Faculté des Sciences & Techniques

Université de Limoges

Master 1st year

Protocols & Network Prog.

Examination December 2022

Duration : 1h30 — Documents allowed

Network audit & Traffic analysis — 7 points

1 – Let the following frames be captured within the same local network :

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 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 00 01 50 12   #...(.........P.
0030   00 00 F4 80 00 00                                 ......
```

Questions :

a. What can you learn from the **content of these frames** about the configuration of the network, the hardware that communicates with each other and the services used ? *Justify your answers.* *(6pts)*

b. According to your analysis, **on which machine in the network** this capture may have taken place ? *(1pt)*

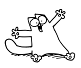
Network programming — 3 points

2 – Let the following program be :

3pts

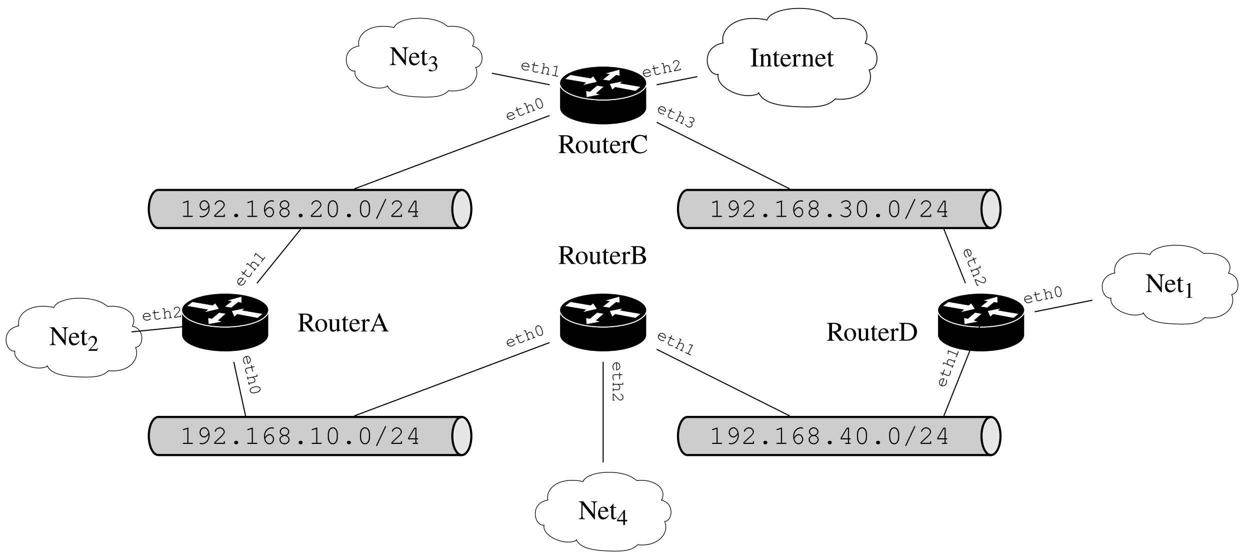
```
#!/usr/bin/python
import os,socket,sys

adresse_hote = ''
numero_port = 6800
tsap_relais = ('',6789)
ma_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM, socket.IPPROTO_TCP)
ma_socket.bind(adresse_hote, numero_port)

while 1:
    (nouvelle_connexion, depuis) = ma_socket.accept()
    pid = os.fork
    if (not pid) :
        socket_relais = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        socket_relais.connect(tsap_relais)
        pid2 = os.fork()
        if pid2 :
            while 1:
                donnees = ma_socket.recv(1024)
                socket_relais.send(donnees)
            nouvelle_connexion.close()
            socket_relais.close()
            sys.exit()
        else :
            while 1:
                donnees = socket_relais.recv(1024)
                nouvelle_connexion.sendall(donnees)
            socket_relais.close()
            nouvelle_connexion.close()
            sys.exit()
ma_socket.close()
```

☐ Describe what it does and correct any errors that have crept in. *(3pts)*



Routing table — 7 points

3 – Let be the following network :
7pts



The router interfaces are configured as follows :

	RouterA	RouterB	RouterC	RouterD
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	-

Router ISP : `193.50.185.254/24`

a. Give the address of each network Net1, Net2, Net3 et Net4. *(1pt)*

b. Net2 and Net1 machines exchange **symmetrically** in terms of traffic quantity. *(3pts)*
Give the routing tables allowing to distribute the load of this traffic in two different paths, i.e. using a different **path** for the packets going from Net1 to Net2 than the one used for the packets going from Net2 to Net1.
You will also need to allow Internet access to all Net1, Net2, Net3 and Net4 networks.

c. Is it possible for the machine M of address `115.217.63.10/26` of Net1 to be reachable from Net2 only by passing through the same path as the one taken by the packets going from M to Net2 (i.e. differently from the answer to question b) ? *(2pts)*
Give the routing changes to be made to the previous routing tables.

d. Can we prevent Net4 machines from accessing the Internet ? *(1pt)*

Fragmentation — 3 points

4 – A UDP datagram is sent :
3pts
▷ from a network *A* whose MTU is 1500 bytes for the content of the frame (i.e. the largest IP datagram that can circulate has a size of 1500 bytes) ;
▷ to a network *B* whose MTU is 1000 bytes for the content of the frame (i.e. the largest IP datagram that can circulate has a size of 1000 bytes) ;

Questions :

a. What is the **maximum data size** that a UDP packet can contain in the network *A* ? *(1pt)*

b. How will a UDP packet of maximum size be **fragmented** when it arrives in the network *B* ? *(2pts)*
You will give the size of each fragment as well as the value of each « offset ».