Practical-2

**AIM:** Understand and identify header fields of layers of TCP/IP protocol stack.

**Tools Required**: WordPad or Notepad.

**Submission**: After writing the answer into this word document, Student needs to change name to his ID followed by a practical number. Ex 20ce005\_Pr1.docx. Upload on assignment segment.

**Rubrics**: Nicely drafted document with clarity in answers leads to full marks. Otherwise, submission carries a proportional mark.

Watch and refer following videos for a better understanding of the header fields of layers of TCP/IP:

Material 1. **Ethernet frame ():** https://[www.youtube.com/watch?v=SoTRqDLND6Y](http://www.youtube.com/watch?v=SoTRqDLND6Y) Material 2. **IPv4 header format ()**: https://[www.youtube.com/watch?v=3Y70y6dM7Cs](http://www.youtube.com/watch?v=3Y70y6dM7Cs) Material 3. **IPv4 Vs IPv6()**: https://[www.youtube.com/watch?v=NkE9\_iRPi1I](http://www.youtube.com/watch?v=NkE9_iRPi1I)

Material 4. **TCP and UDP ():** https://[www.youtube.com/watch?v=r4HbLQuqvrM](http://www.youtube.com/watch?v=r4HbLQuqvrM)

Students need to fill the empty table and write answers to questions.

As per the discussion in classroom, any user starts internet access through browser or network applications. Following figure 2.1 explain scenario of receiving data at NIC Card. NIC card receives signals and it converts into sequence of 0’s and 1‘s. After receiving data it sends data for the further processing to TCP/IP protocol stack. In this exercise you need to identify boundaries of fields of headers, describe and understand flow of information in protocol stack.

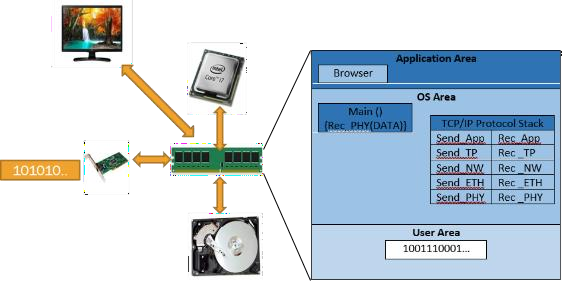


Figure 2.1 Real Scenario

# Input data stream: TCP

This is the data stream which receiver NIC card receives from wire and stores into memory. Length of bits stream is 432 bits.

00000000000110101000110001101011011101101010110011101000110110001101000101000110

11110011111100010000100000000000010001010000000000000000001010000001010110111101

01000000000000001000000000000110000000000000000010101100000100000000110001111011

10001110111110101011011101001110110001010111001100000001101110110100100111101111

01111101000110101111001000001000000110100010011101010000000100000001000000001010

11111110111011100000000000000000

Abstract view of data with respect to the location of headers and data in the actual data stream.

|  |  |  |  |
| --- | --- | --- | --- |
| Data Link (Ethernet)  Header | Network  Header | Transport  Header | Data |

Initial 112 bits contains Ethernet Header (Refer section 2.2), Next 160 bits contains IP Header (Refer section 2.3), Next 160 bits contains TCP Header (Refer section 2.4).

# Header format of Ethernet

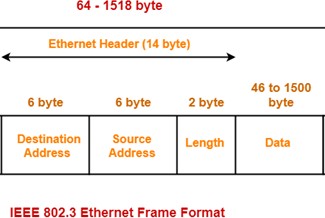


Figure 2.2 Ethernet Header Format

Section 2.1 contains bit stream. copy and paste respected number of bits into following table 2.1 to prepare ethernet header field boundary.

Table 2.1 Header format of ethernet

|  |  |  |
| --- | --- | --- |
| 00000000000110101000110001101011  0111011010101100 | 11101000110110001101000101000110  1111001111110001 | 0000100000  000000 |

From table 2.1, fill table 2.2 with respected value and explanation meaning of each field. Refer the following link for better understanding. Refer video 1 in material 1 for further understanding.

Reference Link : <https://en.wikipedia.org/wiki/Ethernet_frame#Header> <https://en.wikipedia.org/wiki/EtherType>

Table 2.2 Header fields of Ethernet

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field**  **Name** | **Length of**  **Field (in bits)** | **Header field Value**  **(Hex Value)** | **Meaning** |
| Destination  MAC Address | 48 bits | 00:1A:8C:6B:76:AC | Receiver’s MAC address |
| Source MAC  Address | 48 bits | E8:D8:D1:46:F3F1 | Sender’s MAC address |
| Type | 16 bits | 0x800 | 0x800 indicates, Network Header type is  IPv4 Header |

# Header format of Network

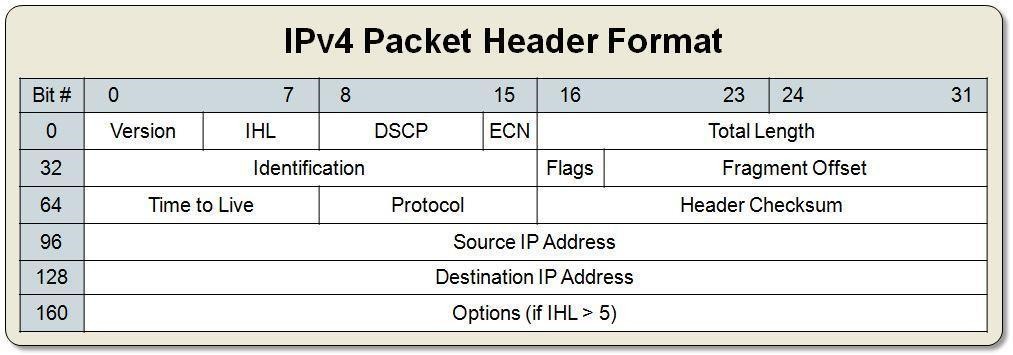


Figure 2.3 IPv4 header format

Section 2.1 contains bit stream. copy and paste respected number of bits into following table 2.3 to prepare ethernet header field boundary.

Table 2.3 Header format of network

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0100 | 0101 | 000000 | 00 | 0000000000101000 | |
| 0001010110111101 | | | | 010 | 0000000000000 |
| 10000000 | | 00000110 | | 0000000000000000 | |
| 10101100000100000000110001111011 | | | | | |
| 10001110111110101011011101001110 | | | | | |

From table 2.3, fill table 2.4 with respected value and explanation meaning of each field. Refer the following link for better understanding. Refer video 2 in material 2 for further understanding.

Reference Links: <https://en.wikipedia.org/wiki/IPv4#Header>

DCSP & ECN: <https://en.wikipedia.org/wiki/Type_of_service#DSCP_and_ECN>

Flags: <https://en.wikipedia.org/wiki/IPv4#Flags>

Protocol: <https://en.wikipedia.org/wiki/List_of_IP_protocol_numbers>

Table 2.4 Header fields of Network

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field**  **Name** | **Length of**  **Field (in bits)** | **Header field**  **Value (Hex Value)** | **Meaning** |
| Version | 4 bits | 0x4 | IP Datagram version 4 |
| IHL | 4 bites | 0x5 | 5\*32bits=160bits=20bytes |
| DSCP | 6 bits | 0x0 | --- |
| ECN | 2 bits | 0x0 | --- |
| Total length | 16 bits | 0x28 | Total length of 40 bytes |
| Identification | 16 bits | 0x15BD |  |
| flags | 3 bits | 0x2 | 2 bit More Fragment (MF) |
| Fragment  offset | 13 bits | 0x0 | This packet does not contain fragments. |
| Time to live | 8 bits | 0x80 | 128 Hops / Routers |
| Protocol | 8 bits | 0x06 | This packet should be give to TCP receive  procedure. As its value indicates TCP. |
| Header  checksum | 16 bits | 0x0 | No checksum included in this header. |
| Source IP  Address | 32 bits | 172.16.12.123 | Source IP: 172.16.12.123, its local machine |
| Destination IP  Address | 32 bits | 142.250.183.78 | Destination: 142.250.183.78, it is situated  in country |

# Header format of transport layer: TCP

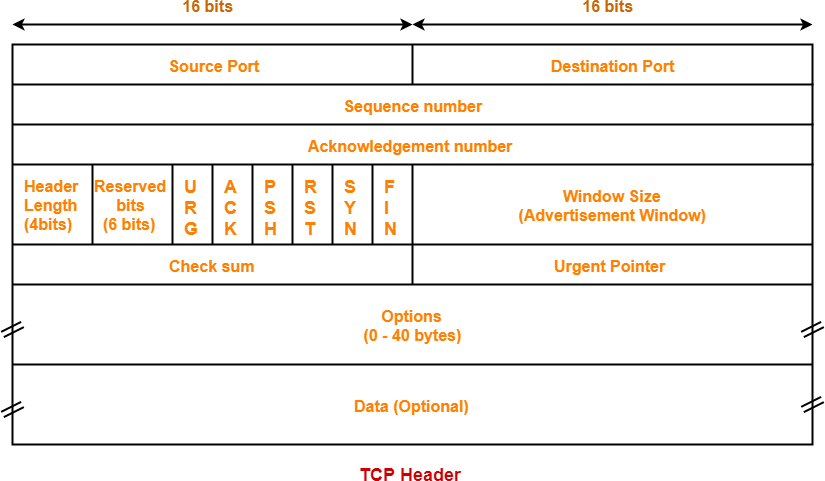


Figure 2.4 TCP Header format

Section 2.1 contains bit stream. copy and paste respected number of bits into following table 2.5 to prepare ethernet header field boundary.

Table 2.5 Header fields of transport layer

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1100010101110011 | | | | | | | | 0000000110111011 |
| 01001001111011110111110100011010 | | | | | | | | |
| 11110010000010000001101000100111 | | | | | | | | |
| 0101 | 000000 | 0 | 1 | 0 | 0 | 0 | 0 | 0001000000001010 |
| 1111111011101110 | | | | | | | | 0000000000000000 |

From table 2.5, fill table 2.6 with respected value and explanation meaning of each field. Refer the following link for better understanding. Refer video 4 in material 4 for further understanding.

Reference Link : <https://en.wikipedia.org/wiki/Transmission_Control_Protocol#TCP_segment_structure> Flags: <https://www.gatevidyalay.com/transmission-control-protocol-tcp-header/>

Port : https://[www.adminsub.net/tcp-udp-port-finder/](http://www.adminsub.net/tcp-udp-port-finder/)

Table 2.6 Header fields of Transport Layer:

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in**  **bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| Source Port | 16 Bits | 50547 | Sender machine’s application’s logical port  number 50547. |
| Destination Port | 16 Bits | 443 | Receiver machine’s receiving logical port  number 443 which indicates source wants communicate security using https protocol. |
| Sequence Number | 32 Bits | 0x49EF7D1A | Unique ID assigned by sender to maintain  order of packers at receiver side. |
| Acknowledgemen  t Number | 32 Bits | 0xF2081A27 | This is acknowledge of sent packet. |
| Header Length | 4 Bits | 0x5 | Total header Length is  5\*32bits=160bits=20bytes |
| Reserved Bits | 6 Bits | 0x0 | - |
| URG | 1 Bit | 0 |  |
| ACK | 1 Bit | 1 | This packet contains valid  acknowledgement number. |
| PSH | 1 Bit | 0 | No Push |
| RST | 1 Bit | 0 | No RST |
| SYN | 1 Bit | 0 | No SYN |
| FIN | 1 Bit | 0 | NO Fin |
| Window Size | 16 Bits | 0x100A | 4106 |
| Checksum | 16 Bits | 0xFEEE | Error identification in packet. |
| Urgent Pointer | 16 Bits | 0x0 | No urgent content in this packet. |

# Exercise-1: Input Sequence TCP

11101000110110001101000101000110111100111111000100000000000110101000110001101011

01110110101011000000100000000000010001010000000000000000001010001011111111010010

01000000000000000100000000000110110011101101000001100111001110111000110001100110

10101100000100000000110001111011000000011011101111110010110010000011000100010000

00000101111000001110010000000000010000101001101001010000000100000000000011111011

10110000100111010000000000000000000000000000000000000000000000000000000000000000

# Header fields of Ethernet

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| Destination  MAC Address | 48 bits | E8:D8:D1:46:F  3:F1 | Receiver's MAC address |
| Source MAC  Address | 48 bits | 00:1A:8C:6B:7  6:AC | Senders MAC address |
| Type | 16 bits | 800 | 0x800 indicates, Network Header type isIPv4  Header |

**Header fields of Network**

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| VERSION | 4bit | 4 | IP Datagram version 4 |
| IHL | 4bit | 5 | 5\*32bits=160bits=20bytes |
| DSCP | 6bit | 0 | ------------ |
| ECN | 2bit | 0 | ------------ |
| TOTAL LENGHT | 16bit | 28 | Total length of 40 bytes |
| IDENTIFICATIO  N | 16bit | BFD2 |  |
| FLAGS | 3bit | 2 | 2 bit More Fragment (MF) |
| FRAGMENT  OFFSET | 13bit | 0 | ------- |
| TIME TO LIVE | 8bit | 40 |  |
| PROTOCOL | 8bit | 6 | This packet should be given to TCP receive  procedure. As its value indicates TCP. |
| HEADER CHECK  SUM | 16bit | CED0 |  |
| SOURCE IP  ADDRESS | 32bit | 673B8C66 | 259.89.320.258 |
| DESTINATION  ADDRESS | 32bit | AC100C7B | 172.16.12.123 |

# Header fields of Transport Layer:

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| Source Port | 16 Bits | 01BB | Sender machine’s application’s logical port  number 01BB. |
| Destination Port | 16 Bits | F2C8 | Receiver machine’s receiving logical port  number F2C8 which indicates source wants communicate security using https protocol. |
| Sequence  Number | 32 Bits | 311005E0 | Unique ID assigned by sender to maintain  order of packers at receiver side. |
| Acknowledgem  ent Number | 32 Bits | E400429A | This is acknowledge of sent packet. |
| Header Length | 4 Bits | 0x5 | Total header Length is  5\*32bits=160bits=20bytes |
| Reserved Bits | 6 Bits | 0x0 | - |
| URG | 1 Bit | 0 |  |
| ACK | 1 Bit | 1 | This packet contains valid acknowledgement  number. |
| PSH | 1 Bit | 0 | No Push |
| RST | 1 Bit | 0 | No RST |
| SYN | 1 Bit | 0 | No SYN |
| FIN | 1 Bit | 0 | NO Fin |
| Window Size | 16 Bits | 0xFB | 4106 |
| Checksum | 16 Bits | B09D | Error identification in packet. |
| Urgent Pointer | 16 Bits | 0x0 | No urgent content in this packet. |

**Exercise-2: Input Sequence of UDP**

11101000110110001101000101000110111100111111000100000000010100000101011010101011

11110010011001110000100000000000010001010000000000000000001010000010000101011001

00000000000000001000000000010001101010011000100110101100000100000000101101000111

10101100000100000000110001111011111001110010000000001101001111010000000000010100

10001101001111111111010010000000000001011110011100000100000000000000000000000100

00000000000000010000111111011001000000000000000000000000000000000000000000000000

# Header fields of Ethernet

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| Destination  MAC Address | 48 bits | E8:D8:D1:46:  F3:F1 | Receiver's MAC address |
| Source MAC  Address | 48bits | 50:56:AB:F2:6  7 | Sender's MAC address |
| Type | 16bits | 0x800 | 800 indicates IPV4 |

**Header fields of Network**

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| VERSION | 4 | 4 | IP Datagram version 4 |
| IHL | 4 | 5 | 5\*32bits=160bits=20bytes |
| DSCP | 6 | 0 | ------------ |
| ECN | 2 | 0 | ------------ |
| TOTAL LENGHT | 16 | 28 | Total length of 40 bytes |
| IDENTIFICATIO  N | 16 | 2159 |  |
| FLAGS | 3 | 0 | 2 bit More Fragment (MF) |
| FRAGMENT  OFFSET | 13 | 0 | ------- |
| TIME TO LIVE | 8 | 80 |  |
| PROTOCOL | 8 | 11 | This packet should be give to TCP receive  procedure. As its value indicates TCP. |
| HEADER CHECK  SUM | 16 | A989 |  |
| SOURCE IP  ADDRESS | 32 | AC100B47 | 172.16.11.71 |
| DESTINATION  ADDRESS | 32 | AC100C7B | 172.16.12.123 |

# Header fields of Transport Layer:

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| Destination MAC  Address | 48 | E7200D3D00 14 | Receiver's MAC address |
| Source MAC  Address | 48 | 8D3FF48005 E7 | Sender's MAC address |
| Type | 16 | 400 |  |

**Exercise-3: Input Sequence: ARP Broadcast**

11111111111111111111111111111111111111111111111111100000011000111101101001010100

01110001010001000000100000000110000000000000000100001000000000000000011000000100

00000000000000011110000001100011110110100101010001110001010001001010110000010000

00001000001000010000000000000000000000000000000000000000000000001010110000010000

00000111100111010000000000000000000000000000000000000000000000000000000000000000

00000000000000000000000000000000000000000000000000000000000000000000000000000000

# Header fields of Ethernet

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| Destination  MAC Address | 48 bits | FFFFFFFFFFFF | Receiver's MAC address |
| Source MAC Address | 48 bits | E063DA5471 4  4 | Sender's MAC address |
| Type | 16bits | 806 |  |

**Header fields of**

|  |  |  |  |
| --- | --- | --- | --- |
| **Header field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| VERSION | 4 | 0 | ------ |
| IHL | 4 | 0 | ------- |
| DSCP | 6 | 0 | ------ |
| ECN | 2 | 1 |  |
| TOTAL LENGHT | 16 | 800 |  |
| IDENTIFICATIO  N | 16 | 604 |  |
| FLAGS | 3 | 0 | ------- |
| FRAGMENT  OFFSET | 13 | 1 |  |
| TIME TO LIVE | 8 | E0 |  |
| PROTOCOL | 8 | 63 |  |
| HEADER CHECK  SUM | 16 | DA54 |  |
| SOURCE IP  ADDRESS | 32 | 7144AC10 | 113.68.172.16 |
| DESTINATION  ADDRESS | 32 | 8210000 | 8.33.0.0 |

# Exercise-4: Input Sequence: ARP Reply

11111111111111111111111111111111111111111111111101001100000100011011111110011101

11110011100010110000100000000110000000000000000100001000000000000000011000000100

00000000000000100100110000010001101111111001110111110011100010111010110000010000

00001110011010110000000000000000000000000000000000000000000000001010110000010000

00001110011010110000000000000000000000000000000000000000000000000000000000000000

00000000000000000000000000000000000000000000000000000000000000000000000000000000

# Header fields of Ethernet

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| Destination  MAC Address | 48 bits | FFFFFFFFFFFF | Receiver's MAC address |
| Source MAC  Address | 48 bits | 4C11BF9DF3  8B | Sender's MAC address |
| Type | 16bits | 806 |  |

**Header fields of**

|  |  |  |  |
| --- | --- | --- | --- |
| **Header Field Name** | **Length of Field (in bits)** | **Header field Value (Hex**  **Value)** | **Meaning** |
| VERSION | 4 | 0 | ------ |
| IHL | 4 | 0 | ------ |
| DSCP | 6 | 0 | ------ |
| ECN | 2 | 1 |  |
| TOTAL LENGHT | 16 | 800 |  |
| IDENTIFICATIO  N | 16 | 604 |  |
| FLAGS | 3 | 0 | ------ |
| FRAGMENT  OFFSET | 13 | 2 |  |
| TIME TO LIVE | 8 | 4C |  |
| PROTOCOL | 8 | 11 |  |
| HEADER CHECK  SUM | 16 | BF9D |  |
| SOURCE IP  ADDRESS | 32 | F38BAC10 | 243.139.172.16 |
| DESTINATION  ADDRESS | 32 | E6B0000 | 14.107.0.0 |

# Questions and answers:

1. What do you mean by TTL (Time to Live)?

Answer: TTL (Time to Live) in networking refers to a value in IP packets that dictates the maximum time or number of hops a packet can traverse in a network before being discarded.

1. What is the significance of Sequence Number and Acknowledgment Number in TCP format? Answer: The sequence number in TCP format is essential for ordering and reconstructing data packets, while the acknowledgment number confirms the receipt of data and facilitates reliable data transmission between sender and receiver.
2. What is the full form of the MAC address? What is the significance of source and destination MAC address?

Answer: The full form of MAC address is Media Access Control address, and the source MAC address identifies the sender of a network packet while the destination MAC address specifies the intended recipient, crucial for data forwarding at the data link layer.

1. What is the full form of IP, TCP, UDP and ARP? Answer: IP: Internet Protocol

TCP: Transmission Control Protocol UDP: User Datagram Protocol

ARP: Address Resolution Protocol

**Gate Questions:**

# What is the maximum size of data that the application layer can pass on to the TCP layer below?

* 1. Any size
  2. 216 bytes - size of TCP header
  3. 216 bytes
  4. 1500 bytes

# The protocol data unit (PDU) for the application layer in the Internet stack is:

* 1. Segment
  2. Datagram C) Message

D) Frame

# A TCP message consisting of 2100 bytes is passed to IP for delivery across two networks. The first network can carry a maximum payload of 1200 bytes per frame and the second network can carry a maximum payload of 400 bytes per frame, excluding network overhead. Assume that IP overhead per packet is 20 bytes. What is the total IP overhead in the second network for this transmission?

* 1. 40 bytes
  2. 80 bytes
  3. 120 bytes D) 160 bytes

# Which one of the following statements is FALSE?

* 1. TCP guarantees a minimum communication rate
  2. TCP ensures in-order delivery
  3. TCP reacts to congestion by reducing sender window size
  4. TCP employs retransmission to compensate for packet loss

# In TCP, a unique sequence number is assigned to each

* 1. byte
  2. word
  3. segment
  4. message

# Consider the following statements about the timeout value used in TCP.

1. **The timeout value is set to the RTT (Round Trip Time) measured during TCP connection establishment for the entire duration of the connection.**

# Appropriate RTT estimation algorithm is used to set the timeout value of a TCP connection.

1. **Timeout value is set to twice the propagation delay from the sender to the receiver. Which of the following choices hold?**
   1. (i) is false, but (ii) and (iii) are true
   2. (i) and (iii) are false, but (ii) is true
   3. (i) and (ii) are false, but (iii) is true
   4. (i), (ii) and (iii) are false

# Consider an IP packet with a length of 4,500 bytes that includes a 20-byte IPv4 header ans 40- byte TCP header. The packet is forwarded to an IPv4 router that supports a Maximum Transmission Unit (MTU) of 600 bytes. Assume that the length of the IP header in all the outgoing fragments of this packet is 20 bytes. Assume that the fragmentation offset value stored in the first fragment is 0.

**The fragmentation offset value stored in the third fragment is . Note –This was ..Numerical Type question.**

* 1. 0
  2. 72

C) 144

D) 216

# Consider two hosts P and Q connected through a router R. The maximum transfer unit (MTU) value of the link between P and R is 1500 bytes, and between R and Q is 820 bytes.

**A TCP segment of size 1400 bytes was transferred from P to Q through R, with IP identification value as 0×1234. Assume that the IP header size is 20 bytes. Further, the packet is allowed to be fragmented, i.e., Don’t Fragment (DF) flag in the IP header is not set by P.**

# Which of the following statements is/are correct?

* 1. Two fragments are created at R and the IP datagram size carrying the second fragment is 620 bytes.
  2. If the second fragment is lost, R will resend the fragment with the IP identification value 0×1234.
  3. If the second fragment is lost, P is required to resend the whole TCP segment.
  4. TCP destination port can be determined by analysing only the second fragment.

# One of the header fields in an IP datagram is the Time to Live(TTL)field.Which of the following statements best explains the need for this field?

* 1. It can be used to prioritize packets
  2. It can be used to reduce delays
  3. It can be used to optimize throughput D) It can be used to prevent packet looping

# In an IPv4 datagram, the M bit is 0, the value of HLEN is 10, the value of total length is 400 and the fragment offset value is 300. The position of the datagram, the sequence numbers of the first and the last bytes of the payload, respectively are:

* 1. Last fragment, 2400 and 2789
  2. First fragment, 2400 and 2759 C) Last fragment, 2400 and 2759

D) Middle fragment, 300 and 689

# The maximum number of IPv4 router addresses that can be listed in the record route (RR) option field of an IPv4 header is 9.

1. **Consider an IP packet with a length of 4, 500 bytes that includes a 20 − byte IPv4 header ans 40**

# - byte TCP header. The packet is forwarded to an IPv4 router that supports a Maximum

**Transmission Unit (MTU) of 600 bytes. Assume that the length of the IP header in all the**

# outgoing fragments of this packet is 20 bytes. Assume that the fragmentation offset value stored in the first fragment is 0.

**The fragmentation offset value stored in the third fragment is 119.**

# For which one of the following reasons does internet protocol(IP) use the time-to-live(TTL) field in IP datagram header?

* 1. Ensure packets reach destination within that time
  2. Discard packets that reach later than that time
  3. Prevent packets from looping indefinitely
  4. Limit the time for which a packet gets queued in intermediate routers

# One of the header fields in an IP datagram is the Time-to-Live (TTL) field. Which of the following statements best explains the need for this field?

* 1. It can be used to prioritize packets.
  2. It can be used to reduce delays.
  3. It can be used to optimize throughput. D) It can be used to prevent packet looping.

# Host A (on TCP/IP v4 network A) sends an IP datagram D to host B (also on TCP/IP v4 network B). Assume that no error occurred during the transmission of D. When D reaches B, which of the following IP header field(s) may be different from that of the original datagram D?

1. **TTL**

# Checksum

1. **Fragment Offset**
   1. i only
   2. i and ii only C) ii and iii only

D) i, ii and iii

# Host A sends a UDP datagram containing 8880 bytes of user data to host B over an Ethernet LAN. Ethernet frames may carry data up to 1500 bytes (i.e. MTU = 1500 bytes). Size of UDP header is 8 bytes and size of IP header is 20 bytes. There is no option field in IP header. How many total number of IP fragments will be transmitted and what will be the contents of offset field in the last fragment?

* 1. 6 and 925 B) 6 and 7400

1. 7 and 1110
2. 7 and 8880

# An IP datagram of size 1000 bytes arrives at a router. The router has to forward this packet on a link whose MTU (maximum transmission unit) is 100 bytes . Assume that the size of the IP header is 20 bytes .

**The number of fragments that the IP datagram will be divided into for transmission is 11.**

# In the TCP/IP protocol suite, which one of the following is NOT part of the IP header?

* 1. Fragment Offset
  2. Source IP address
  3. Destination IP address D) Destination port number