UPMC - Licence d'Informatique
3I014 – 2017/2018

Reportez ici votre numéro d'anonymat :

Examen 3I014 « Réseaux »
Mercredi 20 Juin 2018 – Durée : 2 heures

Autorisés : 1 feuille A4 manuscrite recto/verso + 1 calculette
Interdit : tout dispositif électronique (téléphone, traducteur, …)

Voici :
- 4 feuilles contenant les énoncés et les zones de réponse à compléter (sans déborder). V**ous devez reporter votre numéro d'anonymat sur chacune des feuilles.**
- 1 feuille d'annexe que vous pouvez détacher.

Exercice 1 : Commutation (5 points)

Deux machines A et B sont reliées par un réseau à commutation de paquets en mode circuit virtuel (de type X25). On suppose que le circuit virtuel a été établi et passe par N nœuds de commutation. Toutes les liaisons du réseau sont supposées avoir le même débit de D bit/s et une longueur moyenne de x mètres. La vitesse de propagation de l'information sur chaque lien est de C m/s. L'en-tête de tous les paquets est de E_p bits et tous les paquets sont encapsulés dans des trames dont l'en-tête, l'en-queue et les fanions (d'ouverture et de fermeture) totalisent E_t bits. On supposera que la longueur moyenne du champ de données d'un paquet de données est de l bits et que celle d'un paquet d'acquittement est nulle (un paquet d'acquittement se réduit à son en-tête).

1. Donner, en fonction des paramètres de l'énoncé, les temps moyens de transmission et de propagation d'un paquet de donnée et d'un paquet d'acquittement, sur chaque liaison du réseau.

Temps moyen de transmission d'un paquet de données sur un lien : $tt_d =$

Temps moyen de propagation d'un paquet de données sur un lien : $tp_d =$

Temps moyen de transmission d'un paquet d'acquittement sur un lien : $tt_a =$

Temps moyen de propagation d'un paquet d'acquittement sur un lien : $tp_a =$

On suppose dans un premier temps que A doit envoyer un seul paquet à B et que celui-ci doit l'acquitter de bout en bout. Plus précisément, le paquet de A doit traverser les N nœuds du réseau avant de parvenir à B, puis l'acquittement de B doit faire le chemin inverse pour revenir jusqu'à A.

2. Exprimer en fonction de tt_d, tp_d, tt_a et tp_a (et des paramètres de l'énoncé), le temps moyen séparant le début d'émission du paquet de A et la fin de réception par A de l'acquittement correspondant.

$T = (\quad) \times tt_d + (\quad) \times tp_d + (\quad) \times tt_a + (\quad) \times tp_a$

On suppose maintenant que A doit envoyer deux paquets à B et que celui-ci doit les acquitter globalement de bout en bout avec un seul acquittement. La fenêtre d'émission de niveau paquet est supposée suffisante pour ne pas bloquer l'émission des deux paquets issus de A (sur aucun nœud du réseau). Plus précisément, A envoie consécutivement deux paquets dans le réseau, ces deux paquets traversent les N nœuds du réseau avant de parvenir à B, puis l'acquittement de B (acquittant simultanément les deux paquets issus de A) se propage sur le chemin inverse.

3. Exprimer en fonction de tt_d, tp_d, tt_a et tp_a, le temps moyen séparant le début d'émission du premier paquet de A et la fin de réception par A de l'acquittement correspondant.

T = ( ) × tt_d + ( ) × tp_d + ( ) × tt_a + ( ) × tp_a

On suppose, comme précédemment, que A doit envoyer deux paquets à B, mais on considère maintenant que le mécanisme d'acquittement dans le réseau est local. Chaque nœud du réseau renvoie donc localement un seul paquet d'acquittement (qui acquitte simultanément les deux paquets de données et qui n'est pas retransmis par les autres nœuds) dès qu'il a reçu les deux paquets de données issus de A.

4. Exprimer en fonction de tt_d, tp_d, tt_a et tp_a, le temps moyen séparant le début d'émission du premier paquet de A et la fin de réception de l'acquittement correspondant.

T = ( ) × tt_d + ( ) × tp_d + ( ) × tt_a + ( ) × tp_a

On revient à un mécanisme d'acquittement global. On suppose toujours que A doit envoyer deux paquets à B et que celui-ci doit les acquitter de bout en bout avec un seul acquittement. On considère maintenant que le réseau est relativement chargé et on suppose que lorsqu'un paquet (de donnée ou d'acquittement) est commuté, il trouve en moyenne 5 autres paquets de données en attente de transmission sur le lien de sortie avant lui. On considèrera que le temps de transmission de tous les paquets de données (ceux de A et les autres) sont identiques (en moyenne).

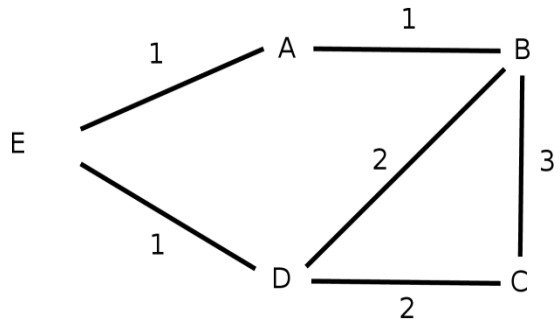
5. Exprimer en fonction de tt_d, tp_d, tt_a et tp_a, le temps moyen séparant le début d'émission du premier paquet de A et la fin de réception de l'acquittement correspondant.

T = ( ) × tt_d + ( ) × tp_d + ( ) × tt_a + ( ) × tp_a

Reportez ici votre numéro d'anonymat :

Exercice 2 : Routage (5 points)

On considère le réseau suivant utilisant un algorithme de routage avec vecteur de distance :



1. Donnez les tables de routage initiales pour chacun des nœuds.

A

dest	next	dist

B

dest	next	dist

C

dest	next	dist

D

dest	next	dist

E

dest	next	dist

2. En supposant que l'ordre des routeurs pour l'envoi des vecteurs de distance soit A, B C, D puis E, donnez l'état des tables de routage après une itération des mises à jour :

A

dest	next	dist

B

dest	next	dist

C

dest	next	dist

D

dest	next	dist

E

dest	next	dist

3. L'algorithme de routage a-t-il convergé ?

4. Le nœud D tombe en panne. Quelles tables sont touchées par cette panne et quel est leur nouvel état ?

A

dest	next	dist

B

dest	next	dist

C

dest	next	dist

D

dest	next	dist

E

dest	next	dist

5. On suppose que ces tables sont immédiatement envoyées aux routeurs voisins. Décrivez l'état des tables qui ont été modifiées par ces envois.

A			B			C			D			E		
dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist

6. Quelle mise à jour est encore nécessaire pour que l'algorithme de routage converge. Décrivez les modifications faites.

7. Le lien entre B et C tombe en panne. Décrivez comment le problème de comptage à l'infini peut survenir.

8. Quelle solution classique peut-on envisager pour pallier le problème ?

Reportez ici votre numéro d'anonymat :

Exercice 3 : Adressage IP (5 points)

Un réseau privé est un réseau qui utilise les plages d'adressage IP définies par la RFC 1918 « Address Allocation for Private Internets ». Ces adresses ne sont pas routées sur Internet.
Les adresses réservées pour des réseaux privés sont les suivantes :
10.0.0.0 / 8 172.16.0.0 / 12 192.168.0.0 / 16
Remarque : un masque « /8 » correspond à « 255.0.0.0 », celui « /12 » correspond à « 255.240.0.0 » et « /16 » à « 255.255.0.0 ».

Une société dispose d'un parc de 1400 machines réparties équitablement entre 20 sous-réseaux.

1. Définir le nombre minimum de bits consacrés aux identifiants des sous-réseaux.

2. Combien de bits faut-il pour coder les différentes machines de chaque sous-réseau ?

3. Quel(s) préfixe(s) réseau parmi ceux proposés permet(tent) de proposer un plan d'adressage qui fonctionne ?

4. L'administrateur souhaite retenir l'adresse privée 172.16.0.0 / 12.

   a. Combien de stations pourra-t-on alors avoir au maximum dans chaque sous-réseau créé ?

   b. Proposez les identifiants des 4 premiers sous-réseaux créés ainsi que leur masque et les plages d'adresses possibles pour les stations. Vous commencerez la notation du premier sous-réseau en mettant les *x* bits de sous-réseau à 0, puis (*x*-1) premiers bits à 0 et le dernier à 1, …

Sous-réseau 1 :____ . ____ . ____ . ____ / _____

Plage d'adresses des machines : ____ . ____ . ____ . ____ → ____ . ____ . ____ . ____

Sous-réseau 2 : ____ . ____ . ____ . ____ / _____

Plage d'adresses des machines : ____ . ____ . ____ . ____ → ____ . ____ . ____ . ____

Sous-réseau 3:____ . ____ . ____ . ____ / _____

Plage d'adresses des machines : ____ . ____ . ____ . ____ → ____ . ____ . ____ . ____

Sous-réseau 4 :____ . ____ . ____ . ____ / _____

Plage d'adresses des machines : ____ . ____ . ____ . ____ → ____ . ____ . ____ . ____

Reportez ici votre numéro d'anonymat :

Exercice 4 : Protocole POP (5 points)

POP (Post Office Protocol) est un protocole de niveau applicatif qui permet de rapatrier localement des messages électroniques stockés sur un serveur de messagerie. Pour cela, POP établit une connexion avec le serveur, qui demande une authentification du client avant que les messages puissent être téléchargés.

On considère d'abord la première trame d'un échange POP généré avec Thunderbird, donnée sans préambule ni CRC :

```
0000  00 04 80 5f 68 00 00 06  5b 26 d9 02 08 00 45 00   ..._h... [&....E.
0010  00 30 0f 01 40 00 80 06  00 00 89 c2 c0 f5 89 c2   .0..@... ........
0020  a0 3c 05 9d 00 6e 98 27  bf 3b 00 00 00 00 70 02   .<...n.' .;....p.
0030  40 00 70 fa 00 00 02 04  05 b4 01 01 04 02        @.p..... ......
```

Pour les questions suivantes, **vous justifierez vos réponses** en mettant en valeur la partie de la trame qui vous a permis de répondre. **Il n'est pas demandé d'analyser tous les champs de la trame**.

1. Quel est le numéro de port du serveur sur lequel se connecte le protocole POP ? Vous donnerez cette valeur en décimal.

2. Quel est le protocole de niveau transport utilisé par POP ? Donner le numéro de séquence de cette trame.

3. Quels sont les « flags » de niveau transport positionnés à 1 ? Déduisez-en l'objectif de cette trame, et expliquez brièvement quelles seront les deux prochaines trames de l'échange.

4. Quel est le nombre maximal d'octets de données applicatives qui pourront être encapsulés dans une trame ? (On suppose que le serveur est d'accord avec la valeur proposée ici par le client)

L'échange se poursuit avec plusieurs trames visant à authentifier le client auprès du serveur. L'authentification utilisée est ici du type login / mot de passe.

La trame dans laquelle le client envoie son mot de passe est la suivante :

```
0000  00 04 80 5f 68 00 00 06  5b 26 d9 02 08 00 45 00   ..._h... [&....E.
0010  00 32 0f 06 40 00 80 06  00 00 89 c2 c0 f5 89 c2   .2..@... ........
0020  a0 3c 05 9d 00 6e .. ..  .. .. b1 e8 78 f1 50 18   .<...n.' .X..x.P.
0030  43 e6 74 df 00 00 64 6d  46 79 65 58 4d 3d 0d 0a   C.t...dm FyeXM=..
```

5. Sachant que 28 octets de données applicatives ont été émis par le client entre la première trame (étudiée ci-dessus) et cette trame (les données de cette trame n'étant pas comptées dans les 28 octets), complétez la partie manquante de cette trame en justifiant brièvement.

6. Quel est le nombre d'octets de données applicatives encapsulés dans cette trame ? Justifiez. Déduisez-en ce qui semble être le mot de passe client transmis dans cette trame, tel que vous pouvez le lire en ASCII.

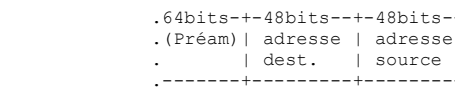
En réalité, le mot de passe du client est transmis sur le réseau en base 64, qui est simplement un format pour le codage de l'information et peut donc être facilement décodé. Dans le cas présent, on peut alors rapidement déduire que le mot de passe utilisé par le client est en fait *varys*.

7. Que pouvez-vous dire de la sécurité de cette authentification dans le protocole POP ? (A quelle attaque simple est-elle sensible, et quelle idée simple proposez-vous pour y remédier ?)

Accessoirement, on pourrait également conseiller au client de choisir un mot de passe un peu plus robuste…

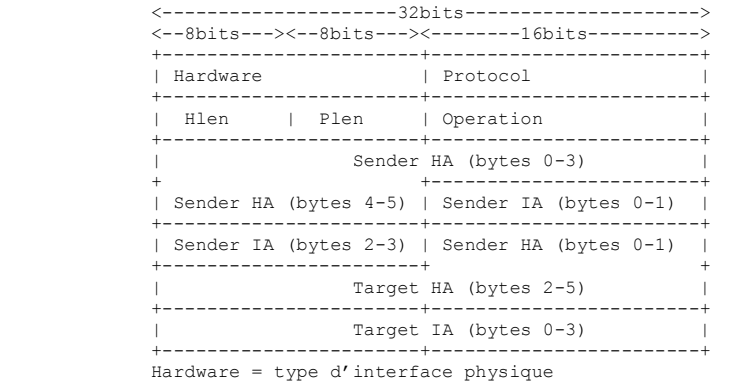
Annexe

Structure d'une trame Ethernet



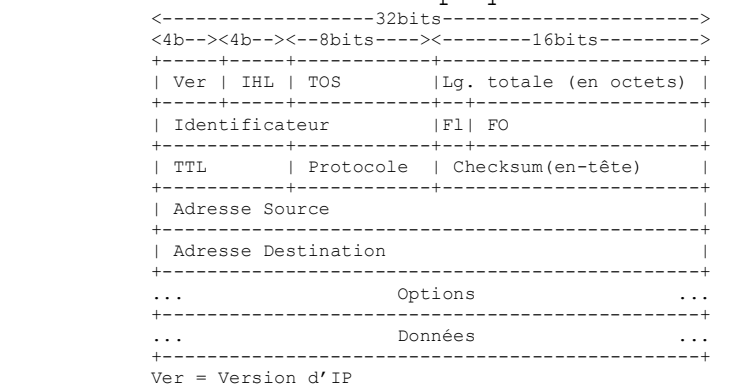
Quelques types : 0x0800 = DoD Internet (IP)
0x0806 = ARP
0x8035 = RARP

Structure d'un paquet ARP



Hardware = type d'interface physique
ex : 0x0001 pour Ethernet
Protocol = type de protocole pour lequel une requête a été émise
ex : 0x0800 pour IP
Hlen = lg de l'adresse physique (en octets)
Plen = lg de l'adresse protocolaire (en octets)
Operation = type d'opération à effectuer par le récepteur
ex : 0x0001 pour une requête ARP
0x0002 pour une réponse ARP
Sender HA = adresse physique (Ethernet) de l'émetteur
Sender IA = adresse protocolaire (IP) de l'émetteur
Target HA = adresse physique (Ethernet) du récepteur
Target IA = adresse protocolaire (IP) du récepteur

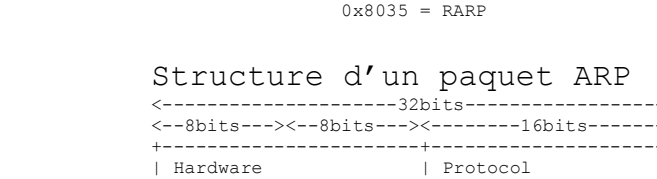
Structure d'un paquet IP



Ver = Version d'IP
IHL = Longueur de l'en-tête IP (en mots de 4 octets)
TOS = Type de service (zéro généralement)
Fl (3 premiers bits) = Bits pour la fragmentation
* 1er = réservé
* 2ème = DF (Ne pas fragmenter)
* 3ème = MF (Fragment suivant existe)
FO (13 bits suivants) = Position relative du fragment dans le datagramme initial (déplacement exprimé en mots de 8 octets (seuls un datagramme complet ou un premier fragment peuvent avoir ce champ à 0)
TTL = Durée de vie restante

Protocole = protocole transporté
ex : 1 = ICMP
2 = IGMP
6 = TCP
8 = EGP
11 = GLOUPS
17 = UDP
89 = OSPF

Structure d'un message UDP



Structure d'un segment TCP

THL = Longueur de l'entête TCP sur 4 bits (en mots de 4 octets)
Flags = indicateur codé sur 6 bits, de gauche à droite
* 1er = URG (Données urgentes)
* 2ème = ACK (Acquittement)
* 3ème = PSH (Données immédiates)
* 4ème = RST (Réinitialisation)
* 5ème = SYN (Synchronisation)
* 6ème = FIN

Options = suite d'options codées sur
* un seul octet :
00 = Fin des options
01 = NOP (pas d'opération)
* plusieurs octets, avec un codage TLV
T = un octet pour le type de l'option
2 Négociation de la taille max. du segment
3 Adaptation de la taille de la fenêtre
4 Autorisation des acquittements sélectifs
8 Estampilles temporelles
L = un octet pour la taille totale de l'option
V = valeur de l'option (sur L-2 octets)

Services associés aux ports

ftp-data	20/tcp
ftp	21/tcp
ssh	22/tcp
telnet	23/tcp
smtp	25/tcp
dns	53/udp
www	80/tcp
pop-3	110/tcp
imap	143/tcp
bgp	179/tcp
snmp	161/udp

...