N° Anonymat :

Examen LU3INI033 « Réseaux »

Mardi 19 décembre 2023 – Durée : 2 heures

Autorisé : 1 feuille A4 manuscrite recto/verso + calculatrice (non communicante)

Voici 3 feuilles contenant les énoncés et les zones de réponse à compléter (sans déborder), sans les dégrafer.
Vous devez écrire votre numéro d'anonymat sur chacune des feuilles.

Exercice 1 : Codage/décodage (40 minutes)

Les **trames données en annexe** ont été extraites d'une trace Wireshark obtenue juste après le démarrage du poste client de Bob, connecté à un réseau privé domestique constitué d'une « box » et de 4 équipements terminaux (PC et tablettes). Pour décoder ces trames, **vous trouverez en annexe le format des trames, paquets et messages** nécessaires.

1. Complétez le tableau suivant (en indiquant « *inconnue* » si l'adresse ne peut être déterminée).

Machine	**Adresse MAC** (hexadécimal double pointé)	**Adresse IP** (décimal pointé)
Poste client		
Passerelle du réseau privé		
Serveur NAT du réseau privé		
Serveur DHCP du réseau privé		
Serveur DNS local		
Serveur DNS d'autorité primaire		
Serveur applicatif contacté		

2. Qu'a cherché à faire Bob pour déclencher l'envoi de ces trames (cocher la ou les cases correctes) ?

   - ☐ Consulter sa messagerie électronique
   - ☐ Consulter une page Web
   - ☐ Se connecter à un poste de travail distant
   - ☐ Jouer en ligne
   - ☐ Regarder une vidéo
   - ☐ Télécharger un fichier exécutable

   Argumentez votre réponse.

3. Quel est le nom du serveur applicatif sur lequel Bob est allé ? Possède-t-il des alias ?

4. Un échange ARP ayant forcément eu lieu avant l'envoi des trames données en annexe, donnez le codage hexadécimal de :

   a) la trame encapsulant la requête ARP ;

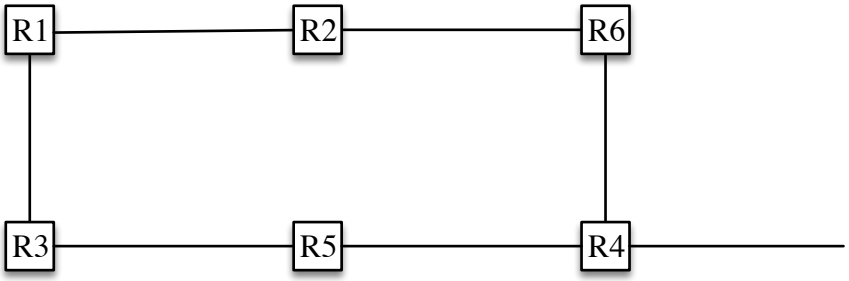
   b) la trame encapsulant la réponse ARP.

5. Donnez le nombre total d'octets de données TCP envoyés par le client au serveur.

N° Anonymat :

Exercice 2 : Routage (40 minutes)

On considère le réseau 11.11.0.0/16 constitué de 7 sous-réseaux interconnectés par 6 routeurs R1 à R6 représenté ci-contre.



Tous les routeurs utilisent le protocole de routage RIP. Les tables de routage (incomplètes) sont données ci-dessous.

R1

Dest	Mask	Gateway	Int	Dst
11.11.0.0	255.255.224.0	*	eth1	1
11.11.32.0	255.255.224.0	*	eth0	1
11.11.64.0	255.255.224.0			
11.11.96.0	255.255.224.0			
11.11.128.0	255.255.224.0			
11.11.160.0	255.255.224.0			
11.11.192.0	255.255.224.0			

R2

Dest	Mask	Gateway	Int	Dst
11.11.0.0	255.255.224.0	*	eth0	1
11.11.32.0	255.255.224.0			
11.11.64.0	255.255.224.0	*	eth1	1
11.11.96.0	255.255.224.0			
11.11.128.0	255.255.224.0			
11.11.160.0	255.255.224.0			
11.11.192.0	255.255.224.0			

R3

Dest	Mask	Gateway	Int	Dst
11.11.0.0	255.255.224.0			
11.11.32.0	255.255.224.0	*	eth0	1
11.11.64.0	255.255.224.0			
11.11.96.0	255.255.224.0	*	eth1	1
11.11.128.0	255.255.224.0			
11.11.160.0	255.255.224.0			
11.11.192.0	255.255.224.0			

R4

Dest.	Mask	Gateway	Int	Dst
11.11.0.0	255.255.224.0			
11.11.32.0	255.255.224.0			
11.11.64.0	255.255.224.0			
11.11.96.0	255.255.224.0			
11.11.128.0	255.255.224.0	*	eth2	1
11.11.160.0	255.255.224.0	*	eth0	1
11.11.192.0	255.255.224.0	*	eth1	1

R5

Dest	Mask	Gateway	Int	Dst
11.11.0.0	255.255.224.0			
11.11.32.0	255.255.224.0			
11.11.64.0	255.255.224.0			
11.11.96.0	255.255.224.0	*	eth0	1
11.11.128.0	255.255.224.0			
11.11.160.0	255.255.224.0	*	eth1	1
11.11.192.0	255.255.224.0			

R6

Dest	Mask	Gateway	Int	Dst
11.11.0.0	255.255.224.0			
11.11.32.0	255.255.224.0			
11.11.64.0	255.255.224.0	*	eth0	1
11.11.96.0	255.255.224.0			
11.11.128.0	255.255.224.0			
11.11.160.0	255.255.224.0			
11.11.192.0	255.255.224.0	*	eth1	1

On supposera que l'adresse IP du routeur Ri sur le sous-réseau 11.11.x.0/19 est : 11.11.x.i

Si deux routes ont le même coût, on choisira celle passant par le routeur de plus petit numéro.

1. Complétez le schéma en indiquant l'adresse IP des 7 sous-réseaux (slashé), ainsi que le nom des interfaces des 6 routeurs (eth0, eth1 et eth2).
2. Complétez les tables de routage (la colonne gateway devant contenir des adresses IP).

On suppose que R4 est la passerelle de sortie du réseau 11.11.0.0/16, le reste de l'internet étant accessible via son interface eth2. Il manque dans chacune des tables précédentes, une ligne indiquant la route par défaut.

3. Donnez cette ligne pour les routeurs R1 et R6.

R1

Dest	Mask	Gateway	Int	Dst

R6

Dest	Mask	Gateway	Int	Dst

Un 8ème sous-réseau est ajouté, reliant R2 via son port eth2 à R4 via son port eth3.

4. Quelle est la seule adresse disponible pour ce sous-réseau, si l'on conserve le même masque pour la décomposition ?

R2 détecte ce nouveau « lien » vers R4. Il modifie sa propre table de routage puis envoie immédiatement son vecteur de distance à R1 et à R6.

5. Donnez les vecteurs de distance envoyés par R2 à R1 et à R6, si la technique de l'horizon partagé est utilisée.

VR2 envoyé à R1 = VR2 envoyé à R6 =

6. Sur réception de ces vecteurs de distance, R1 et R6 modifient leur table de routage. Indiquez uniquement les lignes ajoutées ou modifiées.

R1

Dest	Mask	Gateway	Int	Dst

R6

Dest	Mask	Gateway	Int	Dst

Afin de limiter la taille des tables de routage, certaines entrées redondantes peuvent être supprimées. De même plusieurs entrées peuvent être agrégées en une seule.

7. Donnez toutes les lignes des tables de R1 et de R6 une fois cette optimisation effectuée.

R1

Dest	Mask	Gateway	Int

R6

Dest	Mask	Gateway	Int

N° Anonymat :

Exercice 3 : TCP, HTTP et NAT (40 minutes)

On s'intéresse à une connexion TCP transportant des messages HTTP entre un client et un serveur.

- On suppose que le client se trouve à l'intérieur d'un réseau privé et que le serveur est à l'extérieur du réseau privé (dans le réseau public).
- On suppose que c'est la même machine qui fait NAT et routeur de sortie (*gateway*) du réseau privé.
- On suppose qu'il y a 3 routeurs entre le client et le serveur (incluant le NAT) et que le chemin utilisé par tous les paquets IP est le même dans les 2 sens.
- On suppose que le client comme le serveur envoie des paquets IP avec un TTL initial de 128.
- On observe un même segment TCP transportant des données HTTP, juste avant le NAT (du côté du réseau privé) et juste après le NAT (du côté du réseau public).

Les deux trames transportant ce segment de part et d'autre du NAT sont données en annexe (comme affichées dans Wireshark). Il est rappelé que les trames sont données sans le préambule (de 8 octets) ni le CRC (de 4 octets).

Attention : si les indications de l'énoncé ne permettent pas de répondre à une question, l'indiquer clairement dans la réponse (par exemple : « on ne peut pas savoir »).

1. Qu'est-ce qui prouve que ce segment sort du réseau privé (plutôt que d'y rentrer) ?

2. Quel champ prouve qu'il s'agit du même segment TCP transporté (avant et après le NAT) ?

3. Quelles sont les adresses MAC du routeur de sortie du réseau privé ?

Adresse MAC de l'interface menant au réseau privé :

Adresse MAC de l'interface menant au réseau public :

4. Quelles sont les adresses IP du routeur de sortie du réseau privé ?

Adresse IP de l'interface menant au réseau privé :

Adresse IP de l'interface menant au réseau public :

5. Quelle est l'adresse IP du réseau privé si l'on suppose qu'il a été configuré avec un masque en /26 ?

6. Quelle est l'adresse IP publique que le NAT associe au réseau privé ?

7. Quelle est l'entrée de la table NAT correspondant à la connexion à laquelle appartient le segment TCP observé ?

Privé		**Public**	
Adresse IP	**Numéro de port**	**Adresse IP**	**Numéro de port**

On suppose que le segment observé est le premier segment contenant des données HTTP, envoyé juste après l'établissement de la connexion TCP. Celui-ci contient la première requête GET du client.

8. Quels sont les paramètres des entêtes IP et TCP du segment SYN (envoyé par le client) qui initialise la connexion, coté privé et coté public ? Donnez les numéros de séquence et d'acquittement en valeur réelle (*raw*).

Segment SYN	**Coté privé**		**Coté public**	
Entête IP	TTL		TTL	
	@ IP source		@ IP source	
	@ IP destination		@ IP destination	
Entête TCP	Port source		Port source	
	Port destination		Port destination	
	SeqNum		SeqNum	
	AckNum		AckNum	

9. Quels sont les paramètres des entêtes IP et TCP du segment ACK qui acquitte le segment de données transportant le GET du client, coté privé et coté public ?

Segment ACK	**Coté privé**		**Coté public**	
Entête IP	TTL		TTL	
	@ IP source		@ IP source	
	@ IP destination		@ IP destination	
Entête TCP	Port source		Port source	
	Port destination		Port destination	
	SeqNum		SeqNum	
	AckNum		AckNum	

On suppose que le serveur répond en envoyant l'index de la page dans un message HTTP d'une longueur totale de 4000 octets. Sachant que la MSS est de 1460 octets, celui-ci est transporté par trois segments TCP. On suppose que le mode HTTP utilisé est le mode non persistant. Tous les liens entre le client et le serveur sont à D bit/s et le temps d'aller-retour (supposé constant) entre le client et le serveur est noté *RTT*.

10. Donnez l'expression du temps d'émission d'un segment TCP de longueur minimum (SYN, FIN, ACK), d'un segment TCP de longueur maximum (contenant MSS octets de données), du segment TCP transportant la requête HTTP (GET) et du troisième segment transportant l'index (fin ind).

$t_{t\ \text{TCP min}} =$	$t_{t\ \text{TCP max}} =$	$t_{t\ \text{GET}} =$	$t_{t\ \text{fin ind}} =$

11. Donnez l'expression du temps minimum de la connexion TCP complète (incluant l'ouverture et la fermeture) associée à cet échange HTTP, en fonction des temps précédemment définis.

$T_{min} =$	$t_{t\ \text{TCP min}} +$	$t_{t\ \text{TCP max}} +$	$t_{t\ \text{GET}} +$	$t_{t\ \text{fin ind}} +$	RTT

Annexe exercice 1 : Trace

Time 0.000000 No.1

```
00 09 0f 09 00 02 3c 22 fb 00 b8 af 08 00 45 00
00 4c 71 b1 00 00 ff 11 aa c7 c0 a8 1f 18 c0 a8
1f 01 e5 d4 00 35 00 38 ac fd fe 59 01 00 00 01
00 00 00 00 00 00 07 73 73 68 2d 65 6e 73 0b 75
66 72 2d 69 6e 66 6f 2d 70 36 07 6a 75 73 73 69
65 75 02 66 72 00 00 01 00 01
```

Time 0.010851 N° 2

```
3c 22 fb 00 b8 af 00 09 0f 09 00 02 08 00 45 00
01 23 c7 86 00 00 40 11 15 1c c0 a8 1f 01 c0 a8
1f 18 00 35 e5 d4 01 0f 26 3e fe 59 85 80 00 01
00 01 00 05 00 05 07 73 73 68 2d 65 6e 73 0b 75
66 72 2d 69 6e 66 6f 2d 70 36 07 6a 75 73 73 69
65 75 02 66 72 00 00 01 00 01 c0 0c 00 01 00 01
00 01 51 80 00 04 84 e3 76 76 c0 14 00 02 00 01
00 01 51 80 00 08 05 73 68 69 76 61 c0 20 c0 14
00 02 00 01 00 01 51 80 00 0e 06 73 6f 6c 65 69
6c 04 75 76 73 71 c0 28 c0 14 00 02 00 01 00 01
51 80 00 0e 06 6f 73 69 72 69 73 04 6c 69 70 36
c0 28 c0 14 00 02 00 01 00 01 51 80 00 09 06 74
72 69 74 6f 6e c0 14 c0 14 00 02 00 01 00 01 51
80 00 0e 04 6d 61 79 61 06 69 6e 66 6f 70 36 c0
20 c0 a9 00 01 00 01 00 01 51 80 00 04 86 9d 74
7c c0 4c 00 01 00 01 00 02 a3 00 00 04 86 9d 00
81 c0 7a 00 01 00 01 00 00 15 59 00 04 84 e3 3c
1e c0 60 00 01 00 01 00 01 51 80 00 04 c1 33 18
01 c0 94 00 01 00 01 00 01 51 80 00 04 84 e3 44
2c
```

Time 0.012725 N° 3

```
00 09 0f 09 00 02 3c 22 fb 00 b8 af 08 00 45 00
00 40 00 00 40 00 40 06 27 55 c0 a8 1f 18 84 e3
76 76 cd 83 00 16 18 5a e8 ec 00 00 00 00 b0 02
ff ff 39 e7 00 00 02 04 05 b4 01 03 03 06 01 01
08 0a bd 73 5d 5d 00 00 00 00 04 02 00 00
```

Time 0.015228 N° 4

```
3c 22 fb 00 b8 af 00 09 0f 09 00 02 08 00 45 00
00 34 00 00 40 00 36 06 2b 61 84 e3 76 76 c0 a8
1f 18 00 16 cd 83 d4 3a b4 be 18 5a e8 ed 80 12
fa f0 08 d2 00 00 02 04 05 b4 01 01 04 02 01 03
03 07
```

Time 0.015322 N° 5

```
00 09 0f 09 00 02 3c 22 fb 00 b8 af 08 00 45 00
00 28 00 00 40 00 40 06 27 6d c0 a8 1f 18 84 e3
76 76 cd 83 00 16 18 5a e8 ed d4 3a b4 bf 50 10
10 00 34 95 00 00
```

Time 0.016208 N° 6

```
00 09 0f 09 00 02 3c 22 fb 00 b8 af 08 00 45 00
00 3d 00 00 40 00 40 06 27 58 c0 a8 1f 18 84 e3
76 76 cd 83 00 16 18 5a e8 ed d4 3a b4 bf 50 18
10 00 72 cf 00 00 53 53 48 2d 32 2e 30 2d 4f 70
65 6e 53 53 48 5f 38 2e 31 0d 0a
```

Time 0.017589 N° 7

```
3c 22 fb 00 b8 af 00 09 0f 09 00 02 08 00 45 00
00 28 ca 15 40 00 36 06 61 57 84 e3 76 76 c0 a8
1f 18 00 16 cd 83 d4 3a b4 bf 18 5a e9 02 50 10
01 f6 42 8a 00 00
```

Time 0.029872 N° 8

```
3c 22 fb 00 b8 af 00 09 0f 09 00 02 08 00 45 00
00 51 ca 16 40 00 36 06 61 2d 84 e3 76 76 c0 a8
1f 18 00 16 cd 83 d4 3a b4 bf 18 5a e9 02 50 18
01 f6 b9 7a 00 00 53 53 48 2d 32 2e 30 2d 4f 70
65 6e 53 53 48 5f 37 2e 39 70 31 20 44 65 62 69
61 6e 2d 31 30 2b 64 65 62 31 30 75 32 0d 0a
```

Time 0.029974 N° 9

```
00 09 0f 09 00 02 3c 22 fb 00 b8 af 08 00 45 00
00 28 00 00 40 00 40 06 27 6d c0 a8 1f 18 84 e3
76 76 cd 83 00 16 18 5a e9 02 d4 3a b4 e8 50 10
0f ff 34 58 00 00
```

Time 0.031417 N° 10

```
3c 22 fb 00 b8 af 00 09 0f 09 00 02 08 00 45 00
04 60 ca 17 40 00 36 06 5d 1d 84 e3 76 76 c0 a8
1f 18 00 16 cd 83 d4 3a b4 e8 18 5a e9 02 50 18
01 f6 da 0e 00 00 00 00 04 34 06 14 79 6c 72 5a (…)
00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

(...)

Time 18.095344 N° 150

```
00 09 0f 09 00 02 3c 22 fb 00 b8 af 08 00 45 00
00 64 00 00 40 00 40 06 26 e9 c0 a8 1f 18 84 e3
76 76 cd 83 00 16 18 5a f3 4a d4 3a c3 c4 50 18
10 00 e6 d1 00 00 8a 2f 34 82 66 ac e8 c7 2f 7b
57 f0 0f 68 1d 6e 09 d7 d2 e5 66 4b 33 81 51 6f
50 25 1d 1f 07 5a 74 d9 76 cd 8c 65 de 17 d2 a3
2e cb 3c b7 20 1c 06 b9 ef 20 e7 7d f0 17 a3 21
10 26
```

Time 18.096673 N° 151

```
00 09 0f 09 00 02 3c 22 fb 00 b8 af 08 00 45 00
00 28 00 00 40 00 40 06 27 25 c0 a8 1f 18 84 e3
76 76 cd 83 00 16 18 5a f3 86 d4 3a c3 c4 50 11
10 00 1a f6 00 00
```

Time 18.098568 N° 152

```
3c 22 fb 00 b8 af 00 09 0f 09 00 02 08 00 45 00
00 28 ca 35 40 00 36 06 61 27 84 e3 76 76 c0 a8
1f 18 00 16 cd 83 d4 3a c3 c4 18 5a f3 87 50 10
01 f5 29 01 00 00
```

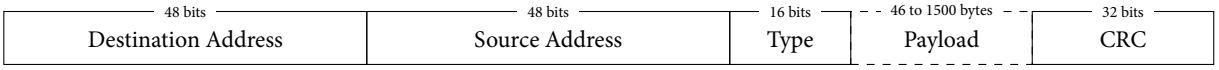
Time 18.102578 N° 153

```
3c 22 fb 00 b8 af 00 09 0f 09 00 02 08 00 45 00
00 28 ca 36 40 00 36 06 61 26 84 e3 76 76 c0 a8
1f 18 00 16 cd 83 d4 3a c3 c4 18 5a f3 87 50 11
01 f5 29 00 00 00
```

Time 18.102731 N° 154

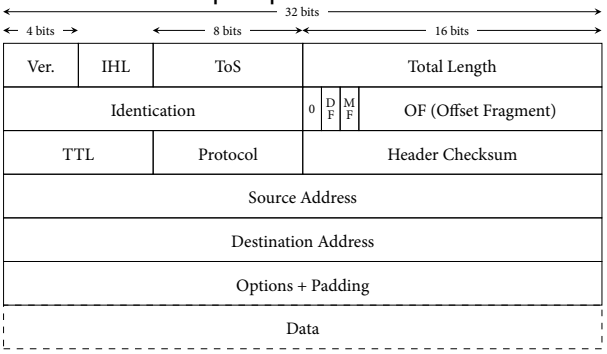
```
00 09 0f 09 00 02 3c 22 fb 00 b8 af 08 00 45 00
00 28 00 00 40 00 40 06 27 25 c0 a8 1f 18 84 e3
76 76 cd 83 00 16 18 5a f3 87 d4 3a c3 c5 50 10
10 00 1a f5 00 00
```

Structure de la trame Ethernet



`0x0200` Xerox PUP | `0x0806` ARP | `0x0800` DOD Internet IP | `0x8035` RARP

Structure du paquet IP

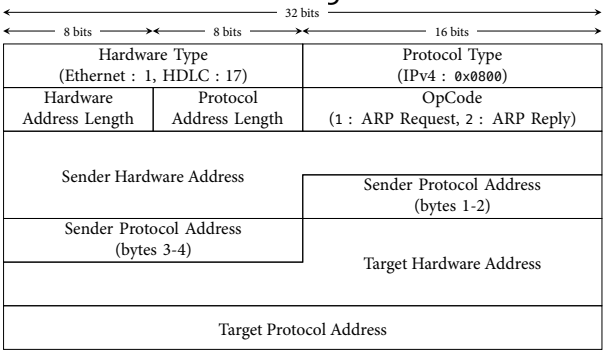


`Ver.` Version d'IP
`IHL` Longueur de l'entête IP ($\times$ 32 bits)
`ToS` Type de service (0 généralement)
`DF` Ne pas fragmenter
`MF` Fragment suivant existe
`OF` Décalage du fragment
`TTL` Durée de vie restante

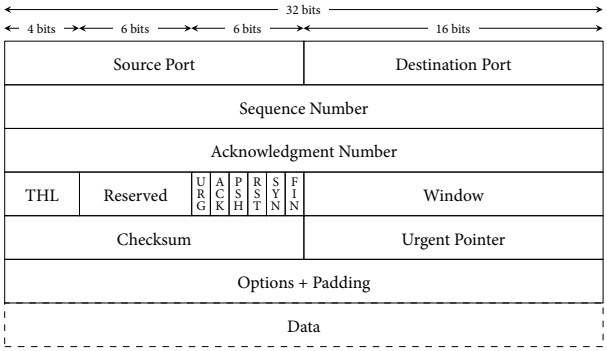
Quelques protocoles transportés :

`1`	ICMP	`8`	EGP
`2`	IGMP	`11`	GLOUP
`4`	IPv4	`17`	UDP
`5`	Stream	`36`	XTP
`6`	TCP	`46`	RSVP

Structure du message ARP



Structure du segment TCP

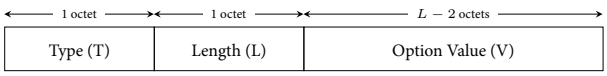


`THL` Longueur de l'entête TCP ($\times$ 32 bits)
`URG` Données urgentes
`ACK` Acquittement
`PSH` Données immédiates
`RST` Réinitialisation
`SYN` Synchronisation
`FIN` Fin

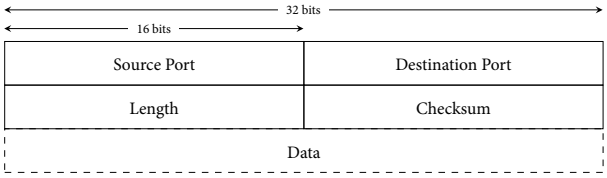
Options codées sur :

- 1 octet à `00` : fin des options
- 1 octet à `01` : NOP (pas d'opération)
- Plusieurs octets de type T-L-V :

T $= 2$ Négociation de la taille max. du segment
T $= 3$ Adaptation de la taille de la fenêtre
T $= 4$ Autorisation des acquittements sélectifs
T $= 8$ Estampilles temporelles



Structure du datagramme UDP



Services associés aux ports (*well-known ports*)

`ftp-data`	`20/tcp`	`dhcp`	`67/68/udp`	`imaps`	`993/tcp`
`ftp`	`21/tcp`	`dns`	`53/udp`	`pop-3`	`110/tcp/udp`
`ssh`	`22/tcp/udp`	`tftp`	`69/udp`	`snmp`	`161/udp`
`telnet`	`23/tcp`	`www`	`80/tcp/udp`	`snmp-trap`	`162/udp`

Structure du message DNS

```
< 2o.>< 2o.><2o.>< 2o.><2o.>< 2o.>   < Qo.>      <Ro.>    < So.>      < Io.>
+-----+-----+----+-----+----+-----+- - - - -+- - - - -+- - - - -+- - - - - - -+
|Ident|Flags|NbQu|NbRep|NbSR|NbInf| Questions |Reponses |Serveurs |Informations |
+-----+-----+----+-----+----+-----+- - - - -+- - - - -+- - - - -+- - - - - - -+
```

`Ident` : identificateur d'échange
`Flags` : indicateurs de paramètres DNS (bit de poids fort à 0 pour une requête et 1 pour une réponse)
`NbQu` : nombre de questions dans le champ `Questions`
`NbRep` : nombre de réponses dans le champ `Reponses`
`NbSR` : nombre de serveurs DNS d'autorité dans le champ `Serveurs`
`NbInf` : nombre d'informations additionnelles dans le champ `Informations`

Une question :

```
<----N-octets----->2octets><2octets>
+- - - - - - - - -+--------+--------+
|       Nom       | Type   | Classe |
+- - - - - - - - -+--------+--------+
```

Une réponse ou **un** serveur d'autorité ou **une** information additionnelle :

```
<--M octets--><  2o. ><  2o. ><4octets>< 2o. ><--D-octets-->
+- - - - - - +------+------+--------+------+- - - - - - -+
| Nom        | Type |Classe| TTL    |Taille|   Données    |
+- - - - - - +------+------+--------+------+- - - - - - -+
```

`Nom` : il est composé de plusieurs labels séparés par des points (ex : www.sorbonne-universite.fr contient 3 labels) et il se termine par 0x00. Un label peut être donné sous la forme d'un *label de nom* ou d'un *label compressé* :

- un label de nom débute par un octet qui précise la longueur en octets du label,
- un label compressé fait référence à un ou plusieurs labels de nom consécutifs déjà présents en début de message. Au lieu de répéter ces labels, le label compressé fait référence à la première occurrence des labels en précisant la valeur 0x0c indiquant qu'il s'agit d'un label compressé, suivie de la position des labels de nom par rapport au début du message DNS. La position est exprimée en nombre d'octets (ex : 0xc010 indique qu'il s'agit d'un label compressé et que le label de nom auquel il fait référence est située à 0x10 octets depuis le début du message DNS).

`Type` : 1 = A (adresse IPv4)
2 = NS (nom de serveur DNS)
5 = CNAME (alias)
15 = MX (serveur de mail)
28 = AAAA (adresse IPv6)

`Classe` : 1 = Internet

`TTL` : validité en secondes

`Taille` : longueur des données en octets

`Données` : nom (pour NS, CNAME et MX) ou adresse (pour A et AAAA)

Table ASCII

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
000	00	␀	032	20	␣	064	40	@	096	60	‘
001	01	☺	033	21	!	065	41	A	097	61	a
002	02	☻	034	22	"	066	42	B	098	62	b
003	03	♥	035	23	#	067	43	C	099	63	c
004	04	♦	036	24	$	068	44	D	100	64	d
005	05	♣	037	25	%	069	45	E	101	65	e
006	06	♠	038	26	&	070	46	F	102	66	f
007	07	•	039	27	'	071	47	G	103	67	g
008	08	◘	040	28	(	072	48	H	104	68	h
009	09		041	29	)	073	49	I	105	69	i
010	0A	◙	042	2A	*	074	4A	J	106	6A	j
011	0B	♂	043	2B	+	075	4B	K	107	6B	k
012	0C		044	2C	’	076	4C	L	108	6C	l
013	0D	♪	045	2D	-	077	4D	M	109	6D	m
014	0E	♫	046	2E	.	078	4E	N	110	6E	n
015	0F	☼	047	2F	/	079	4F	O	111	6F	o
016	10	►	048	30	0	080	50	P	112	70	p
017	11	◄	049	31	1	081	51	Q	113	71	q
018	12	↕	050	32	2	082	52	R	114	72	r
019	13	‼	051	33	3	083	53	S	115	73	s
020	14	¶	052	34	4	084	54	T	116	74	t
021	15	§	053	35	5	085	55	U	117	75	u
022	16	▬	054	36	6	086	56	V	118	76	v
023	17	↨	055	37	7	087	57	W	119	77	w
024	18	↑	056	38	8	088	58	X	120	78	x
025	19	↓	057	39	9	089	59	Y	121	79	y
026	1A		058	3A	:	090	5A	Z	122	7A	z
027	1B	←	059	3B	;	091	5B	[	123	7B	{
028	1C	└	060	3C	<	092	5C	\	124	7C	|
029	1D	↔	061	3D	=	093	5D	]	125	7D	}
030	1E	▲	062	3E	>	094	5E	^	126	7E	~
031	1F	▼	063	3F	?	095	5F	_	127	7F	⌂

Annexe exercice 3 : Premier segment TCP transportant des données HTTP

Avant le NAT (dans le réseau privé) :

```
Frame: 767 bytes on wire (6136 bits), 767 bytes captured (6136 bits)
Ethernet II, 00:22:68:0d:ca:8f, Dst: 00:22:6b:45:1f:1b
    Destination: CiscoLinksys_45:1f:1b (00:22:6b:45:1f:1b)
    Source: HonHaiPrecis_0d:ca:8f (00:22:68:0d:ca:8f)
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.1.100, Dst: 74.125.91.113
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 753
    Identification: 0xa26e (41582)
    010. .... = Flags: 0x2, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 128
    Protocol: TCP (6)
    Header Checksum: 0xec91
    Source Address: 192.168.1.100
    Destination Address: 74.125.91.113
Transmission Control Protocol, Src Port: 4330, Dst Port: 80, Seq: 1, Ack: 1
    Source Port: 4330
    Destination Port: 80
    Sequence Number: 1    (relative sequence number)
    Sequence Number (raw): 952809728
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 2709749796
    0101 .... = Header Length: 20 bytes (5)
    Flags: 0x018 (PSH, ACK)
    Window: 65044
    [Calculated window size: 260176]
    [Window size scaling factor: 4]
    Checksum: 0x6bea [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
Hypertext Transfer Protocol
    GET /safebrowsing/rd/goog-malware-shavar_s HTTP/1.1\r\n
    Host: safebrowsing-cache.google.com\r\n
    User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US;
    rv:1.9.0.14) Gecko/2009082707 Firefox/3.0.14 (.NET CLR 3.5.30729)\r\n
    Accept: text/html,application/xhtml+xml,application/xml\r\n
    Accept-Language: en-us,en;q=0.5\r\n
    Accept-Encoding: gzip,deflate\r\n
    Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
    Keep-Alive: 300\r\n
    Connection: keep-alive\r\n
    Cookie: ...
```

Après le NAT (dans le réseau public) :

```
Frame: 767 bytes on wire (6136 bits), 767 bytes captured (6136 bits)
Ethernet II, 00:08:74:4f:36:2, Dst: 00:0e:d6:bf:6c:01
    Destination: Cisco_bf:6c:01 (00:0e:d6:bf:6c:01)
    Source: Dell_4f:36:23 (00:08:74:4f:36:23)
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 71.192.34.104, Dst: 74.125.91.113
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 753
    Identification: 0xa26e (41582)
    010. .... = Flags: 0x2, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 127
    Protocol: TCP (6)
    Header Checksum: 0x4576
    Source Address: 71.192.34.104
    Destination Address: 74.125.91.113
Transmission Control Protocol, Src Port: 5330, Dst Port: 80, Seq: 1, Ack: 1
    Source Port: 5330
    Destination Port: 80
    Sequence Number: 1    (relative sequence number)
    Sequence Number (raw): 952809728
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 2709749796
    0101 .... = Header Length: 20 bytes (5)
    Flags: 0x018 (PSH, ACK)
    Window: 65044
    [Calculated window size: 260176]
    [Window size scaling factor: 4]
    Checksum: 0xbcb3 [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
Hypertext Transfer Protocol
    GET /safebrowsing/rd/goog-malware-shavar_s HTTP/1.1\r\n
    Host: safebrowsing-cache.google.com\r\n
    User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US;
    rv:1.9.0.14) Gecko/2009082707 Firefox/3.0.14 (.NET CLR 3.5.30729)\r\n
    Accept: text/html,application/xhtml+xml,application/xml\r\n
    Accept-Language: en-us,en;q=0.5\r\n
    Accept-Encoding: gzip,deflate\r\n
    Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
    Keep-Alive: 300\r\n
    Connection: keep-alive\r\n
    Cookie: ...
```