

## TD 7 - De IPv4 à IPv6

### Exercice 1

Dans le protocole IPv6, l'en-tête des datagrammes a été simplifiée. Elle passe de 14 champs à 8 champs. Le champ "next header" ou "en-tête suivant" identifie le type de paquet suivant l'en-tête IP.

1. s'il n'y a pas d'en-têtes d'extension, que contient le champ "next header"? A quel champ correspond t-il dans IPv4?
2. S'il y a des en-têtes d'extension, que devient ce champ?

### Exercice 2

Quelle différence y a-t-il entre les techniques de fragmentation d'IPv4 et d'IPv6?

### Exercice 3

L'en-tête des paquets IPv6 contient un champ "next header" pour pouvoir rajouter des en-têtes supplémentaires appelés aussi des en-têtes d'extension ou des en-tête d'options. Ils sont utilisés par IPng pour réaliser des fonctions non impératives dans le service de base.

1. Est-ce que l'en-tête de IPv6 a été simplifiée en terme de nombre d'octets, de nombre de champs et de délai de traitement?
2. La liste des options est donnée dans l'ordre suivant :
  - en-tête IPv6
  - en-tête de l'option noeud par noeud (next header=0)
  - en-tête de routage (next header=43)
  - en-tête fragmentation (next header=44)
  - en-tête authentification (next header=51)
  - en-tête de confidentialité (Encapsulation Security Payload - ESP) (next header=50)
  - en-tête de l'option de bout-en-bout (next header=60)
  - (a) Rappeler le fonctionnement de l'option noeud par noeud, routage et fragmentation.
  - (b) quelle serait la taille maximale d'un fragment (en octets) si l'on utilise l'option fragmentation? Déduire la taille maximale d'un datagramme.

### Exercice 4

Soit les adresses IPv6 suivantes :

- 1080 :0 :0 :0 :8 :800 :200C :417A
- FF01 :0 :0 :0 :0 :0 :0 :43
- 0 :0 :0 :0 :0 :0 :0 :1
- 0 :0 :0 :0 :0 :0 :0 :0
- 0 :0 :0 :0 :0 :0 :13.1.68.3
- 0 :0 :0 :0 :0 :FFFF :129.144.52.38

1. Identifier le type de chacune des adresses.
2. Donner la forme compressée de chacune.

## Exercice 5

Soient les traces suivantes :

**Trace 1** : le paquet suivant a été capturé lors de l'ouverture d'une connexion telnet :  
`telnet @router1@destination` permet un routage vers la destination en passant par `routeur1`.

Commenter la trace.

```
60 00 00 00 00 48 2b 40 5f 06 b5 00 c0 6c 77 00
00 00 00 00 c0 15 cb a0 5f 06 b5 00 c0 6c 77 00
00 00 1a 00 20 0c 56 62 06 02 00 01 00 00 00 00
5f 06 b5 00 c0 6c 77 00 00 00 00 00 c0 75 d6 af
04 04 00 17 44 4e f4 d3 00 00 00 00 c0 02 40 00
90 61 00 00 02 04 05 a0 01 03 03 00 01 01 08 0a
00 0a 56 95 00 00 00 00 01 01 0c 06 00 00 00 0b
```

**Trace 2** : les paquets suivants correspondent à l'envoi d'un datagramme de longueur 3500 octets en UDP alors que la MTU de l'interface est de 1500.

Commenter ces deux paquets.

```
60 00 00 00 05 a0 2c 40 5f 06 b5 00 c0 2c 44 00
00 01 00 00 f8 30 4f 80 5f 06 b5 00 c0 2c 44 00
00 02 1a 00 20 0c 7a 34 11 00 00 01 00 00 05 36
04 d2 13 89 0d ...
```

```
60 00 00 00 05 a0 2c 40 5f 06 b5 00 c0 2c 44 00
00 01 00 00 f8 30 4f 80 5f 06 b5 00 c0 2c 44 00
00 02 1a 00 20 0c 7a 34 11 00 05 99 00 00 05 36
7e 20 21 22 23 ...
```