	Cross Net Debugger
16	CHAOS	Chaos
17	UDP	User Datagram

- Somme de contrôle de l'en-tête ou Header Checksum (16 bits) : Complément à un de la somme complémentée à un de tout le contenu de l'en-tête afin de détecter les erreurs de transfert. Si la somme de contrôle est invalide, le paquet est abandonné sans message d'erreur.
- Adresse source (32 bits) : Adresse IP de l'émetteur sur 32 bits.
- Adresse destination (32 bits) : Adresse IP du récepteur 32 bits.
- Options (0 à 40 octets par mots de 4 octets) : Facultatif.
- Remplissage ou Padding : Champ de taille variable comprise entre 0 et 7 bits. Il permet de combler le champ option afin d'obtenir un en-tête IP multiple de 32 bits. La valeur des bits de bourrage est 0.

3.3. En-tête ARP IPv4

Octet 1	Octet 2	Octet 3	Octet 4
0x0001		0x0800	
0x06	0x04	*Operation*	
Adresse MAC source (octets 1-4)			
Adresse MAC source (octets 5-6)		Adresse IP source (octets 1-2)	
Adresse IP source (octets 3-4)		Adresse MAC destination (octets 1-2)	
Adresse MAC destination (octets 3-6)			
Adresse IP destination (octets 1-4)			

Signification des champs :

- Hardware type (type de matériel) :

  01 - Ethernet (10Mb)

  02 - Experimental Ethernet (3Mb)

- Protocol type (Type de protocole) :

  0x0800 - IP

  Ce champ indique quel est le type de protocole couche 3 (OSI) qui utilise ARP.

- Hardware Address Length (longueur de l'adresse physique) :

  01 - Token Ring

  06 - Ethernet

Ce champ correspond à la longueur de l'adresse physique. La longueur doit être prise en octets.

- Protocol Address Length (longueur de l'adresse logique) :

  04 - IP v4

  16 - IP v6

  Ce champ correspond à la longueur de l'adresse réseau. La longueur doit être prise en octets.

- Operation :

  01 - Request requête

  02 - Reply réponse

  Ce champ permet de connaître la fonction du message et donc son objectif.

- Sender Hardware Address : Adresse MAC source dans le cadre d'Ethernet.
- Sender Internet Address : Adresse IP de source dans le cadre de TCP/IP.
- Target Hardware Address : Adresse MAC destination dans le cadre d'Ethernet. Si c'est une demande ARP, alors, ne connaissant justement pas cette adresse, le champ sera mis à 0.
- Target Internet Address : Adresse IP de destination dans le cadre de TCP/IP

3.4. Paquet ICMP

Bien qu'il soit à un niveau équivalent au protocole IP, un paquet ICMP est néanmoins encapsulé dans un datagramme IP. Dans le cadre de l'IPv4, la forme générale d'un tel paquet est la suivante :

Bit 0 - 7	Bit 8 - 15	Bit 16 - 23	Bit 24 - 31
Version/IHL	Type de service	Longueur totale	
Identification (fragmentation)		*flags* et *offset* (fragmentation)	
Durée de vie(TTL)	Protocole	Somme de contrôle de l'en-tête	
Adresse IP source			
Adresse IP destination			
Type de message	Code	Somme de contrôle	
Bourrage ou données			
Données (*optionnel et de longueur variable*)			

Un tel datagramme est composé :

- d'un en-tête IP (en bleu), avec Protocole valant 1 et Type de Service valant 0.
- du type de message ICMP (8 bits)

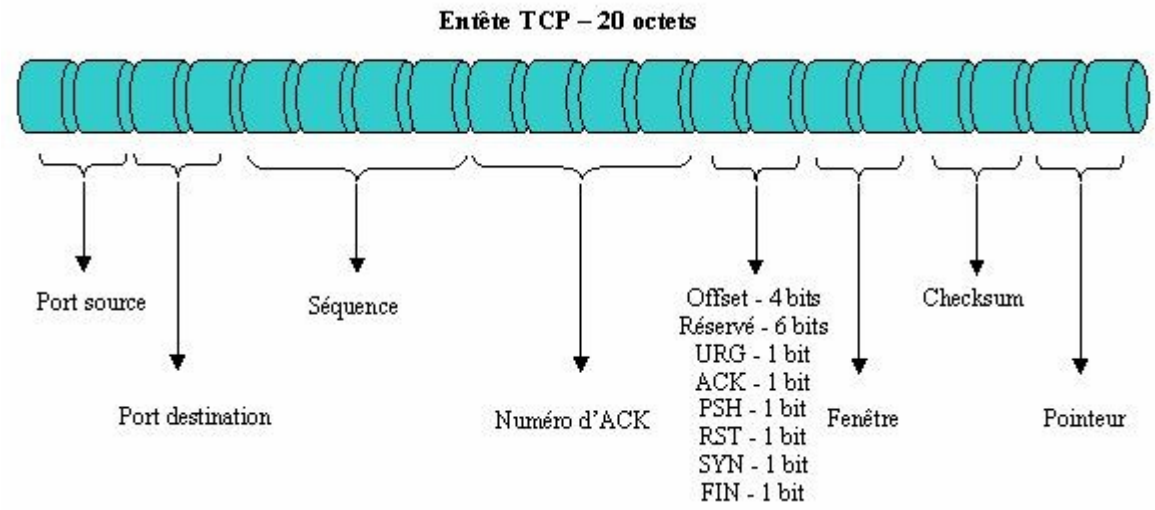
type	**signification du message**

0	Echo Reply (réponse en écho)
3	Destination Unreachable (destination inaccessible)
4	Source Quench (interruption de la source)
5	Redirect (redirection, changement de route)
8	Echo Request (demande d'écho)
11	Time Exceeded for a Datagram (temps de vie d'un datagramme dépassé)
12	Parameter Problem on a Datagram (datagramme mal formé)
13	Timestamp Request (demande de date d'estampillage)
14	Timestamp Reply (réponse à une demande d'estampillage)
15	Information Request (demande d'information)
16	Information Reply (réponse à une demande d'information)
17	Address Mask Request (demande de masque d'adresse)
18	Address Mask Reply (réponse à une demande de masque d'adresse)

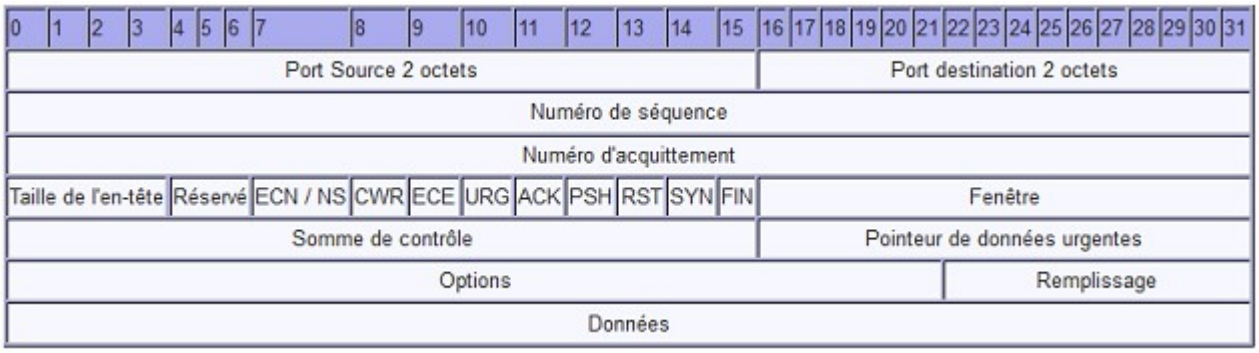
- du code de l'erreur (8 bits)
- d'une somme de contrôle (16 bits), calculée sur la partie spécifique à ICMP (sans l'en-tête IP)
- d'une partie aménagée pour des données relatives aux différents types de réponses (32 bits), si elle n'est pas utilisée, on procède à un bourrage (cette partie peut correspondre aux Identifiant et Numéro de séquence pour un paquet de type Ping par exemple)
- du message

4. Couche transport

4.1. En-tête TCP



4.2. Segment TCP



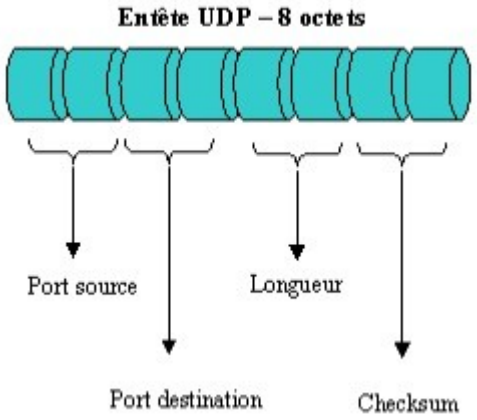
Signification des champs :

- Port source : numéro du port source
- Port destination : numéro du port destination
- Numéro de séquence : numéro de séquence du premier octet de ce segment
- Numéro d'acquittement : numéro de séquence du prochain octet attendu
- Taille de l'en-tête : longueur de l'en-tête en mots de 32 bits (les options font partie de l'en-tête)
- Drapeaux :
  - Réservé : réservé pour un usage futur
  - ECN /NS : signale la présence de congestion, voir RFC 3168 ; ou Nonce Signaling
  - CWR : Congestion Window Reduced : indique qu'un paquet avec ECE a été reçu et que la congestion a été traitée
  - ECE : ECN-Echo : si SYN=1 indique la capacité de gestion ECN, si SYN=0 indique une congestion signalé par IP
  - URG : Signale la présence de données urgentes
  - ACK : signale que le paquet est un accusé de réception (acknowledgement)
  - PSH : données à envoyer tout de suite (push)
  - RST : rupture anormale de la connexion (reset)
  - SYN : demande de synchronisation (SYN) ou établissement de connexion
  - FIN : demande la FIN de la connexion
- Fenêtre : taille de fenêtre demandée, c'est-à-dire le nombre d'octets que le récepteur souhaite recevoir sans accusé de réception
- Somme de contrôle : somme de contrôle calculée sur l'ensemble de l'en-tête TCP et des données, mais aussi sur un pseudo en-tête (extrait de l'en-tête IP)
- Pointeur de données urgentes : position relative des dernières données urgentes
- Options : facultatives
- Remplissage : zéros ajoutés pour aligner les champs suivants du paquet sur 32 bits, si

nécessaire

- Données : séquences d'octets transmis par l'application (par exemple : +OK POP3 server ready...)

4.3. En-tête UDP



Il contient les quatre champs suivants :

- Port Source : indique depuis quel port le paquet a été envoyé.
- Port de Destination : indique à quel port le paquet doit être envoyé.
- Longueur : indique la longueur totale (exprimée en octets) du segment UDP (en-tête et données). La longueur minimale est donc de 8 octets (taille de l'en-tête).
- Somme de contrôle : celle-ci permet de s'assurer de l'intégrité du paquet reçu quand elle est différente de zéro. Elle est calculée sur l'ensemble de l'en-tête UDP et des données, mais aussi sur un pseudo en-tête (extrait de l'en-tête IP)

Note : la présence de ce pseudo en-tête, interaction entre les deux couches IP et UDP, est une des raisons qui font que le modèle TCP/IP ne s'applique pas parfaitement au modèle OSI.

4.4. Segment UDP

Le paquet UDP est encapsulé dans un paquet IP. Il comporte un en-tête suivi des données proprement dites à transporter.

5. Exercices

5.1. Exercice 1 : décoder la trame Ethernet suivante.

```
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....
```

Trame Ethernet Paquet IP Segment TCP

Exercice 2 : Analyse de trame MAC (Ethernet)

Les trames MAC transportent en partie donnée diverses informations qui peuvent elles même être des trames d'un autre protocole (c'est l'encapsulation de protocoles). Étant donné le format de trame de différents protocoles (ici ARP), on veut analyser le contenu d'une trame MAC.

Décoder les trames MAC Ethernet suivantes :

Trame 1

FF FF FF FF FF FF 08 00 20 02 45 9E 08 06 00 01 08 00 06 04 00

01 08 00 20 02 45 9E 81 68 FE 06 00 00 00 00 00 00 81 68 FE 05

Trame 2

08 00 20 02 45 9E 08 00 20 07 0B 94 08 06 00 01 08 00 06 04 00

02 08 00 20 07 0B 94 81 68 FE 05 08 00 20 02 45 9E 81 68 FE 06