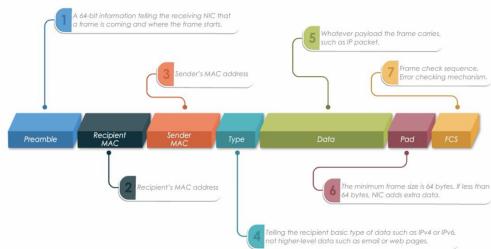


# CARNUS

Enseignement  
supérieur **RODEZ**



## Trame Ethernet

### Cours et exercices

Prof : KAMAL BOUDJELABA

18 octobre 2024

B  
T  
S  
  
C  
I  
E  
L

## Table des matières

<b>1</b>	<b>Format de la trame Ethernet (IEEE 802.3)</b>	<b>1</b>
1.1	EtherType : Numéros de protocole de champ . . . . .	3
1.2	Exemple de protocole 802.1Q . . . . .	4
<b>2</b>	<b>Aperçu sur la trame Ethernet étendue (trame Ethernet II)</b>	<b>5</b>
2.1	Overview : Ethernet frame blocks . . . . .	6
<b>3</b>	<b>Exercices</b>	<b>7</b>
3.1	Exercice 1 . . . . .	7
3.2	Exercice 2 . . . . .	7
3.3	Exercice 3 . . . . .	8
3.4	Exercice 4 . . . . .	9
3.5	Exercice 5 . . . . .	9
<b>A</b>	<b>Annexes</b>	<b>11</b>
A.1	Trame IPv4 . . . . .	11
A.2	Trame IPv6 . . . . .	12
A.3	Trame UDP . . . . .	12
A.4	Trame TCP . . . . .	12
A.5	Trame ARP/RARP . . . . .	13
A.6	Trame ICMP . . . . .	13
A.7	Trame ICMPv6 . . . . .	14
A.8	Charge utile Ethernet . . . . .	14

## Liste des figures

1	Trame Ethernet . . . . .	1
2	Trame Ethernet V2 . . . . .	2
3	Trame Ethernet étendue . . . . .	5

## Liste des tableaux

1	Résumé . . . . .	3
2	EtherType . . . . .	4
3	Ethernet frame blocks . . . . .	6

## 1. Format de la trame Ethernet (IEEE 802.3)

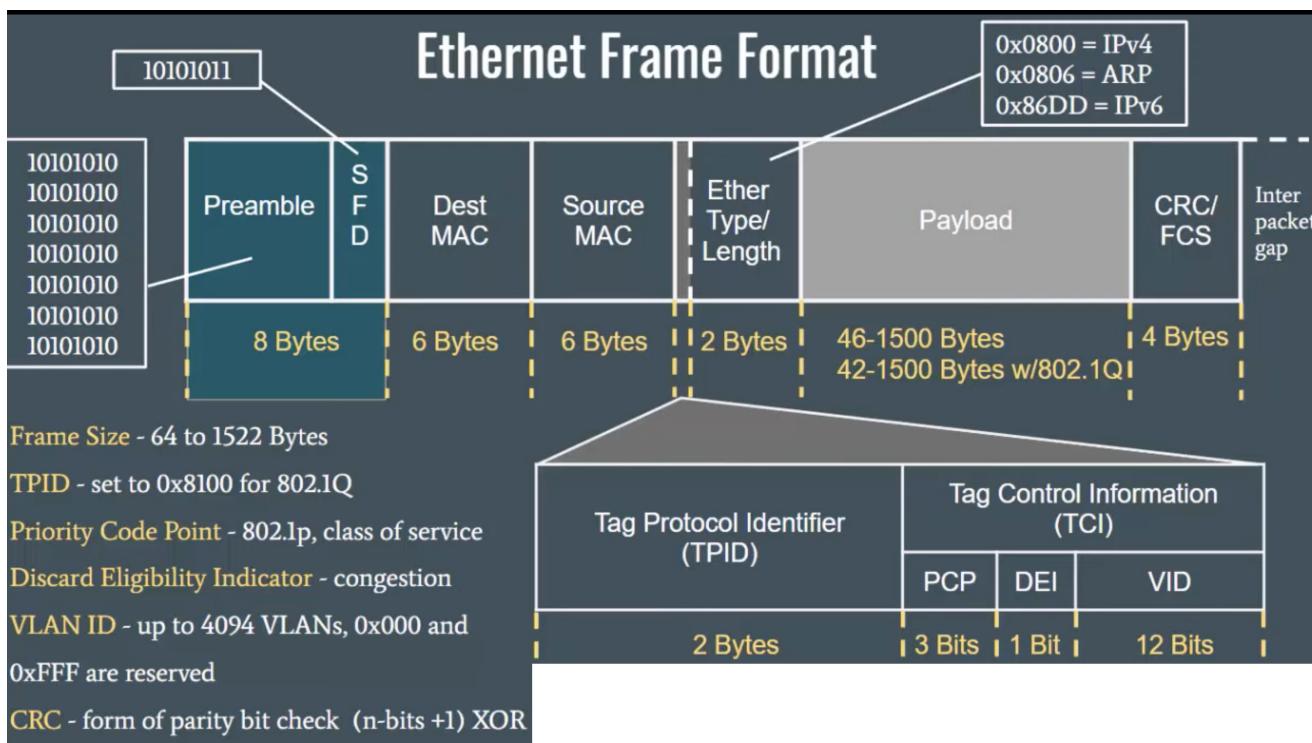
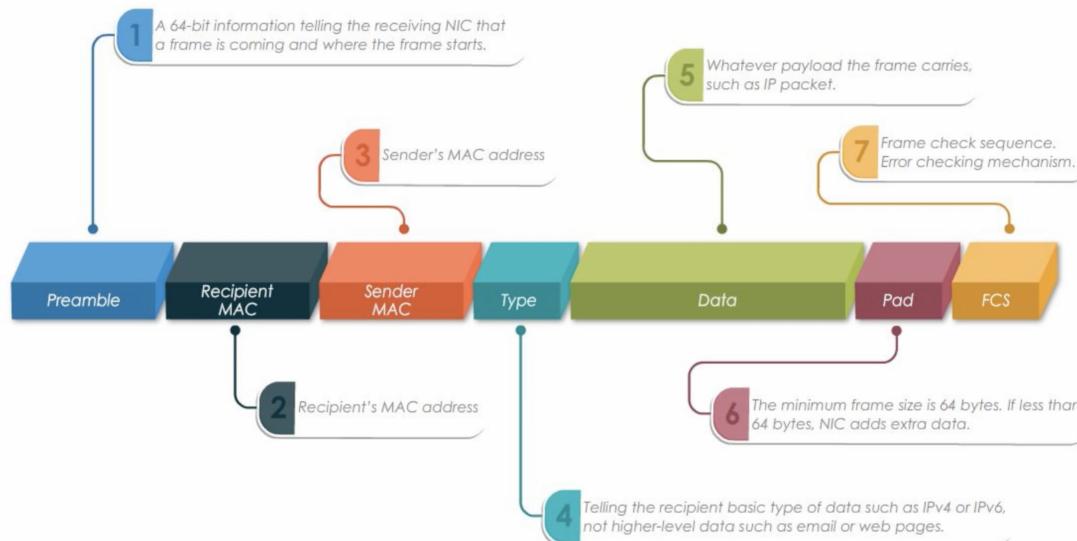
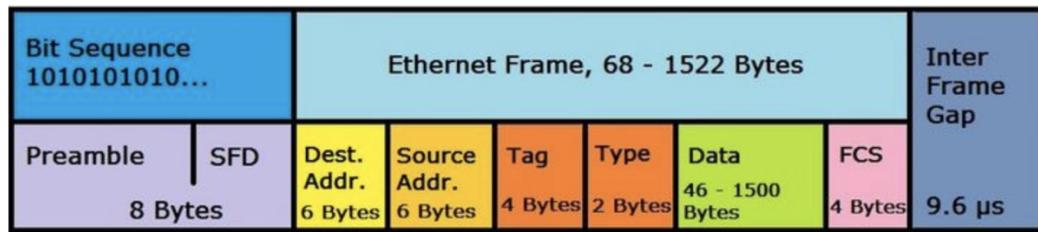


Figure 1. Trame Ethernet

Une trame Ethernet transporte une charge utile de données et est transportée par un paquet Ethernet. La trame peut être IPv4 ou IPv6 et TCP ou UDP. La norme IEEE 802.3 définit la structure de chaque paquet. Lorsqu'un paquet traverse différentes couches, la structure est modifiée en conséquence. Dans la figure suivante, la structure est affichée telle qu'elle traverserait le câble, ou la couche 1. Disséquer la façon dont un paquet est structuré sur le câble permet de comprendre comment le temps système du paquet est affecté et tous les autres composants nécessaires pour envoyer une charge utile. La charge utile se compose de données de protocole pour TCP, UDP ou RTP et IPv4 ou IPv6.

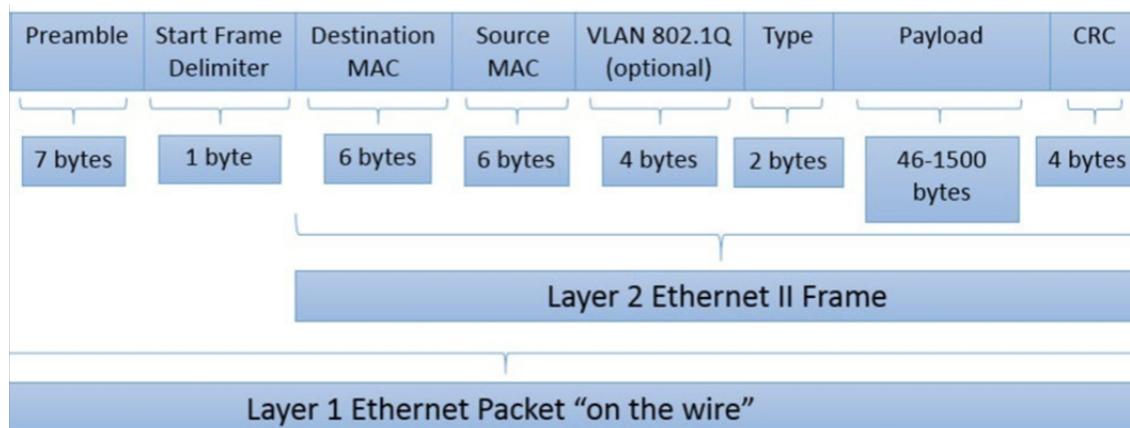


Figure 2. Trame Ethernet V2

- Preamble : Préambule
- SFD : Start Frame Delimiter (délimiteur de début de trame)
- TPID : Tag Protocol IDentifier (Champ d'identifiant de protocole)
- TCI : Tag Control Information (Champ d'information de contrôle)
- PCP : Priority Code Point
- DEI : Discard Eligibility Indicator
- VID : VLAN ID
- CRC : Cyclic Redundancy Check (Contrôle de redondance cyclique)
- FCS : Frame Check Sequence (Séquence de contrôle (vérification) de trame)
- IPG : InterPacket Gap (délai inter-paquet)

**Preamble** – La trame Ethernet commence par un préambule de 7 octets. Il s'agit d'un modèle de 0 et de 1 alternatifs qui indique le début de la trame et permet à l'expéditeur et au destinataire d'établir la synchronisation des bits. Initialement, PRE (Préambule) a été introduit pour permettre la perte de quelques bits en raison des retards de signal. Mais l'Ethernet haut débit d'aujourd'hui n'a pas besoin de préambule pour protéger les bits de trame. PRE (préambule) indique au récepteur que la trame arrive et permet au récepteur de se verrouiller sur le flux de données avant que la trame réelle ne commence.

**Start of frame delimiter (SFD)** – Il s'agit d'un champ de 1 octet qui est toujours défini sur 10101011. SFD indique que les bits suivants commencent la trame, qui est l'adresse de destination. Parfois, SFD est considéré comme faisant partie de PRE, c'est la raison pour laquelle le préambule est décrit sur 8 octets dans de nombreux documents. Le SFD avertit la ou les stations qu'il s'agit de la dernière chance de synchronisation.

**Destination Address** – Il s'agit d'un champ de 6 octets qui contient l'adresse MAC de la machine à laquelle les données sont destinées.

**Source Address** – Il s'agit d'un champ de 6 octets qui contient l'adresse MAC de la machine source. Comme l'adresse source est toujours une adresse individuelle (Unicast), le bit le moins significatif du premier octet est toujours 0.

**Length** – La longueur est un champ de 2 octets, qui indique la longueur de la trame Ethernet entière. Ce champ de 16 bits peut contenir une valeur de longueur comprise entre 0 et 65 534, mais la longueur ne peut pas être supérieure à 1 500 octets en raison de certaines limitations propres à Ethernet.

**Data** – C'est l'endroit où les données réelles sont insérées, également appelée charge utile (Payload). L'en-tête et les données IP seront insérés ici si le protocole Internet est utilisé sur Ethernet. Les données maximales présentes peuvent être aussi longues que 1500 octets. Dans le cas où la longueur des données est inférieure à la longueur minimale, c'est-à-dire 46 octets, des 0 de remplissage sont ajoutés pour respecter la longueur minimale possible.

**Cyclic Redundancy Check (CRC)** – CRC est un champ de 4 octets. Ce champ contient un code de hachage de 32 bits de données, qui est généré sur les champs Adresse de destination, Adresse source, Longueur et Données. Si la somme de contrôle (checksum) calculée par la destination n'est pas la même que la valeur de somme de contrôle (checksum) envoyée, les données reçues sont corrompues.

**Remarque 1.1 :**

La taille de trame d'Ethernet IEEE 802.3 varie de 64 octets à 1518 octets, y compris la longueur des données (46 à 1500 octets).

Bloc	Fonction
Preamble	informe le système récepteur qu'une trame est en train de démarrer et active la synchronisation
SFD (Start Frame Delimiter)	signifie que le champ Adresse MAC de destination commence par l'octet suivant
Destination MAC	identifie le système de réception
Source MAC	identifie le système émetteur
Type	définit le type de protocole (protocole de couche réseau) à l'intérieur de la trame, par exemple IPv4 ou IPv6.
Data and Pad	contient les données utiles (de la couche réseau). Les données de bourrage (Padding data ) sont ajoutées pour répondre à l'exigence de longueur minimale pour ce champ (46 octets)
FCS (Frame Check Sequence)	contient un contrôle de redondance cyclique (CRC) 32 bits qui permet de détecter les données corrompues. La valeur de FCS (Frame Check Sequence) est le résultat d'un calcul polynomial appelé CRC (Cyclic Redundancy Code). A la réception de la trame, la couche liaison effectue le même calcul et compare les deux résultats qui doivent être égaux afin de valider la conformité de la trame reçue. Le champ FCS est le seul champ présent dans la fin Ethernet. Il permet au récepteur de découvrir si des erreurs se sont produites dans la trame. Notez qu'Ethernet ne détecte que la corruption des données en transit - il ne tente pas de récupérer une trame perdue. D'autres protocoles de niveau supérieur (par exemple TCP) effectuent une récupération d'erreur.

Table 1. Résumé

### 1.1 EtherType : Numéros de protocole de champ

Indique quel protocole est encapsulé dans la trame Ethernet

EtherType	Protocol
0x0800	Internet Protocol version 4 (IPv4)
0x0806	Address Resolution Protocol (ARP)
0x8035	Reverse Address Resolution Protocol (RARP)
0x8100	VLAN-tagged frame (IEEE 802.1Q)
0x86DD	Internet Protocol Version 6 (IPv6)
0x8808	Ethernet flow control
0x8809	Link Aggregation Control Protocol
0x8847	MPLS unicast
0x8848	MPLS multicast
0x8863	PPPoE Discovery Stage
0x8864	PPPoE Session Stage
0x888E	EAP over LAN (IEEE 802.1X)
0x889A	HyperSCSI (SCSI over Ethernet)
0x88A2	ATA over Ethernet
0x88CC	Link Layer Discovery Protocol (LLDP)
0x88E5	MAC security (IEEE 802.1AE)
0x8906	Fibre Channel over Ethernet (FCoE)
0x8914	FoE Initialization Protocol
0x9100	VLAN-tagged (IEEE 802.1Q) frame with double tagging

**Table 2.** EtherType

## 1.2 Exemple de protocole 802.1Q

- a – Priorité (PCP) : Ce champ est codé sur 3 bits et représente une information sur la priorité de la trame. Il y a donc 8 niveaux où 000 représente une priorité basse et 111 une haute.
- b – CFI (DEI) : Ce champ est codé sur 1 bit et doit être marqué à 0. CFI (canonical format indicator) est utilisé pour des raisons de compatibilité entre les réseaux Ethernet et les réseaux de type Token ring.
- c – VLAN ID : Ce champ est codé sur 12 bits et représente le numéro du VLAN. Il est donc possible d'intégrer la trame dans 1 VLAN parmi 4096 possibilités. La valeur 0 indique qu'il n'y a pas de VLAN, c'est souvent utilisée dans le cas où l'on désire appliquer une priorité sans avoir besoin de la notion de VLAN.

## 2. Aperçu sur la trame Ethernet étendue (trame Ethernet II)

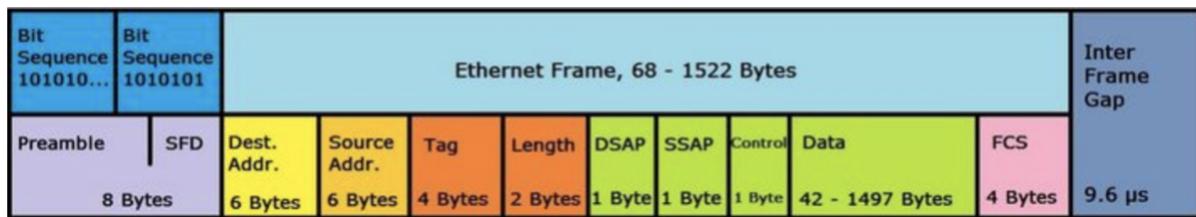


Figure 3. Trame Ethernet étendue

- DA [Destination MAC Address] : 6 bytes (6 octets)
- SA [Source MAC Address] : 6 bytes
- Type [0x8870 (Ethertype)] : 2 bytes
- DSAP [802.2 Destination Service Access Point] : 1 byte
- SSAP [802.2 Source Service Access Point] : 1 byte
- Ctrl [802.2 Control Field] : 1 byte
- Data [Protocol Data] : > 46 bytes
- FCS [Frame Checksum] : 4 bytes

Bien que le champ de longueur soit manquant dans la trame Ethernet II, la longueur de la trame est connue du fait que la trame est acceptée par le interface réseau.

## 2.1 Overview : Ethernet frame blocks

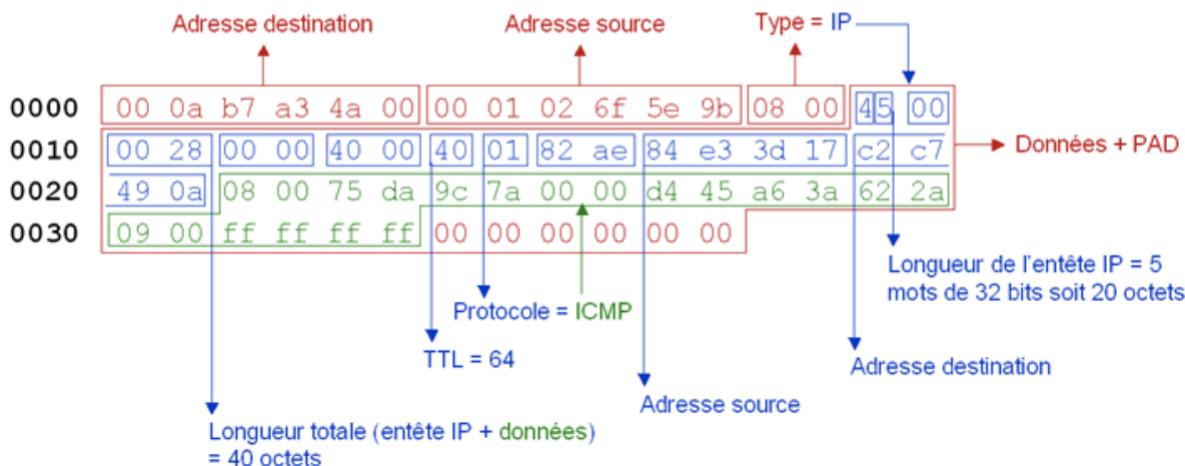
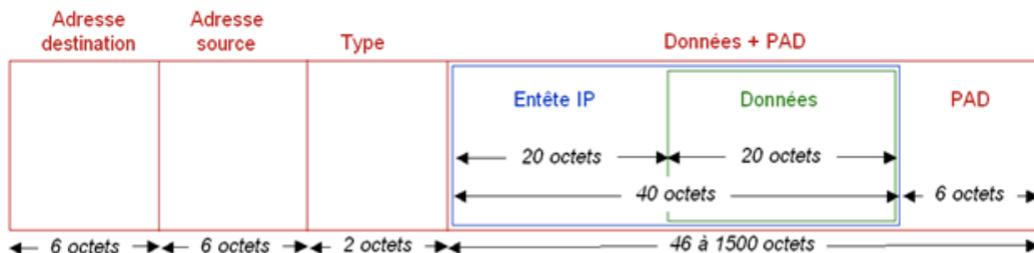
Building block	Size	Function
PreambleStart frame delimiter (SFD)	8 bytes	Synchronization of the receiver's Bit sequence that initiates the frame
Destination address (MAC)	6 bytes	Hardware address of the destination network adapter
Source address (MAC)	6 bytes	Hardware address of the source network adapter
Tag	4 bytes	Optional VLAN tag for integration in VLAN networks (IEEE 802.1q)
Type	2 bytes	Ethernet II : labeling of layer 3 protocols
Length	2 bytes	Length information about the record
Destination service access point (DSAP)	1 byte	Individual address of the addressed service access point
Source service access point (SSAP)	1 byte	Source address of the sending device
Control	1 byte	Defines the LLC frame (logical link)
SNAP	5 bytes	Field for the definition of the organizationally unique identifier (OUI) of the manufacturer and the protocol number (like "Type")
Data	44-1,500 bytes (limit depending on frame structure)	The data to be transmitted
Frame check sequence (FCS)	4 bytes	Checksum that computes the entire frame
Inter frame gap (IFS)	12 bytes	Transmission break of 9.6 $\mu$ s

Table 3. Ethernet frame blocks

### 3. Exercices

#### 3.1 Exercice 1

Un champ sur 2 octets indiquant le type de données encapsulées, et enfin un champ sur au minimum 46 octets contenant les données. Le champ Type valant 0x800, il s'agit de données IP.



Donner les paramètres suivants :

- Adresse IP de la machine ayant initié l'échange
- Adresse MAC de la machine ayant initié l'échange
- Adresse IP de la machine ayant répondu
- Adresse MAC de la machine ayant répondu
- Total Length
- Le protocole encapsulé

#### 3.2 Exercice 2

```
aa aa aa aa aa aa ab 00 40 07 03 04 2b 02 60
8c e8 02 91 08 00 45 00 00 2c 14 ee 00 00 3c 06
85 7a 93 d2 5e 63 93 d2 5e 5c 10 a4 09 e7 42 0c
56 01 00 00 00 00 60 02 40 00 c1 29 00 00 02 04
05 b4 02 80 9a b2 5c 48
```

- Que représentent les 8 octets de début ?
- Donner les adresses MAC du destinataire et de l'émetteur ?
- Donner le protocole encapsulé dans la trame ?
- Que représente les 4 octets de la fin ?

### 3.3 Exercice 3

On a représenté ci-dessous le résultatat d'une capture par Ethereal de trames Ethernet (ni le préambule, ni le FCS ne sont représentés).

**Trame 1 :**

```
0000  00 12 17 41 c2 c7 00 1a 73 24 44 89 08 00 45 00
0010  00 3c 27 30 00 00 80 01 8f d6 c0 a8 01 69 c0 a8
0020  01 01 08 00 4d 56 00 01 00 05 61 62 63 64 65 66
0030  67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76
0040  77 61 62 63 64 65 66 67 68 69
```

**Trame 2 :**

```
0000  00 1a 73 24 44 89 00 12 17 41 c2 c7 08 00 45 00
0010  00 3c 00 29 00 00 96 01 a0 dd c0 a8 01 01 c0 a8
0020  01 69 00 00 55 56 00 01 00 05 61 62 63 64 65 66
0030  67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76
0040  77 61 62 63 64 65 66 67 68 69
```

Pour chaque trame, donner les paramètres suivants :

- MAC source
- MAC destination
- Contenu du champ type de protocole
- Version du protocole
- Longueur de l'entête
- Valeur du champ TOS
- Longueur totale du datagramme IP
- Identifiant affecté au d'anagramme
- Valeur des champs DF MF et fragment offset
- En déduire si d'anagramme est fragmenté
- Valeur du champ TTL
- Contenu du champ protocole
- En déduire le protocole encapsulé dans le paquet IP
- Adresse IP source
- Adresse IP destination
- Message ICMP encapsulé dans le d'anagramme IP :
  - valeur du champ type
  - Valeur du champ code
  - Contenu du champ de données du message ICMP

### 3.4 Exercice 4

On a représenté ci-dessous le résultat d'une capture par le logiciel wireshark d'une trame Ethernet (ni le préambule, ni le FCS ne sont représentés).

#### Trame

0000	00 12 17 41 c2 c7 00 1a 73 24 44 89 08 00 45 00	...A.... s\$D...E.
0010	01 bb da c2 40 00 3c 06 fc 9d d5 e4 00 2a 3e 93	....@.<.....*>.
0020	51 3b 00 50 04 85 87 c7 14 d5 00 12 b0 cb 50 19	Q;.P.....P.
0030	19 20 95 45 00 00 3e 20 0a 3c 74 64 20 77 69 64	. .E..> .<td wid
0040	74 86 3d 22 33 30 25 22 20 20 68 65 69 67 68 74	th="30%" height

- Trame Ethernet :
  - Adresse MAC source
  - Adresse MAC Destination
  - Le contenu du champ type de protocole. En déduire le protocole encapsulé dans la trame
- Paquet IP contenu dans la trame Ethernet :
  - La version du protocole
  - La longueur de l'entête
  - La valeur du champ TOS
  - La longueur totale du datagramme IP
  - L'identifiant affecté au datagramme
  - La valeur des champs DF, MF et fragment offset. En déduire si datagramme est fragmenté
  - La valeur du champ TTL
  - Le contenu du champ protocole. En déduire le protocole encapsulé dans le paquet IP
  - Les adresses IP source et destination
- Message ICMP encapsulé dans le datagramme IP :
  - La valeur du champ type et du champ code. En déduire la nature du message ICMP
  - Le contenu du champ de donnée du message ICMP

### 3.5 Exercice 5

Donner les différents paramètres des trames suivantes :

#### Trame 1

0000	f8 32 e4 9c f3 b3 f0 7f 06 ae 4c c0 08 00 45 00	.2.....L...E.
0010	33 ff 2a 9c 40 00 3d 06 79 61 0a 65 80 18 0a 65	3.*.=@.=ya.e...e
0020	d1 19 00 50 80 fe 78 63 0f 7b 08 17 40 10 80 18	...P..xc.{...@...}
0030	00 fc 99 ed 00 00 01 01 08 0a 06 bf 30 64 7c 85	.....0d .
0040	30 9f 48 54 54 50 2f 31 2e 31 20 32 30 30 20 4f	0.HTTP/1.1 200 0
0050	4b 0d 0a 43 6f 6e 74 65 6e 74 2d 54 79 70 65 3a	K..Content-Type:
0060	20 74 65 78 74 2f 68 74 6d 6c 0d 0a 43 6f 6e 74	text/html..Cont

#### Trame 2

0000	33 33 00 00 00 16 do 23 db 54 8f 2a 86 dd 60 00	33.....#.T.*...`.
0010	00 00 00 38 00 01 00 00 00 00 00 00 00 00 00 00	...8.....
0020	00 00 00 00 00 00 ff 02 00 00 00 00 00 00 00 00 00	.....
0030	00 00 00 00 00 16 3a 00 01 00 05 02 00 00 8f 00	.....:
0040	9c a2 00 00 00 02 04 00 00 00 ff 02 00 00 00 00 00	.....
0050	00 00 00 00 00 02 6f d3 d0 7b 04 00 00 00 ff 02	.....o...{.....
0060	00 00 00 00 00 00 00 01 ff 54 8f 2a	.....T.*

**Trame 3**

0000	ff ff ff ff ff d0 23 db 54 8f 2a 08 06 00 01	.....#.T.*....
0010	08 00 06 04 00 01 d0 23 db 54 8f 2a c0 a8 87 ca	.....#.T.*....
0020	00 00 00 00 00 00 c0 a8 87 fe 00 00 00 00 00 00	.....
0030	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....

**Trame 4**

0000	00 14 6c 39 26 3c c8 9c dc bd b4 1f 08 00 45 00	.19&<.....E.
0010	00 54 00 00 40 00 40 01 a2 da c0 a8 87 19 c0 a8	.T..@.0.....
0020	8f 64 08 00 8a 11 0a 7e 00 01 bc 00 91 4f 00 00	.d.....~.....0..
0030	00 00 54 4c 03 00 00 00 00 00 10 11 12 13 14 15	.TL.....
0040	16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25	..... !%"\$%
0050	26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35	&'()*,-./012345
0060	36 37	

**Trame 5**

0000	33 33 00 01 00 02 cc af 78 1a 2c 53 86 dd 60 00	33.....x.,S..`.
0010	00 00 00 64 11 01 fe 80 00 00 00 00 00 00 85 e8	...d.....
0020	05 2c df 4a 60 d0 ff 02 00 00 00 00 00 00 00 00 00	,,J`.....
0030	00 00 00 01 00 02 02 22 02 23 00 64 68 ae 01 9f	.....".#.dh...
0040	da 5c 00 08 00 02 05 dc 00 01 00 0e 00 01 00 01	.\\.....
0050	15 97 8c 4f b8 70 f4 a3 9c c4 00 03 00 0c 17 cc	...0.p.....
0060	af 78 00 00 00 00 00 00 00 00 27 00 0e 00 0c	.x.....!....
0070	53 6c 61 76 6f 2d 4c 61 70 74 6f 70 00 10 00 0e	Slavo-Laptop....
0080	00 00 01 37 00 08 4d 53 46 54 20 35 2e 30 00 06	...7..MSFT 5.0..
0090	00 08 00 18 00 17 00 11 00 27	.....!

## 1. Annexes

### A.1 Trame IPV4

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	IHL	Type Of Service	ECN	Total Length																											
Identification		0		DF	MF	Fragment Offset																									
Time to live	Protocol	Header Checksum																													
Source IP address		Destination IP address																													
Option + Padding 0 to 40 bytes		DATA																													
...																															

IPV4  
Header

Version : numéro de version du protocole IP

- 04 - IP V4
- 05 - ST Datagram Mode
- 06 - IP V6

IHL (Internet header length) : longueur de l'entête IP

Service : gestion d'une qualité de service traitée (Priorité, Délai, Débit, Fiabilité, ...)

Longueur totale : longueur du paquet incluant l'entête IP et les Data  
 Identification : identification pour reconstituer les différents fragments  
 Flags : état de la fragmentation. Voici le détail des différents bits constituant ce champ.

- DF : Don't Fragment
- MF : More Fragments
- le troisième bit indique si le fragment est le dernier

Position fragment : indique la position du fragment par rapport à la première trame

TTL (Time To Live) : indique la durée de vie maximale du paquet

Protocole : représente le type de Data :

- 1 - 00000001 - 0x01 - ICMP
- 2 - 00000010 - 0x02 - IGMP
- 4 - 00000100 - 0x04 - IP
- 6 - 00000110 - 0x06 - TCP
- 17 - 00010001 - 0x11 - UDP
- 27 - 00011011 - 0x1B - RDP
- 41 - 00101001 - 0x29 - IPV6
- 50 - 00110010 - 0x32 - ESP
- 51 - 00110011 - 0x33 - AH
- 58 - 00111010 - 0x3A - IPV6-ICMP

Checksum : représente la validité du paquet de la couche 3

Adresse IP source : adresse IP source ou de réponse

Adresse IP destination : adresse IP destination

Options : optionnel

Bourrage : permet de combler le champ option

**A.2 Trame IPV6**

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	Trafic Class	Flow Label																																					
Payload Length										Next Header			Hop Limit																										
Source IP address (128 bits)																																							
...																																							
Destination IP address (128 bits)																																							
...																																							

**A.3 Trame UDP**

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																			
Source Port																Destination Port																																		
Length																Checksum																																		
Application Layer Data (0 Octets ou plus)																																																		
...																																																		

**A.4 Trame TCP**

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																							
Source Port																Destination Port																																						
Sequence Number																Acknowledgement Number																																						
Data Offset	Reserved	URG	ACK	PSH	RST	SYN	FIN	Window																Urgent Pointer																														
Checksum																																																						
Options																																																						
...																																																						
Data (Optional)																																																						
...																																																						

Port source : port relatif à l'application sur la machine source.

Port destination : port relatif à l'application sur la machine de destination.

Numéro de séquence : numéro du paquet.

Numéro de l'accusé de réception : acquittement pour les paquets reçus.

Offset : indique donc où les données commencent.

Réservé : pour des besoins futur.

Flags :

- URG : champ Pointeur de donnée urgente est utilisé.
- ACK : numéro de séquence pour les acquittements est valide.
- PSH : indique au récepteur de délivrer les données à l'application
- RST : demande la réinitialisation de la connexion.
- SYN : indique la synchronisation des numéros de séquence.
- FIN : indique fin de transmission.

Fenêtre : nombre d'octets à partir de la position marquée dans l'accusé de réception que le récepteur est capable de recevoir.

Checksum : validité du paquet de la couche 4 TCP.

Pointeur de donnée urgente : position d'une donnée urgente en donnant son décalage par rapport au numéro de séquence.

Options : optionnel.

Bourrage : permet de combler le champ option.

22	ssh	123	ntp
23	telnet	143	imap
25	smtp	161	snmp
53	dns	162	snmptrap
67	dhcp Server/Relay	443	https
68	dhcp client	546	Dhcp-v6 Client
69	tftp	547	Dhcp-v6 Server/Relay
80	http		

### A.5 Trame ARP/RARP

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																									
Hardware Type																Protocol Type																																								
Hardware Add Len								Protocol Add Len								Reserved (Opcode)																																								
Sender Hardware Address																																																								
Sender Hardware Address																Sender Protocol Address																																								
Sender Protocol Address																Target Hardware Address																																								
Target Hardware Address																																																								
Target Protocol Address																																																								

Opération :

- 1 ARP request.
- 2 ARP response.
- 3 RARP request.
- 4 RARP response.

### A.6 Trame ICMP

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	
Type																Checksum																
Identifier																Sequence Number																
Address Mask																																

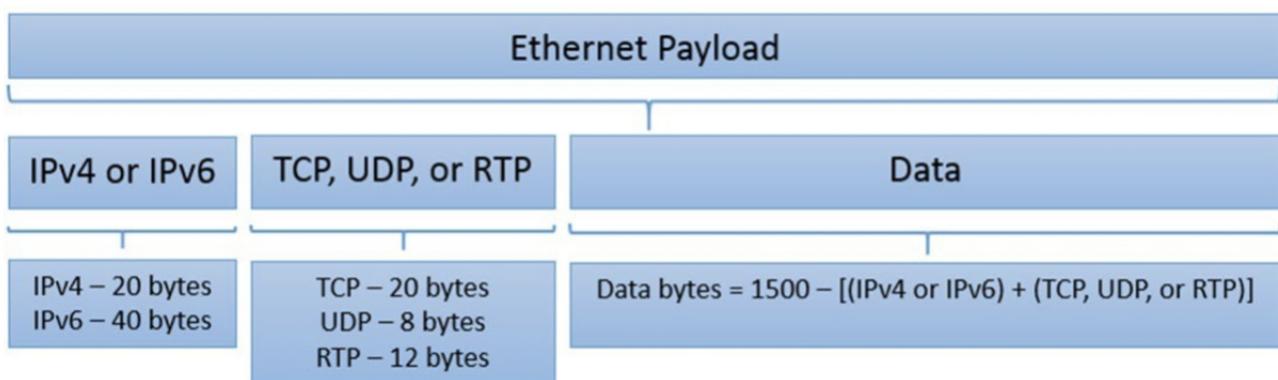
Type	Code	Description
0		Echo reply
3		Destination unreachable
3	0	Net unreachable
3	1	Host unreachable
3	2	Protocol unreachable
3	3	Port unreachable
5		Redirect
5	0	Redirect datagrams for the network
5	1	Redirect datagrams for the host
5	2	Redirect datagrams for the type of service and network
5	3	Redirect datagrams for the type of service and host
8		Echo. (request)
11		Time exceeded
11	0	Time to live exceeded in transit
135	0	Neighbor Solicitation
135	0	Neighbor advertisement

### A.7 Trame ICMPv6

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

### A.8 Charge utile Ethernet

La charge utile Ethernet varie en fonction du type de données qu'elle transporte. Il s'agit d'une combinaison d'en-têtes TCP, UDP ou RTP combinés à un en-tête IPv4 ou IPv6, et surtout de la charge utile réelle qui contient les données envoyées. Les champs de la charge utile sont affichés dans la figure suivante :



Comme le montre la figure, la quantité de données réelles envoyées dans une trame Ethernet dépend du nombre d'octets consommés par les autres champs. D'autres options sont disponibles qui ne sont pas listées ici. Par exemple, les hôtes Linux ajoutent automatiquement un horodatage à la pile TCP, ajoutant 12 octets.