

Durée : 1h30 — Documents autorisés

■ ■ ■ **Audit réseau & Analyse de trafic — 7 points**

1– Soient les trames suivantes capturées au sein du même réseau local :

7pts ⇒ trame ❶

0000	24 11 45 AB 39 A5 4C 6D 58 23 68 A8 08 00 45 00	\$.E.9.LmX#h...E.
0010	00 28 AA 8E 00 00 3F 06 6B D8 C1 32 B9 11 C9 11	.(....?.k..2....
0020	22 14 00 16 4F E2 00 00 00 00 00 00 01 50 14	"...O.....P.
0030	00 00 FA 6D 00 00	...m..

⇒ trame ❷

0000	01 56 BC AB 10 20 4C 6D 58 23 68 A8 08 00 45 00	.V... LmX#h...E.
0010	00 28 64 62 00 00 3E 06 86 E8 A4 51 01 19 C9 11	.(db..>....Q....
0020	23 0A 01 BB 28 10 00 00 00 00 00 00 01 50 12	#... (.....P.
0030	00 00 F4 80 00 00	.....

**Questions :**

- Que pouvez vous apprendre d'après le **contenu de ces trames** sur la configuration du réseau, les matériels qui communiquent entre eux et les services utilisés ? (6pts)  
*Justifiez vos réponses.*
- D'après votre analyse, **sur quelle machine du réseau** cette capture peut avoir eu lieu ? (1pt)

■ ■ ■ **Programmation réseau — 3 points**

2– Soit le programme suivant :

3pts

```

1 #!/usr/bin/python
2 import os, socket, sys
3
4 adresse_hote = ''
5 numero_port = 6800
6 tsap_relais = ('', 6789)
7 ma_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM, socket.IPPROTO_TCP)
8 ma_socket.bind(adresse_hote, numero_port)
9
10 while 1:
11     (nouvelle_connexion, depuis) = ma_socket.accept()
12     pid = os.fork
13     if (not pid) :
14         socket_relais = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
15         socket_relais.connect(tsap_relais)
16         pid2 = os.fork()
17         if pid2 :
18             while 1:
19                 donnees = ma_socket.recv(1024)
20                 socket_relais.send(donnees)
21                 nouvelle_connexion.close()
22                 socket_relais.close()
23                 sys.exit()
24         else :
25             while 1:
26                 donnees = socket_relais.recv(1024)
27                 nouvelle_connexion.sendall(donnees)
28                 socket_relais.close()
29                 nouvelle_connexion.close()
30                 sys.exit()
31 ma_socket.close()

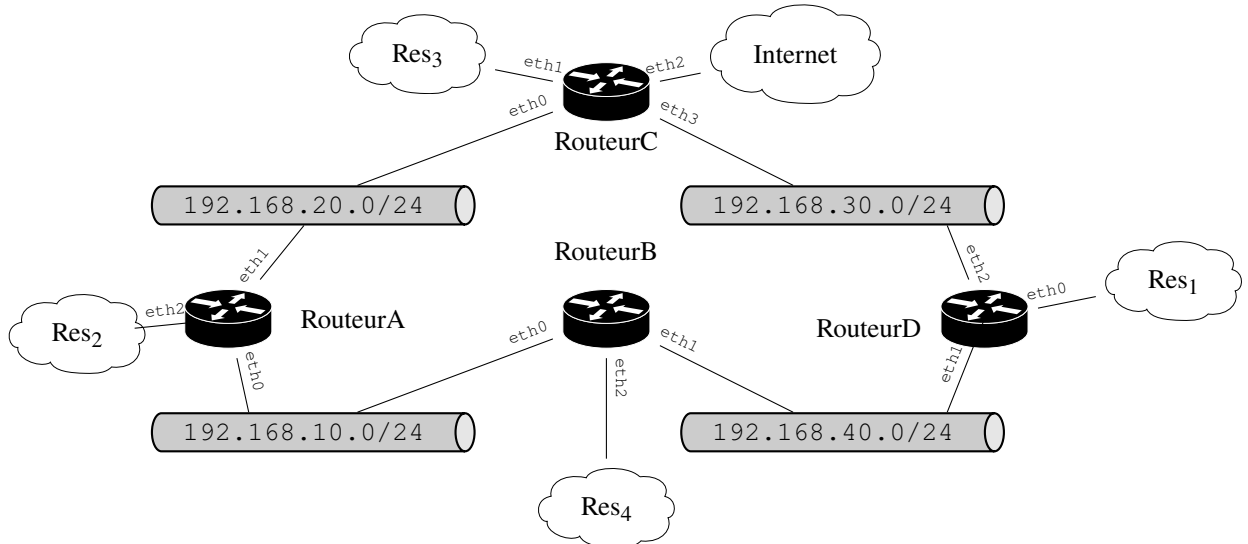
```

- Décrivez ce qu'il fait et corrigez les erreurs qui se sont glissées dedans. (3pts)



■ ■ ■ Table de routage — 7 points

3– Soit le réseau suivant :  
7pts



Les interfaces des routeurs sont configurées de la manière suivante :

	RouteurA	RouteurB	RouteurC	RouteurD
eth0	192.168.10.254/24	192.168.10.253/24	192.168.20.253/24	115.217.63.62/26
eth1	192.168.20.254/24	192.168.40.254/24	115.217.63.190/26	192.168.40.253/24
eth2	115.217.63.126/26	115.217.63.254/26	193.50.185.25/24	192.168.30.253/24
eth3	–	–	192.168.30.254/24	–

Routeur FAI : 193.50.185.254/24

- Donner l'adresse de chacun des réseaux Res1, Res2, Res3 et Res4. (1pt)
- Les machines de Res2 et Res1 réalisent des échanges **symétriques** en terme de quantité de trafic. (3pts)  
Donnez les tables de routage permettant de répartir la charge de ce trafic en deux chemins différents, c-à-d utilisant un **chemin différent** pour les paquets allant de Res1 vers Res2 de celui utilisé pour les paquets allant de Res2 vers Res1.  
*Vous devrez également permettre l'accès à Internet à l'ensemble des réseaux Res1, Res2, Res3 et Res4.*
- Est-il possible pour la machine M d'adresse 115.217.63.10/26 de Res1 d'être joignable depuis Res2 uniquement en passant par le même chemin que celui emprunté par les paquets allant de M vers Res2 (c-à-d différemment de la réponse à la question b) ? (2pts)  
Donnez les modifications de routage à apporter aux tables de routage précédentes.
- Peut-on empêcher les machines de Res4 d'accéder à Internet ? (1pt)

■ ■ ■ Fragmentation — 3 points

4– Un datagramme UDP est envoyé :

- 3pts
- ▷ d'un réseau A dont la MTU est de 1500 octets pour le contenu de la trame (c-à-d que le datagramme IP le plus grand qui peut y circuler a une taille de 1500 octets) ;
  - ▷ dans un réseau B dont la MTU est de 1000 octets pour le contenu de la trame (c-à-d que le datagramme IP le plus grand qui peut y circuler a une taille de 1000 octets) ;

Questions :

- Quel est la **taille maximale des données** que peut contenir un paquet UDP dans le réseau A ? (1pt)
- Comment va être **fragmenté** un paquet UDP de taille maximale à son arrivée dans le réseau B ? (2pts)  
*Vous donnerez la taille de chaque fragment ainsi que la valeur de chaque « offset ».*