Universidad de León
Ingeniería Informática
Course on Computer Networks

Reference Solution to Homework On LAN and IP Switching

Details about this homework submission are published in the agora

***Notice**: The solutions included in this document are for reference only, consequently they do not represent any endorsed style of exam answer nor any exam rubric whatsoever.*

Exercises similar to those included in this homework assignment can be found in paloalto.unileon.es/cn under heading titled "Weekly Homework while on lockdown in year 2020 (For reference only))"

Exercise 1. The raw sockets that we have been using the past weeks provide access to the services offered by the datalink layer, in our case, Ethernet. In Linux, those sockets are created with the socket() syscall by passing it a protocol family of PF_PACKET; the type of socket can be selected from among SOCK_RAW and SOCK_DGRAM.

a. What's the technical name for the service interface that provides access to Internet Architecture layer 2, the IP layer? Is that *raw sockets,* also?

   `PF_INET/RAW_SOCK`

b. Type the C code that invokes the socket() syscall that will create a socket that accesses the IP layer interface.

```
//Raw socket for sending/receiving ICMP (Protocol 1)
int protocol = 1;
int sock = socket(PF_INET, RAW_SOCK, protocol);
```

c. Explain the meaning of the third actual parameter that you have passed to the socket call from the preceding question. Specifically, what's the relation of that 3rd parameter to the Internet Architecture? What kind of architectural element does this third parameter represent? Is it a layer? Is it a service? *What* is it?

   A `PF_INET/RAW_SOCK` socket is created upon the IP layer; the third formal parameter in the socket() syscall is **IP multiplexing key value** which will be used by the kernel for selecting the traffic that is delivered to the socket.

Exercise 2. You are to build a new implementation of the *Longest Prefix Matching* algorithm.

a. Which layer's interface would you have to access to receive the IP packets that your algorithm is to forward? In other words, what specific type of socket would you need to create? Write the C code that creates the correct socket.
A `PF_PACKET/SOCK_RAW` socket with protocol=0x0800 so that the IP packets would be delivered to the receiving application (The router's forwarding algorithm).

b. Discuss the consequences of sending frames which source address is not that of the sending interface.

The responses to those packets would be sent to the legitimate owner of the MAC address; otherwise, frames would be dropped.

Exercise 3. Consider the extended LAN in fig. 1. Solve the following exercises:

a. Develop the evolution of the forwarding tables of all the switches as the following transmissions take place:

1. Ha sends a frame to Hg

   B0 learns Ha. All switches flood this frame since they haven't learned Hg, yet; then all switches learn Ha

2. Ha sends a new frame to Hg

   B0 learns Ha. All switches flood this frame since they haven't learned Hg, yet; then all switches learn Ha

3. H_c sends a frame to H_a

   B2 learns Hc
   Since all switches already learned Ha, the frame travels from Hc to B2, then to B1, then to B0 and finally to Ha, that is, no flooding

4. H_e sends a frame to the broadcast address

   B3 learns He; since the dest MAC is ethernet broadcast, all switches flood the frame, consequently, all switches learn He

5. H_b sends a frame to the broadcast address

   B0 learns Hb; since the dest MAC is ethernet broadcast, all switches flood the *frame, consequently, all switches learn Hb*

6. H_d sends a frame to H_e

   B2 learns Hd; since all switches learned He in step 4, frame travels through this path: Hd -> B2 -> B1 ->B3 -> He

b. Host H_b sends a frame which SRC MAC is that of H_g. Explain how H_b can do this assuming that it is running a Linux stack and have the forwarding tables updated after the said frame sending by H_b.

· Use a PF_PACKET/SOCK_RAW socket
· Bridge B0 learns Hg at port 2. Switches standing on the path to destination learn Hg

c. Now, H_a sends a frame to H_g. Update the forwarding tables and explain which hosts receive that frame.

· Bridge B0 learns Ha.
· B0 forwards frame onto port 2 according to the learning done on the preceding step

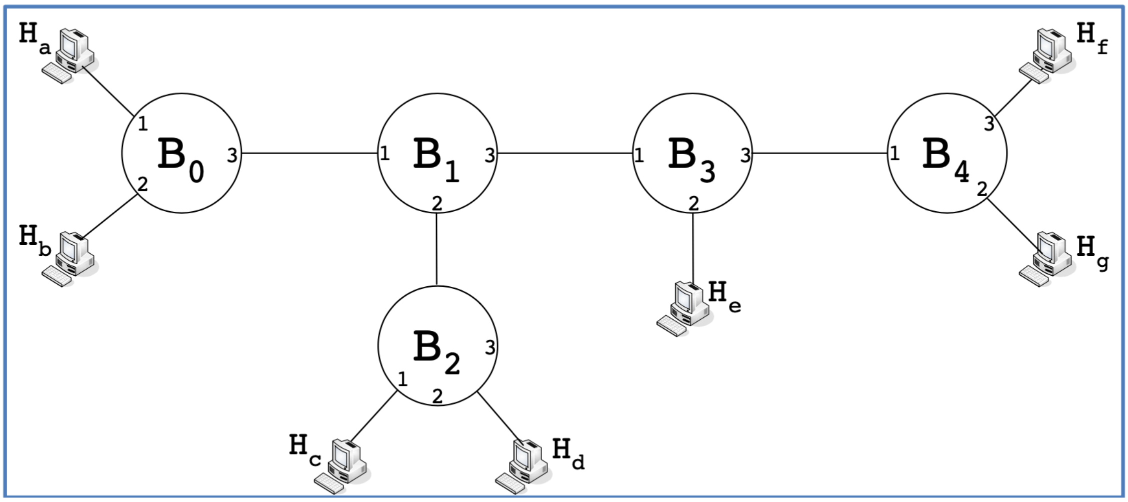
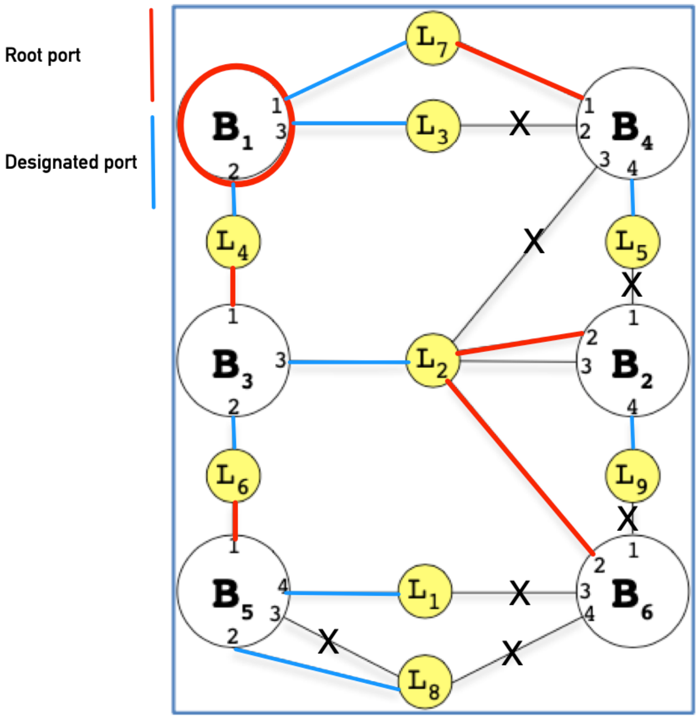


Figure 1. Extended LAN

Exercise 4. STP.

a. Obtain the *Spanning Tree* to the following Switched LAN comprised of STP Switches (B_i) and LANs (L_n).



b. Assume that the switches that comprise the extended LAN in fig. 2 don't implement STP. Devise a strategy for avoiding flooding storms in that Extended LAN altogether. Briefly, explain what changes to the LAN Switching algorithm should be necessary.

· Modify the LAN Switching algorithm so that broadcast frames are not flooded
· Modify the LAN switching algorithm so that unknown MACs don't cause flooding of the frame which MAC is unkown
· These modifications might need other modifications of the algorithm open for discussion

Exercise 5. EuroNet Associates needs to partition prefix P_a= 192.168.0.0/19 into 8 equal sized IP blocks for numbering their internal subnetworks.

a. What's the size of P_a?

```
8192
```

b. Obtain the IP block parameters of P_a= 192.168.0.0/19. You maywish to use the Linux ipcalc utility.

```
Network:    192.168.0.0/19
HostMin:  192.168.0.1
```

```
HostMax: 192.168.31.254
Broadcast: 192.168.31.255
```

c. Divide size(P_a)/8. The resulting size is the size of each of the 8 resulting subnets. What's the CIDR prefix of each resulting subnet?

8192 / 8 = 1024; CIDR prefix = 32 – $\log_2 1024$ = 32 -10 = **22**

d. What are the parameters of the first resulting IP block? (The initial IP address in prefix 192.168.0.0/19 is inherited by the first IP block resulting after the breakdown)

```
192.168.0.0/22
```

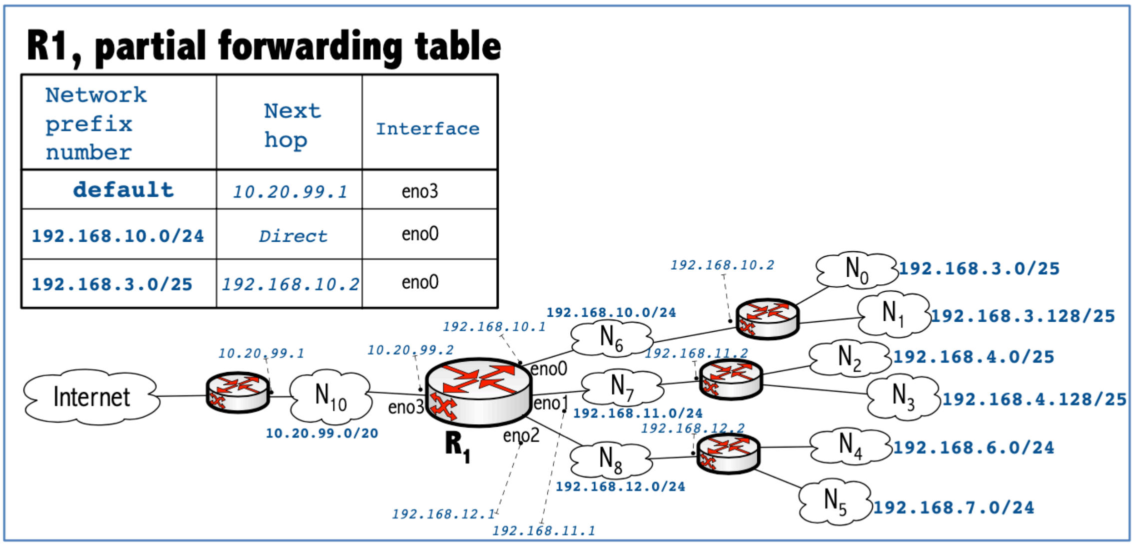
e. Calculate the parameters of all the remaining IP Blocks (#0 - #7)

```
Block #0 = 192.168.0.0/22
           192.168.3.255
Block #1 = 192.168.4.0/22
           192.168.7.255
Block #2 = 192.168.8.0/22
           192.168.11.255 …
```

Exercise 6. Can IP blocks 192.168.211.0/24 and 192.168.212.0/24 be joined into a single IP block that encompasses both without adding additional space? Discuss why and do the joining if possible.

Join of the two blocks is not possible since the resulting aggregate block has a CIDR prefix /23 and 192.168.211.0 is not divisible by the IP block size $2^{(32-23)} = 2^9$.

Exercise 7. Complete the forwarding table of Router R1 so that it can *efficiently* forward traffic to all networks included in the internetwork diagram in Figure 3.



R1, partial forwarding table

Network prefix number	Next hop	Interface
default	*10.20.99.1*	eno3
192.168.10.0/24	*Direct*	eno0
192.168.3.0/25	*192.168.10.2*	eno0

Figure 3. Internetwork

Prefix	Next gw	Interface
10.20.99.0/20	Direct	eno3
0.0.0.0	10.20.99.1	eno3
192.168.10.0/24	Direct	eno0
192.168.11.0/24	Direct	eno1
192.168.12.0/24	Direct	eno2
192.168.3.0/24	192.168.10.2	eno0
192.168.4.0/24	192.168.11.2	eno1
192.168.6.0/23	192.168.12.2	eno2

Table for solution to ex. 7

Exercise 8. Explain the trip of an IP packet that is transmitted from host 10.20.99.10 to host 192.168.12.12. Give meaningful symbolic names to the relevant MAC addresses, which are missing on the diagram, for example the MAC used by the transmitting host can be represented as MAC_{H1}.

Src host checks whether the destination IP 192.168.12.12 belongs to its own, directly-connected network (10.20.99.0/20); this is not the case since the Longest Prefix Matching algorithm responds that 10.20.99.0 doesn't match the local prefix 10.20.99.10. Src host send the IP packet to src host's default router (R_1) interface eno3; first, it requests the mac address of R_1 eno3 interface by sending an ARP Request for 10.22.99.2 (The default router). When the ARP reply is received, src host encapsulates the packet into an ethernet frame from itself to R1 eno3.

When R_1 receives the ethernet frame it deencapsulates its payload after checking its ethertype 0x0800; the IP packet is handed on to Longest Prefix Matching which will compute the best match of the dest IP 192.168.12.12 by looking it up in the above routing table. The longest prefix match is at the routing table entry 192.168.12.0/24, so the IP packet is encapsulated into a new ethernet frame to be sent onto direct interfaces eno2; dest mac is the dest host's mac which is discovered by ARP.

Exercise 9. An IP router has built the following forwarding table and at some instant it receives an IP packet P which destination IP is 10.20.76.100. Simulate the execution of the *Longest Prefix Matching* algorithm when the router is switching P. Clearly indicate the router interface selected for the retransmission of P. Assume no ordering whatsoever in the forwarding table entries.

Dest Network Prefix	Next hop	Interface
192.168.1.0/24	Direct	enp1s0
192.168.2.0/24	Direct	enp2s0
192.168.3.0/24	Direct	enp3s0
10.20.77.0/24	192.168.2.1	enp2s0
10.20.76.0/25	192.168.1.1	enp1s0
10.20.76.128/25	192.168.1.1	enp1s0
10.20.76.128/25	192.168.3.1	enp2s0
10.20.76.0/24	192.168.1.1	enp1s0
10.20.76.0/23	192.168.1.1	enp1s0

Table 1. Forwarding table

Matches for 10.20.76.100 in the forwarding table from Table 1:

- `10.20.76.0/23`
- `10.20.76.0/24`
- `10.20.76.0/25`

```
The longest matching prefix is 25, therefore the packet
will be forwarded onto enp1s0 via 192.168.1.1 according to
the forwarding table.
```

Exercise 10. Given the following Internet autonomous system based on the OSPF protocol, calculate the SPT (Shortest Path Tree) of router A by applying the Dijkstra's algorithm.

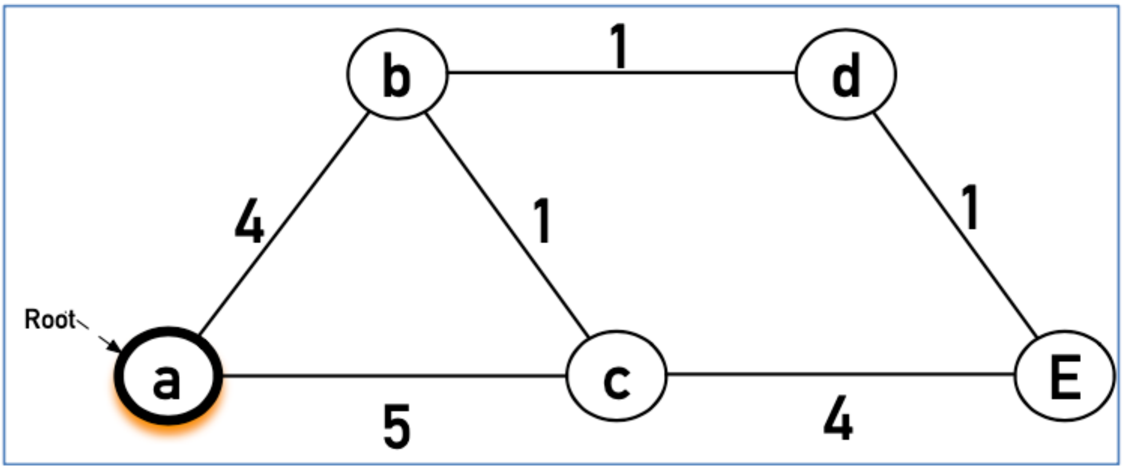
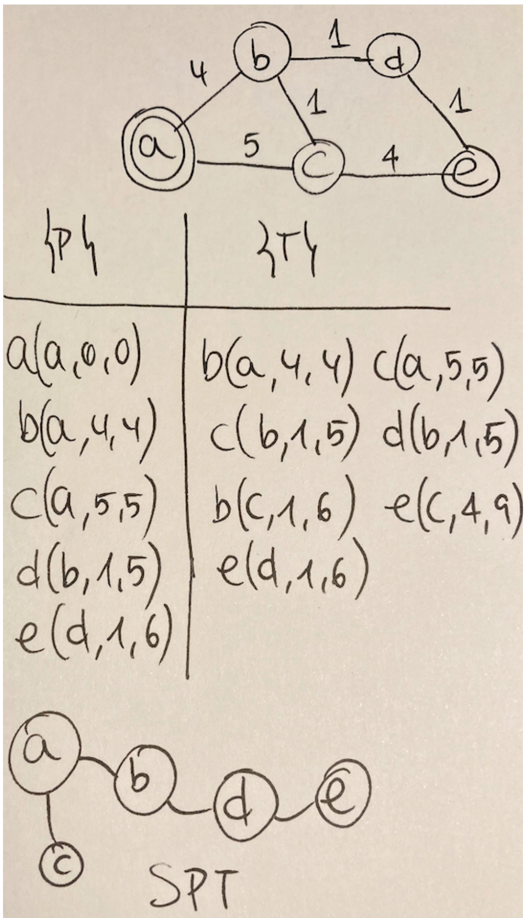


Figure 4. Internetwork



a
b
c
d
e
4
5
1
1
1
4
{P}
{T}
a(a,0,0)
b(a,4,4)
c(a,5,5)
d(b,1,5)
e(d,1,6)
b(a,4,4) c(a,5,5)
c(b,1,5) d(b,1,5)
b(c,1,6) e(c,4,9)
e(d,1,6)
SPT