

The standard answer key

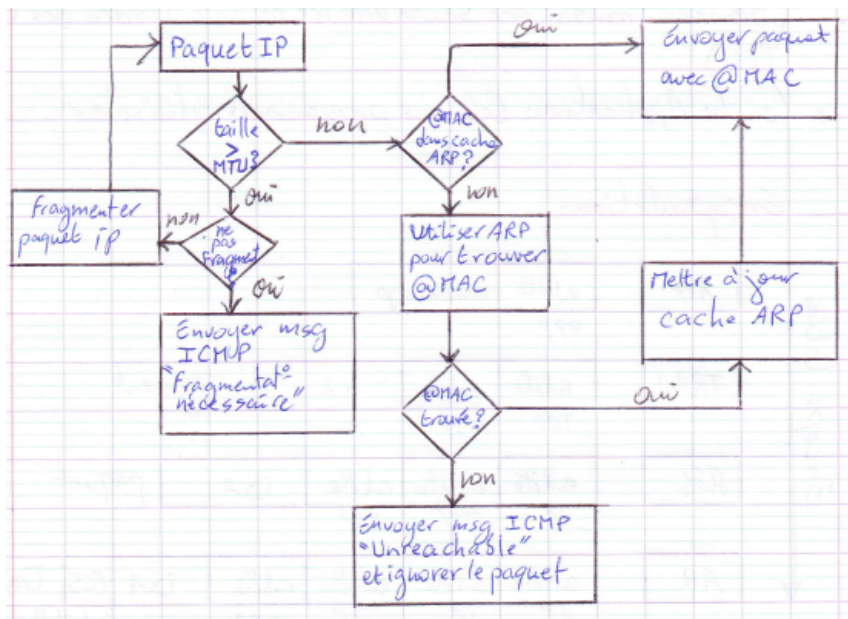
Exercice 1 (10pts)

1. (1.5Pts)

Classe d'adresses	Plage d'adresses public
A	1.0.0.0 - 9.255.255.255 11.0.0.0 - 126-255.255.255
B	128.0.0.0 - 172.15.255.255 172.32.0.0 - 191-255.255.255
C	192.0.0.0- 192.167.255.255 à 192.169.0.0 - 223-255.255.255

TABLE 1 – Plages d'adresses public.

2. 218.115.20.0 \rightarrow 218.115.20.31. 218.115.20.0 est l'adresse réseau (**0.5Pt**), 218.115.20.31 est l'adresse de broadcast (**0.5Pt**). La plage d'adresse valide est : 218.115.20.1 \rightarrow 218.115.20.30 (**0.5Pt**)
3. Oui, elle peut sortir puisque le NAT est désactivé et que l'adresse destination est publique (donc routable). Le problème se situe au niveau de la réponse qui ne peut pas être routée depuis la machine destination vers la machine émettrice de la requête, puisque l'adresse source de sortie est privée (donc non routable). (**1.5Pt**)
4. Le **NAT statique** offre l'accès à Internet à une machine, même si elle possède une adresse privée. Faisant toujours la même correspondance, on donne ainsi la possibilité à une machine d'être vue sur Internet, ce qui est intéressant pour héberger des services vus de l'extérieur. Le serveur WEB de l'entreprise devant être accessible de manière simple de l'extérieur, il lui faut une adresse publique fixe. Pour cela on utilisera donc une translation statique. (**1Pt**)
5. EUI-64 Address :
 - 02 :00 :4c :ff :fe :4f :4f :50
 - 1111 1110 10 + 54 zeros + EUI-64 Address
 - fe80 :0000 :0000 :0200 :4cff :fe4f :4f50 /64 \rightarrow fe80 : :200 :4cff :fe4f :4f50 /64 (**0.5Pt**)
6. **2a01 :5d8 :ccf1 :4 :200 :4cff :fe4f :4f50/64** (**1Pt**)
7. — IPv4 : $(2^{32})/(10^9) = 4.295$ secondes (moins de 5 secondes!) (**0.5Pt**)
 - IPv6 : With 16 bytes there are 2^{128} or $3.4 * 10^{38}$ addresses. $3.4 * 10^{38} * 10^{-12} * 10^{-6} = 3.4 * 10^{20}$ secondes (**0.5Pt**)
8. (**2Pt**)



Exercice 2 (10pts)

1. 08 :00 :20 :01 :b4 :32 → @MAC Destination **(0.25Pt)**
2. 08 :00 :20 :00 :61 :f3 → @MAC Source **(0.25Pt)**
3. 0x800 → Ether Type → Protocole de la couche sup.=IPv4 **(0.25Pt)**
4. 0x4 → (4)₁₀ → Version 4 du protocole IP **(0.25Pt)**
5. 0x5 → (5)₁₀ → 5*4 octets = 20 octets (5 mots) → Taille de l'entête IPv4 **0.25Pt**
6. 0x00 → (0)₁₀ → Aucun Service. **(0.25Pt)**
7. 0x0028 → $0 * 16^3 + 0 * 16^2 + 2 * 16^1 + 8 * 16^0 = 40$ octets → Longueur totale du datagramme IP. **(0.25Pt)**
8. 0x0c39 → $0 * 16^3 + c * 16^2 + 3 * 16^1 + 9 * 16^0 = 3129$ → Identificateur du datagramme IP. **(0.25Pt)**
9. 0x0000 → 0000000000000000 → Les 3 drapeaux sont à zéro (bit réservé, don't fragment et more Fragment). Les 13 bits suivants représentent l'offset = (0)₁₀ **(0.5Pt)**
10. 0x1e → $1 * 16^1 + e * 16^0 = (30)_{10}$ → TTL (Time To Live). **(0.25Pt)**
11. 0x06 → $0 * 16^1 + 6 * 16^0 = 6_{10}$ → Protocole encapsulé dans le datagramme IP = TCP **(0.25Pt)**
12. 0x8077 → $8 * 16^3 + 0 * 16^2 + 7 * 16^1 + 7 * 16^0 = 32887$ → Checksum **(0.25Pt)**
13. 0xc0.0x09. 0xc8. 0x0b → 192.9.200.11 → @IPv4 source **(0.25Pt)**
14. 0xc0.0x09. 0xc8. 0x01 → 192.9.200.1 → @IPv4 destination **(0.25Pt)**
15. 0x0450 → $0 * 16^3 + 4 * 16^2 + 5 * 16^1 + 0 * 16^0 = 1104$ → Port Source **(0.25Pt)**
16. 0x0015 → $0 * 16^3 + 0 * 16^2 + 1 * 16^1 + 5 * 16^0 = 21$ → Port Destination **(0.25Pt)**

17. $0x0006e802 \rightarrow 0 * 16^7 + 0 * 16^6 + 0 * 16^5 + 6 * 16^4 + e * 16^3 + 8 * 16^2 + 0 * 16^1 + 2 * 16^0 = 452610_{10} \rightarrow$ Numéro de séquence (d'ordre). **(0.25Pt)**
18. $0x00803e08 \rightarrow 0 * 16^7 + 0 * 16^6 + 8 * 16^5 + 0 * 16^4 + 3 * 16^3 + e * 16^2 + 0 * 16^1 + 8 * 16^0 = 8404488_{10} \rightarrow$ Numéro de séquence (d'ordre). **(0.25Pt)**
19. $0x5 \rightarrow (5)_{10} \rightarrow 5 * 4 \text{octets} = 20 \text{octets} \rightarrow$ Longueur de l'entête TCP. **(0.25Pt)**
20. $0x010 \rightarrow 00000010000 \rightarrow$ Les six bits en rouge sont des bits réservés et dans les second 6bits, on constate que le bit Ack=1 \rightarrow C'est un segmente TCP Accusé de réception. **(0.5Pt)**
21. $0x1000 \rightarrow 4096 \rightarrow$ Taille de la fenêtre **(0.25Pt)**
22. $0x64be \rightarrow (25790)_{10} \rightarrow$ Checksum **(0.25Pt)**
23. $0x0000 \rightarrow (0)_{10} \rightarrow$ Pointeur Urgent **(0.25Pt)**
24. $0x000000000000 \rightarrow$ Il s'agit 6 octets de bourrage pour amener la trame Ethernet à la longueur minimale (64 octets en tout, c'est à dire 46 octet de données+18 octets d'entête (6+6+2+4). **(2Pt)**
25. D'après la valeur du port destination, la machine 1 se connecte à un serveur FTP (21). **(1.5Pt)**