CS 542 –
Computer Networks I: Fundamentals-
Assignment 2

Team members:

SL NO	Last name	First name	CWID
1	LNU	Naga Surya Suresh	A20492550
2	Patel	Shraddhaben	A20499171
3	Dhir	Aastha	A20468022

1. **(10 points)** A host with an IP address 110.12.33.19 and a physical address B2:34:55:10:22:10 has a packet to send to a host in another network. The destination IP and physical addresses are 131.83.57.21 and A5:6D:F3:59:83:AB, respectively. The next hop for this destination found in the sender's routing table is router R1 with an IP address 110.12.33.24 and a physical address B2:53:45:01:33:10. Show the ARP request and reply packets. Fill all the necessary fields. Ethernet and IPv4 protocols are implemented at the data link layer and the network layer, respectively.

ARP REQUEST FROM SOURCE: Broadcast

0x0001		0x0800
0x06	0x04	0x0001
0xb23455102210		
0x6e0c2113		
0x000000000000		
0x6e0c2118		

Preamble and SFD	0xFFFFFFFFFFFF	0xb23455102210	0x0806	Data	CRC

ARP RESPONSE FROM ROUTER: Unicast response

0x0001		0x0800
0x06	0x04	0x0002
0xb25345013310		
0x6e0c2118		
0xb23455102210		
0x6e0c2113		

Preamble and SFD	0xb23455102210	0xb25345013310	0x0806	Data	CRC

IN
ROUTER

ARP REQUEST FROM ROUTER: Broadcast

0x0001		0x0800
0x06	0x04	0x0001
0xb25345013310		
0x6e0c2118		
0x000000000000		
0x83533915		

Preamble and SFD	0xFFFFFFFFFFFF	0xb25345013310	0x0806	Data	CRC

ARP RESPONSE FROM DESTINATION: Unicast

0x0001		0x0800
0x06	0x04	0x0002
0xa56df35983ab		
0x83533915		
0xb25345013310		
0x6e0c2118		

Preamble and SFD	0xb25345013310	0xa56df35983ab	0x0806	Data	CRC

2. **(10 points)** Consider a freshly updated ARP cache table at time *t* (see below). The maximum number of attempts is 9, and the time-out is 500 seconds. The following packets were received:

   • at time *t*+30 s: an ARP reply from the host with IP address 113.4.6.89 and physical address 543742ACAE33

   • at time *t*+50 s: an ARP reply from the host with IP address 202.10.55.6 and physical address B46EF45983BC

   • at time *t*+80 s: an IP packet that must be forwarded to the next hop with IP address 187.11.8.71 and physical address 87AC859DE9B3

The cache table is updated every 70 seconds. Show the updated cache table at times *t*+70 seconds and *t*+140 seconds.

T

State	Queue	Attempt	Time-out	Protocol Address	Hardware Address
R	5		400	179.2.5.2	ACAE32457342
P	2	2		130.44.3.7	
P	14	6		202.10.55.6	
R	8		70	113.4.6.88	543742ACAE32
F					

An entry will be created in the cache table as no corresponding entry has been found. A queue 15 is assigned with "R" state and time-out set to 500(as mentioned). The updated cache table is shown below

T+30

State	Queue	Attempt	Time-out	Protocol Address	Hardware Address
R	5		400	179.2.5.2	ACAE32457342
P	2	2		130.44.3.7	
P	14	6		202.10.55.6	
R	8		70	113.4.6.88	543742ACAE32
R	15		500	113.4.6.89	543742ACAE33

An entry is found with state as P so, now the state is set to "R" and timeout is set to 500(as mentioned) and hardware address is updated. After this, the packets in the Queue 14 are sent one by one to the data link layer. The updated cache table is shown below

T+50

State	Queue	Attempt	Time-out	Protocol Address	Hardware Address
R	5		400	179.2.5.2	ACAE32457342
P	2	2		130.44.3.7	
R	14		500	202.10.55.6	B46EF45983BC
R	8		70	113.4.6.88	543742ACAE32
R	15		500	113.4.6.89	543742ACAE33

After 70s the cache table is updated as below. It checks every entry and updates the value accordingly. For the first entry, it decreases the timeout by 70s. for the second entry the attempt is increased by 1 and an ARP request is sent to the data link layer. For the third entry, time-out is decreased by (70-50) = 20s since the timeout was updated only 20s ago. For the 4th entry the timeout is decreased by 70s, since the time-out is zero, the state is set to Free, the other entries are deleted, and the queue is cleared. For the last entry, the time-out is reduced by 70-30 = 40s. The updated cache table is shown below

T+70

State	Queue	Attempt	Time-out	Protocol Address	Hardware Address
R	5		330	179.2.5.2	ACAE32457342
P	2	3		130.44.3.7	
R	14		480	202.10.55.6	B46EF45983BC
F					
R	15		460	113.4.6.89	543742ACAE33

Since the IP packet already has the resolved hardware address with it, it is directly forwarded to the data link layer for the transmission. Cache table is not modified, and it remains the same.
When the new IP packet arrives, it checks the table since no entry is found, it adds a new entry with the pending state, this is because an IP packet cannot accommodate a mac address within its header.
T+80

State	Queue	Attempt	Time-out	Protocol Address	Hardware Address
R	5		330	179.2.5.2	ACAE32457342
P	2	3		130.44.3.7	
R	14		480	202.10.55.6	B46EF45983BC
P	19	1		187.11.8.71	
R	15		460	113.4.6.89	543742ACAE33

After 140s the cache table is updated as below. It checks every entry and updates the value accordingly. For the first entry, it decreases the timeout by 70s. for the second entry the attempt is increased by 1 and an ARP request is sent to the data link layer. For the third entry, time-out is decreased by 70s. For the last entry, the time-out is reduced by 70s. The updated cache table is shown below

T+140

State	Queue	Attempt	Time-out	Protocol Address	Hardware Address
R	5		260	179.2.5.2	ACAE32457342
P	2	4		130.44.3.7	
R	14		410	202.10.55.6	B46EF45983BC
P	19	2		187.11.8.71	
R	15		390	113.4.6.89	543742ACAE33

3. **(4 points)** Are the following values of the HLEN field (in the IP datagram header) valid or invalid? Explain your answer.
   (a) 1111 (binary format)
   Since the HLEN is a 4-bit field, max length value can be 4*15 = 60 bytes which is **valid**.
   (b) 14 (hexadecimal format)
   Hex to dec = 16*1+ 1*4 = 20 since the maximum value of a HLEN 4-bit field is 15 in decimal, this is **not valid**.
   (c) E (hexadecimal format)
   Hex to Dec = E = 14, the binary equivalent for this hex is 1110 and is in the range of HLEN, it is a **valid** field.
   (d) 0101 (binary format)
   HLEN value for given bit is (5(0101) *4) 20 byte which is minimum valid header length, hence this is **valid**.

4. **(2 points)** The value of the HLEN field of an IP packet is B_{16}, and the value of the total length field is 0040_{16}. How many bytes of data is this packet carrying? Are there any options? If so, what is the length of the options in bytes?

   The HLEN value is $B_{16} = 11_{10}$, which means the total number of bytes in the header is 11 × 4 or 44 bytes.
   Therefore, Base header = 20 bytes and Option value = 44-20 = 24 bytes.
   The total length is 0040_{16} = 16*4+0 = 64 bytes, which means the packet is carrying 20 bytes of data (40 − 20).
   Data = Total length – HLEN*4 = 64 – 11*4 = 64 -44 = 20 bytes.

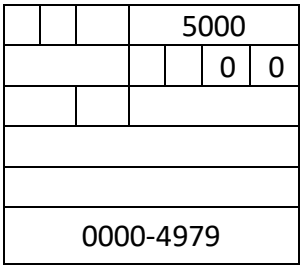
5. **(2 points)** The total length of an IP datagram is 70 bytes, out of which 34 bytes are data. Is this a valid IP datagram? Explain your answer.

   Total length = 70 bytes.
   Data value= 34 bytes.
   Header value = 70 – 34 = 36 bytes. Therefore HLEN = 36/4 = 9 = 1001 which is a valid datagram, header since it is between 20 – 60 bytes.

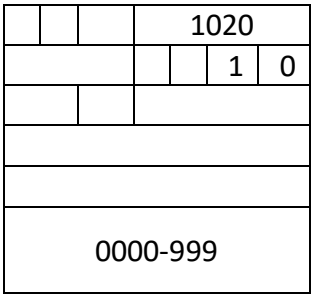
6. **(3 points)** An IP fragment has arrived with the first few hexadecimal digits as follows:
   4500 003C 0051 2064 .....
   This is the second fragment of an original IP datagram. How many bytes of data does this fragment contain? Is there the next fragment? If so, what is its offset? Explain your answer.
   HLEN = 5 = 5*4 = 20 (no options) bytes
   Total length = 3*16+12*1 = 48+12 = 60 bytes
   Data = 60-20 = 40 bytes.
   This fragment contains 40 bytes of data.
   Identification = 0051
   2064 = 0010 0000 0110 0100 = Flag (3 bit) + fragmentation offset(13-bit)
   Do not fragment bit = 0
   More bit = 1, **this is not the last fragment.**
   Therefore, there is more fragments.
   Offset of 2nd fragment = 0010 0000 0110 0100 = 100.
   Data of the first fragment = 100*8 = 800 bytes. Therefore, the range is 0…799 bytes.
   So, the data range of 2nd fragment = 800 … (800+40-1) = 800…839.
   Therefore, the next fragment's offset value is 840/8 = 105.

7. (4 points) Consider fragmenting an original IP datagram of size 5000 bytes with a base header only. The offset of the second fragment is 125. Answer the following questions:

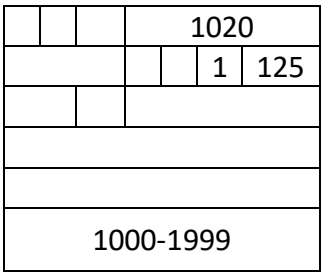
Original



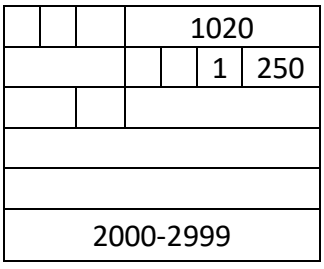
First



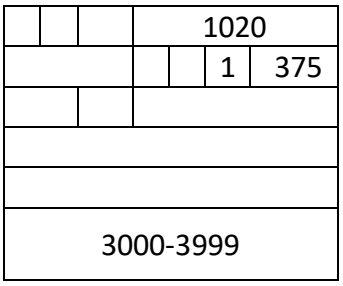
Second



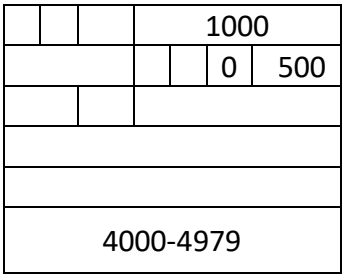
Third.



Forth



Fifth



(a) What is the number of fragments? (Assume that all fragments except the last one are equal.)
There are 5 fragments.

(b) What is the size of the 2nd IP fragment?
1000 bytes ranging from 1000-1999 bytes

(c) What is the size of the last IP fragment?
4980 – 4000 = 980 bytes

(d) What is the fragmentation offset of the 3rd IP fragment?
Offset value of the 3rd fragment is 250.

8. **(2 points)** An IP packet has arrived with the first few hexadecimal digits as follows:

4600 0040 0301 0000 1106 .....

The initial "*Time to live"* value in the hexadecimal format was 1A.

How many hops has this packet already traveled?

Given: Initial TTL value = 1A = 26

To find the time-to-live field from given hex input, we skip 8 bytes (16 hexadecimal digits). The time-to-live field is the ninth byte, which is $11 = 17_{10}$.

Therefore, the packet has travelled 26 – 17 = 9 times.

How many hops can this packet still travel before being dropped?

It can still travel 17 times before it is dropped .

9. **(2 points)** A client has received a UDP datagram. The corresponding port number has been found in the control block table but there is no queue number. Is it possible? Why?

Yes, this is possible since a process can be started by the client. As there were no User datagrams arrived before this UDP datagram, no queue has been assigned. Now, when the UDP datagram has arrived, a new queue will be allocated by the input module.

10. **(5 points)** The following TCP header dump is given in the hexadecimal format:

03451071 00000331 00000026 501007EE 01300000

(a) What is the source port number?

0345 = $3*16^2 + 4*16 + 5*1 = 837$

(b) What is the sequence number?

00000331 = $3*16^2 + 3*16 + 1*1 = 817$

(c) What is the header length?

5 = 5*1 = 5 in 4 words

5 * 4 = 20 bytes.

(d) Which flags are set? What does it mean?

010 = 0000 0001 0000

0000 0001 0000 = reserved

0000 0001 0000 = Flags. Only ACK flag is enabled. Which implies that this has a valid acknowledgement sequence number.

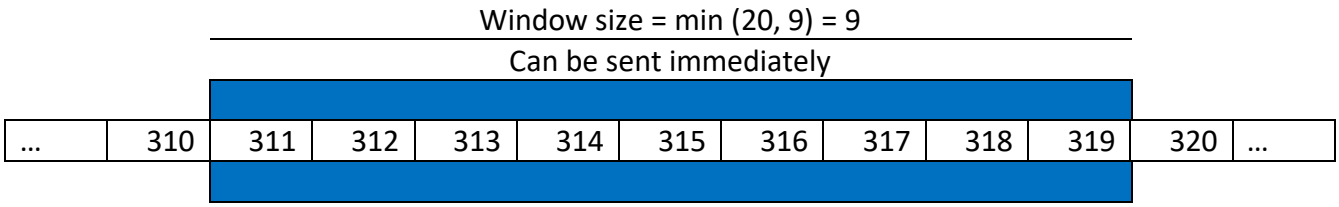
(e) What is the window size?

07EE = $7*16^2 + 14*16 + 14*1 = 2030$

Give all your answers in the decimal format.

11. **(4 points)** The current *cwnd*=9 and *rwnd*=20. The last acknowledgment received is 311. Draw the diagram showing the current TCP window. A new TCP segment has just arrived with an acknowledgment number of 315 and *rwnd*=*x*. What is the minimum value of *x* to avoid shrinking the window?

Since the last acknowledgement received was 311, all the segments from start to 310 inclusive has been received. Below is the diagram showing the current window.



new rwnd ≥ (last ack + last rwnd) − new ack

new rwnd = x ≥ (311 + 9) – 315

new rwnd = x ≥ 5

12. **(3 points)** A header of a UDP datagram is

E323 00BC 00AB E217 in the hexadecimal format.

(a) What is the source port number?

E323 = $14*16^3 + 3*16^2 + 2*16 + 3*1 = 58147$

(b) What is the destination port number?

00BC = 11*16 + 12*1 = 188

(c) What is the total length of this UDP datagram?

00AB = 10*16 + 11*1 = 171

Give all your answers in the decimal format.

13. **(2 points)** Consider an ARP packet with the *"Operation"* field set to 1 and the *"Target hardware address"* field filled with a valid physical address. Is it a valid ARP packet? Explain your answer.

ARP associates an IP address with its physical address. Anytime a host, or a router, needs to find the physical address of another host or router on its network it sends an ARP query packet. As per the given information if operation field is 1 so Packet is ARP Request packet and Target hardware address field is filled with valid physical address. Host or router will use ARP request packet only in case if it doesn't know the target hardware address. So, it cannot be a Valid ARP Request Packet.