Nathan Fox

CS 460

February 17, 2015

Link Layer Homework (Chapter 5)

1) P5

1011011100

|-----------

10011 | 10101010100000

10011

-----

11001

10011

-----

10100

10011

-----

11110

10011

-----

11010

10011

-----

10010

10011

-----

0100

R = 0100

2) P14

c. Sending an IP packet from E->B

1. E knows the IP address of the first-hop router on the path to B

2. E looks up the MAC address for the first-hop router using ARP

3. E creates a frame addressed to B's IP address

4. E sends the frame to the first-hop router (via MAC address) into subnet 3

5. The first router adapter sees that the link-layer frame is addressed to it and passes the frame to the network layer

6. The first router knows the IP address of the first-hop router on the path from itself to B (the second router on the path from E to B)

7. The first router looks up the MAC address for the second router using ARP

8. The first router creates a frame addressed to B's IP address

9. The first router sends the frame to the second router (via MAC address) into subnet 2

10. The second router adapter sees that the link-layer frame is addressed to it and passes the frame to the network layer

11. The second router consults its forwarding table, which tells it to forward the datagram via the interface to B's IP address

12. The second router uses ARP to look up B's MAC address

13. The interface passes the datagram to its adapter, which creates a new frame and sends the frame to B (via MAC address) into subnet 1

14. The switch in subnet 1 sees the MAC address and forwards the datagram to B

d. Sending an IP packet from E->B (where E has an empty ARP table)

1. E knows the IP address of the first-hop router on the path to B

2. E looks up the MAC address for the first-hop router using ARP

3. The ARP table is empty, so E broadcasts ARP query for the first-hop router

4. All adapters on subnet 3 receive ARP query

5. The first-hop router sends an ARP response to E (via E's MAC address, contained in the ARP query)

6. E creates a frame addressed to B's IP address

7. E sends the frame to the first-hop router (via MAC address) into subnet 3

8. The first router's adapter sees that the link-layer frame is addressed to it and passes the frame to the network layer

9. The first router gets the IP address of the first-hop router on the path from itself to B (the second router on the path from E to B) from its forwarding entry table

10. The first router looks up the MAC address for the second router using ARP

11. The first router creates a frame addressed to B's IP address

12. The first router sends the frame to the second router (via MAC address) into subnet 2

13. The second router adapter sees that the link-layer frame is addressed to it and passes the frame to the network layer

14. The second router consults its forwarding table, which tells it to forward the datagram via the interface to B's IP address

15. The second router uses ARP to look up B's MAC address

16. The interface passes the datagram to its adapter, which creates a new frame and sends the frame to B (via MAC address) into subnet 1

17. The switch in subnet 1 sees the MAC address and forwards the datagram to B

3) P15

a. E will not ask R1 for help. E will have F's IP address in its own forwarding entry table and will know that the "first hop" to F is F itself. E will use ARP to get F's MAC address. The ethernet frame's destination address will be F's MAC address, and the source address will be E's MAC address. The IP datagram that is carried will have a destination address (F's IP address) and a source address (E's IP address).

b. E will not perform an ARP query to find B's MAC address. E's forwarding entry table will tell it that the first hop on the path to B is through R1, so E will broadcast an ARP query for R1. In the ethernet frame, the destination address will be R1's MAC address. The source address will be E's MAC address. The IP datagram that is carried will have a destination address (B's IP address) and a source address (E's IP address).

c. When S1 receives the ARP request message, it will check to see if its IP address matches the destination IP address in the ARP packet. They will not match. S1 will look at the source IP and MAC addresses (both Host A's addresses) and will create an ARP entry for Host A. R1 will not receive the request, as R1 is not within the subnet, and the frame is only broadcast to the adapters on the subnet. Host B will not send an ARP query to ask for Host A's MAC address because the ARP request from Host A contained Host A's IP and MAC addresses, so Host B can create an ARP entry for Host A based on the information from Host A's ARP request. S1 will not receive the response message from Host B because Host B will use unicast to respond to Host A, and S1 is not on the path from Host B to Host A.

4) P27

a. (8L bits) / (128E3 bits/s) = (0.0625)L ms

b. L=1500 => 93.75 ms packetization delay

L=50 => 3.125 ms packetization delay

c. D\_trans = L/R = (1500 \* 8 bits) / (622E6 bits/s) = 19.29 microseconds

D\_trans = L/R = (50 \* 8 bits) / (622E6 bits/s) = 0.643 microseconds

d. One advantage of using a small packet size is that the packets are filled faster, decreasing packetization delay and avoiding the unpleasant echo that may be caused by higher packetization delays. Another advantage is that each packet can be transmitted faster, making the communication seem more seamless. Also, if our VoIP application is using UDP, packet loss is likely to occur, and if packets are smaller then occasional packet loss will be less noticeable.

5) P29

R4

-----------------------------------------------

| in label | out label | dest | out interface |

-----------------------------------------------

| 14 | 10 | A | 0 |

| 16 | 8 | A | 1 |

-----------------------------------------------

R5

-----------------------------------------------

| in label | out label | dest | out interface |

-----------------------------------------------

| | 16 | A | 0 |

-----------------------------------------------

R6

-----------------------------------------------

| in label | out label | dest | out interface |

-----------------------------------------------

| | 14 | A | 0 |

-----------------------------------------------

6) P32

a. 125 Mbps (80 flows going over a single 10 Gbps link => 125 Mbps per flow)

b. 1 Gbps (bottleneck is connection between TOR switch and host)

c. 62.5 Mbps (160 flows going over a single 10 Gbps link => 62.5 Mbps per flow)

500 Mbps (bottleneck is connection between TOR switch and Tier-2 switch, 20 flows going over a 10 Gbps link => 500 Mbps per flow)