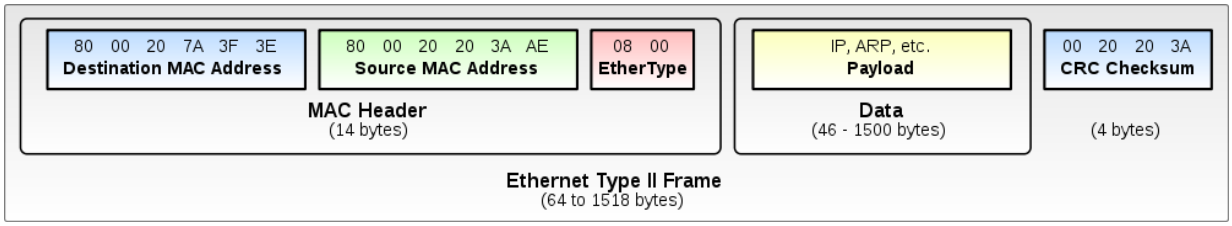
TD n°2 *Encapsulation et Structure des paquets*

Exercice 1. Encapsulation

Un *segment* UDP est encapsulé dans un *datagramme* IPv4 qui lui même est encapsulé dans une *trame* Ethernet :



IPv4 Header Format

Offsets Octet		0								1								2								3							
Octet	Bit	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
0	0	Version				IHL				DSCP						ECN		Total Length															
4	32	Identification																Flags			Fragment Offset											
8	64	Time To Live								Protocol								Header Checksum															
12	96	Source IP Address																															
16	128	Destination IP Address																															
20	160	Options (if IHL > 5)																															

UDP Header

Offsets Octet		0								1								2								3							
Octet	Bit	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
0	0	Source port																Destination port															
4	32	Length																Checksum															

IPv4

- Version : version du protocole IP
- IHL (Internet Header Length) : nombre de mots de 32 bits dans l'entête IP
- DSCP (Differentiated Service Code Point)
- ECN (Explicit Congestion Notification)
- Total Length : nombre d'octets dans le paquet (headers+data)
- Identification : utilisé quand il y a fragmentation
- Flags : champ de 3 bits (bit 0 doit être 0, bit 1 "don't fragment", bit 2 "more fragments")
- Fragment Offset : utilisé quand il y a fragmentation
- Time To Live : décrémenté à chaque passage au travers d'un routeur. quand il atteint 0, le paquet est abandonné.

- Protocol : protocole utilisé dans la portion data du datagramme (6=TCP, 17=UDP)
- Header Checksum : complément à 1 de la somme des mots de 16 bits dans le header (pour calculer cette somme on considère que le champ checksum est à 0)

UDP
- Length : longueur en octets du segment (header+data)
- Checksum : pour header+data

Exercice 2. Décodage d'une interaction client/server.

2.1 Décodez la trame contenant la requête du client (formattée avec `hexdump -C`) :

```
00000000  00 00 00 00 00 00 00 00  00 00 00 00 08 00 45 00  |..............E.|
00000010  00 20 ab 84 40 00 40 11  91 46 7f 00 00 01 7f 00  |. ..@.@..F......|
00000020  00 01 9e 6b 15 b3 00 0c  fe 1f 64 61 74 65        |...k......date|
0000002e
```

Réponse:

2.2 Décodez la trame contenant la réponse du serveur (formattée avec `hexdump -C`) :

```
00000000  00 00 00 00 00 00 00 00  00 00 00 00 08 00 45 00  |..............E.|
00000010  00 37 ab 85 40 00 40 11  91 2e 7f 00 00 01 7f 00  |.7..@.@.........|
00000020  00 01 15 b3 9e 6b 00 23  fe 36 32 30 31 34 2d 30  |.....k.#.62014-0|
00000030  34 2d 30 37 20 31 37 3a  33 38 3a 32 39 2e 31 36  |4-07 17:38:29.16|
00000040  37 39 35 36 0a                                    |7956.|
00000045
```

Réponse: