Université Paris-Est Créteil
Faculté de Droit - IUT de Sénart-Fontainebleau
2018/2019
Deuxième semestre – Première année
Master Droit du Numérique
Réseau et normes

PREMIÈRE SESSION – Jeudi 2 mai 2019
Corrigé

- 1^o) Cette trame de 25 x16 + 7 = 407 octets utiles provient d'une interface Ethernet donc commence par un en-tête Ethernet (sans préambule ni CRC).

Analyse de l'en-tête Ethernet.-

- Les six premiers octets donnent l'adresse MAC du destinataire : `00:1b:21:be:d1:f6`.
- Les six octets suivants donnent l'adresse MAC de l'expéditeur : `34:27:92:60:95:ca`.
- les deux octets suivants, 08 00h, soit 2 048 donne le type Ethernet, donc IP (certainement IPv4).

Analyse de l'en-tête IPv4.- Puisque le type Ethernet est IP, cet en-tête est suivi d'un en-tête IP, certainement IPv4 :

- Le premier demi-octet, 4, donne la version d'IP. Il s'agit bien d'IPv4.
- Le second demi-octet, 5, donne la longueur de l'en-tête IP en mots de 4 octets. On a donc 20 octets, soit le minimum pour un en-tête IPv4. Il n'y a pas d'options.
- L'octet suivant donne le type de service. Il est nul, ce qui est habituel pour une extrémité.
- Les deux octets suivants, `0x01 89`, donnent la longueur du paquet, soit 393 octets. Avec les 14 octets de l'ent-tête Ethernet, on retrouve bien les 407 octets de la trame ; il n'y a donc rien d'autre que le paquet IPv4 comme données de la trame.
- Les deux octets suivants, `0x00 00`, donnent l'identificateur du paquet, sur lequel il n'y a rien de plus à dire puisqu'il est choisi presqu'au hasard.

- Les deux octets suivants, `0x40 00`, représentent deux champs :
  - Les trois premiers bits `010b` sont des drapeaux : le premier est « réservé » donc sa valeur nulle n'est pas étonnante ; le second, DF, qui est levé, indique que le paquet originel ne doit pas être fragmenté ; le dernier, MF, qui n'est pas levé, indique qu'il n'y a pas de fragment sui suit, ce qui est normal puisque le paquet originel ne devait pas être fragmenté.
  - Les treize derniers bits donnent le décalage par rapport au paquet originel. Il est nul, ce qui est normal puisque le paquet originel ne devait pas être fragmenté. Le paquet originel n'a donc effectivement pas été fragmenté.
- L'octet suivant `0x39`, donne la durée de vie, c'est-à dire le nombre de sauts que peut traverser le paquet avant d'être écarté : la valeur 57 montre qu'il a traversé quelques sauts avant de nous parvenir.

  Le plus probable est que le TTL était à l'origine à 64 et qu'il est passé par 7 sauts.
- L'octet suivant, `06h`, donne le « protocole » : il s'agit ici de TCP.
- Les deux octets suivants donnent la somme de contrôle, `0x2d 77`. Nous y reviendrons à la question 2.
- Les quatre octets suivants, `0xc1 30 8f f4`, donnent l'adresse IP de l'expéditeur : 193.48.143.244.
- Les quatre octets suivants, `0xc0 a8 01 2b`, donnent l'adresse IP du destinataire : 192.168.1.43. Il s'agit d'une adresse IP privée, certainement attribuée dynamiquement par la box.

Analyse de l'en-tête TCP.- Puisque le protocole IP est TCP, l'en-tête IPv4 est suivi d'un en-tête TCP :

- Les deux premiers octets, `0x00 50`, donnent le port de l'expéditeur. Il s'agit d'un port bien connu : 80 pour un service HTTP.
- Les deux octets suivants, `0xc8 9a`, donnent le port du destinataire. Il est supérieur à 2 048, il s'agit donc certainement d'un port client attribué de façon presqu'aléatoire par le module TCP du client.
- Les 4 octets suivants, `0x99 1b f5 78`, donnent le numéro de séquence.
- Les 4 octets suivants, `0x8a 17 26 4d`, donnent le numéro d'accusé de réception.
- Les deux octets suivants, `0x50 18`, représentent deux champs :
  - Le demi-octet suivant, `0x5`, donne la longeur de l'en-tête TCP en mots de 4 octets. On a donc 20 octets, ce qui est la taille minimum. Il n'y a donc pas d'options.

  - Les 12 bits suivants, `0000 0001 1000b`, sont des drapeaux : les six premiers sont à l'origine réservés, donc nuls ; le drapeau urgent URG n'est pas levé ; le drapeau d'accusé de réception ACK est levé, ce qui indique qu'il faut prendre en compt le numéro d'accusé de réception ; le drapeau PuSH est levé, c'est donc la fin du message, qu'il faudra envoyer à l'application ; les drapeaux ReSeT, SYn et FIN ne sont pas levés.

- Les deux octets suivants, `0x02 01`, donnent la taille de la fenêtre : 513, donc le débit n'est pas élevé.
- Les deux octets suivants, `0x54 d7`, donnent la somme de contrôle TCP, dont nous ne occuperons pas ici.
- Les deux derniers octets, `0x00 00`, donnent la valeur du pointeur urgent, qui est nul, ce qui est normal puisque le drapeau urgent n'est pas levé.

Analyse du message HTTP.- Nous sommes arrivés à la fin des 20 octets de l'entête TCP, la suite est donc constituée du message HTTP.

En s'aidant de la fenêtre ASCII de l'éditeur hexadécimal, on trouve le message HTTP suivant :

```
HTTP/1.1 200 OK\r\n
Date:  Wed, 10 Apr 2019 12:11:52 GMT\r\n
Server:  Apache/2.4.29 (Ubuntu)\r\n
Last-modified:  Wed, 10 Apr 2019 12:09:12 GMT\r\n
Etag:  "65-5862bf0030a00"\r\n
Accept-Ranges:  bytes\r\n
Content-Length:  101\r\n
Vary:  Accept-Encoding\r\n
Content-Type:  text/plain\r\n
\r\n
Congratulations\r\n
You have reached the end of the first question.\r\n
It is time to compute the checksum.\r\n
```

Il s'agit d'une réponse.

- 2^o) Calculons la somme de contrôle de l'en-tête IP :

```
45 00 :  0100 0101 0000 0000
01 89 :  0000 0001 1000 1001
         ---------------------------------------
         0100 0110 1000 1001
40 00 :  0100 0000 0000 0000
         ---------------------------------------
         1000 0110 1000 1001
39 06 :  0011 1001 0000 0110
         ---------------------------------------
         1011 1111 1000 1111
c1 30 :  1100 0001 0011 0000
         ---------------------------------------
         1000 0000 1011 1111
```

```
       + 0000 0000 0000 0001
         -------------------------------------
         1000 0000 1100 0000
8f f4 :  1000 1111 1111 0100
         -------------------------------------
         0001 0000 1011 0100
       + 0000 0000 0000 0001
         -------------------------------------
         0001 0000 1011 0101
c0 a8 :  1100 0000 1010 1000
         -------------------------------------
         1101 0001 0101 1101
01 2b :  0000 0001 0010 1011
         -------------------------------------
         1101 0010 1000 1000
```

Le complément à 1 est `0010 1101 0111 0111b = 0x2d 77`. On retrouve bien la somme de contrôle.