

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

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

Voici :

- 4 feuilles contenant les énoncés et les zones de réponse à compléter (sans déborder). **Vous 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.

$$T = (\quad) \times tt_d + (\quad) \times tp_d + (\quad) \times tt_a + (\quad) \times tp_a$$
$$T = (N+1) \times tt_d + (N+1) \times tp_d + (N+1) \times tt_a + (N+1) \times tp_a$$

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.

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

$$T = (N+2) \times tt_d + (N+1) \times tp_d + (N+1) \times tt_a + (N+1) \times tp_a$$

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.

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

$$T = (2) \times tt_d + (1) \times tp_d + (1) \times tt_a + (1) \times tp_a$$

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).

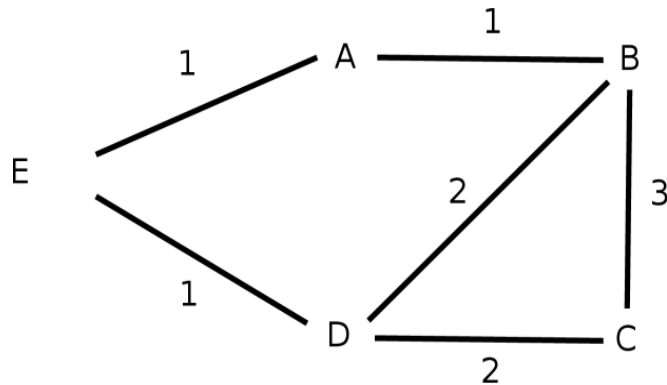
5. Exprimer en fonction de tt_d , tp_d , tt_a et tp_a , le temps moyen séparant le début d'émission du premier paquet de A et la fin de réception de l'acquittement correspondant.

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

$$T = (6(N+2)) \times tt_d + (N+1) \times tp_d + (N+1) \times tt_a + (6(N+1)) \times tp_a$$

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			B			C			D			E		
dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist
A	-	0	A	A	1							A	A	1
B	B	1	B	-	0	B	B	3	B	B	2			
			C	C	3	C	-	0	C	C	2			
			D	D	2	D	D	2	D	-	0	D	D	1
E	E	1							E	E	1	E	-	0

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			B			C			D			E		
dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist
A	-	0	A	A	1	A	B	4	A	E	2	A	A	1
B	B	1	B	-	0	B	B	3	B	B	2	B	A	2
C	B	4	C	C	3	C	-	0	C	C	2	C	D	3
D	E	2	D	D	2	D	D	2	D	-	0	D	D	1
E	E	1	E	A	2	E	D	3	E	E	1	E	-	0

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

Oui. Les tables contiennent les distances optimales.

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

Les tables de B, C et E sont affectées.

A			B			C			D			E		
dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist
A	-	0	A	A	1	A	B	4				A	A	1
B	B	1	B	-	0	B	B	3				B	A	2
C	B	4	C	C	3	C	-	0				C	D	-
D	E	2	D	D	-	D	D	-				D	D	-
E	E	1	E	A	2	E	D	-				E	-	0

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 reçoit VB. Aucune modification.
C reçoit VB ce qui modifie son entrée pour E ($E - B - 5$).
B reçoit VC. Aucune modification.
A reçoit VE et se rend compte que son entée pour D n'est plus valide car elle passe par E qui n'a plus de chemin pour D et donc met une distance à l'infini pour cette entrée.

A			B			C			D			E		
dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist	dest	next	dist
A	-	0	A	A	1	A	B	4				A	A	1
B	B	1	B	-	0	B	B	3				B	A	2
C	B	4	C	C	3	C	-	0				C	D	-
D	E	-	D	D	-	D	D	-				D	D	-
E	E	1	E	A	2	E	B	5				E	-	0

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

Il faut que E reçoive le vecteur de distance de A pour qu'il mette à jour son entrée pour C qui devient $C - A - 5$.

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

B met à jour sa table de routage et envoie normalement son vecteur de distances à A. Mais si dans le même temps, A envoie son propre vecteur de distances à B avant d'avoir reçu celui de B, alors B pense que A a un chemin de poids 4 pour aller à C et modifie son entrée. Lors que B enverra à nouveau son vecteur à A, ce dernier mettra sa table à jour avec un chemin de poids 6 pour aller à C et ainsi de suite.

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

Les solutions habituelles (vecteur de chemin, horizon partagé) permettraient de résoudre le problème.

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.

Pour coder 20 sous-réseaux, il faut 5 bits.

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

Il faut $1400/20=70$ adresses de machines par sous-réseau ce qui nécessite 7 bits qui permettent de coder au plus 128 machines.

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

Il faut 7 bits pour les machines et 5 pour les sous-réseaux soit au moins 12 bits. Il faut donc que le masque de l'adresse privé soit inférieur ou égal à $32 - 12 = 20$ bits. Les 3 adresses de réseau privé correspondent :

10.0.0.0 / 8 , 172.16.0.0 / 12 et 192.168.0.0 / 16

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 ?

Il y a $8+4=12$ bits de réservé pour le préfixe réseau auquel on ajoute 5 bits pour les sous-réseaux soit 17 bits. Reste donc $32 - 17 = 15$ bits pour les machines. $2^{15} - 2 = 32766$ machine par sous-réseau.

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 : ____ . ____ . ____ . ____ / ____

On retient donc le préfixe réseau privé 172.16.0.0 / 12 auquel on va ajouter les 5 bits au masque pour créer les sous-réseaux soit un masque en /17

10101100 . 0001 0000 . 0 000 0000 . 0000 0000 / 17

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

172.16.0.1 → 172.16.127.254

10101100 . 0001 0000 . 0 000 0000 . 0000 0001 → 10101100 . 0001 0000 . 0 111 1111 . 1111 1110

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

172.16.128.0 / 12 10101100 . 0001 0000 . 1 000 0000 . 0000 0000

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

172.16.128.1 → 172.16.255.254

10101100 . 0001 0000 . 1 000 0000 . 0000 0001 → 10101100 . 0001 0000 . 1 111 1111 . 1111 1110

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

172.17.0.0/17 10101100 . 0001 0001 . 0 000 0000 . 0000 0000

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

172.17.0.1 → 172.17.127.254

10101100 . 0001 0001 . 0 000 0000 . 0000 0001 → 10101100 . 0001 0001 . 0 111 1111 . 1111 1110

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

172.18.0.0/17 10101100 . 0001 0001 . 1 000 0000 . 0000 0000

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

Adresses 172.17.128.1 → 172.17.255.254

10101100 . 0001 0001 . 1 000 0000 . 0000 0001 → 10101100 . 0001 0001 . 1 111 1111 . 1111 1110

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.

110

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

TCP.
98 27 bf 3b.

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.

SYN. Ouverture de connexion TCP.
Trames SYN/ACK et ACK qui complètent cette ouverture.

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)

MSS = 1460 octets

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.

28 octets de données + 1 octet consommé par SYN = 29 octets de plus pour le numéro de séquence.
98 27 bf 58

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.

10 octets
dmFyeXM=

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 ?)

Sensible à toute attaque de type espionnage passif du réseau. Il faut chiffrer le mot de passe.

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