Assignment : 01

Name : Binod Kumar

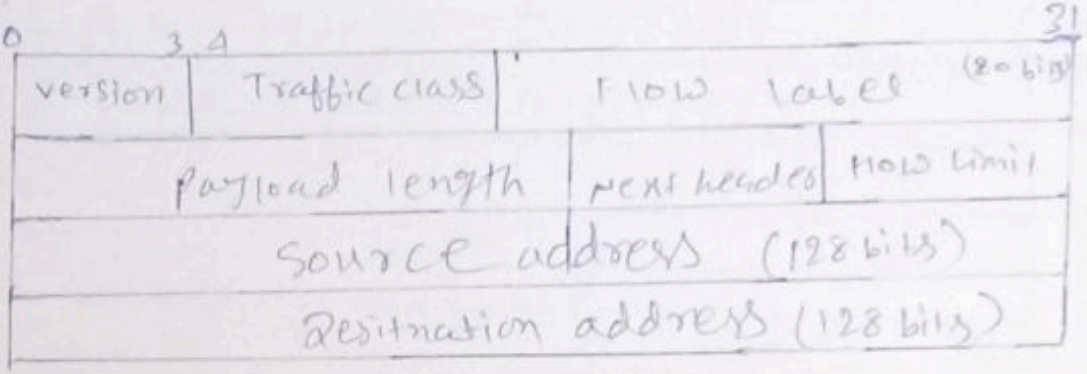
Roll no : 23203006

Class : M.Tech CSE

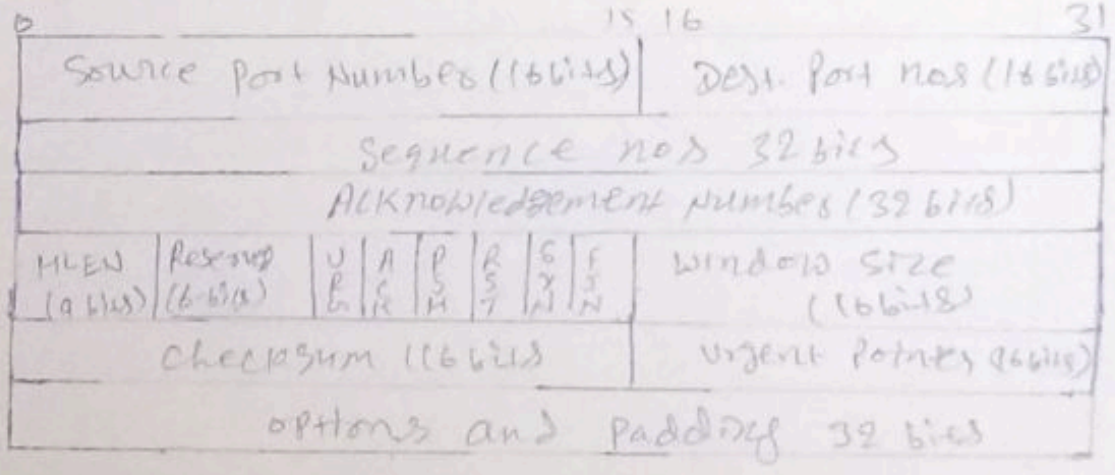
Subject : Advanced Computer Network

Q7) An IPV6 packet consists of the base header and a TCP segment. The length of data is 320 bytes. Show the packet and enter a value for each field

Ans: As we know the IPV6 packet formate as

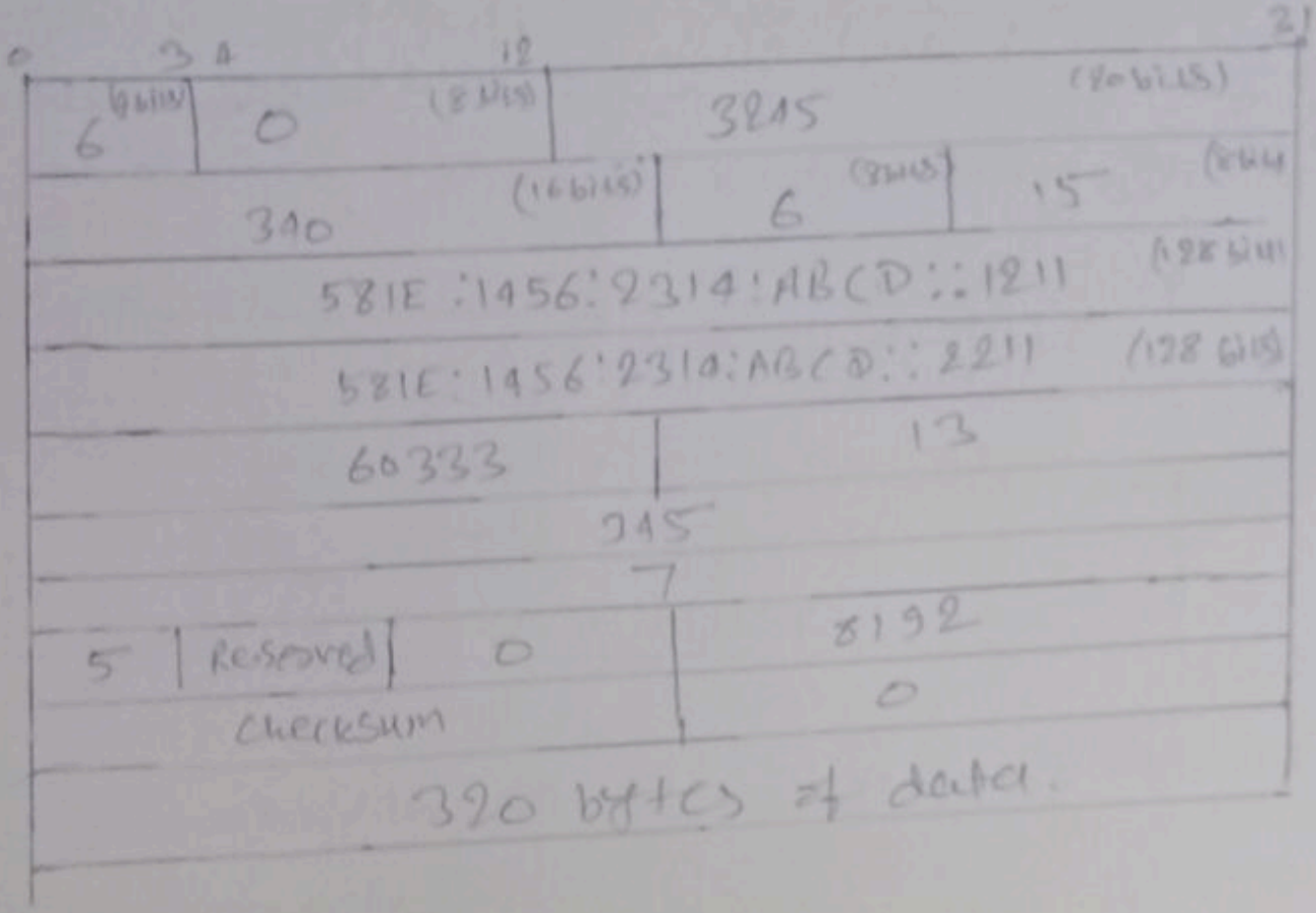


Header formate of TCP as below



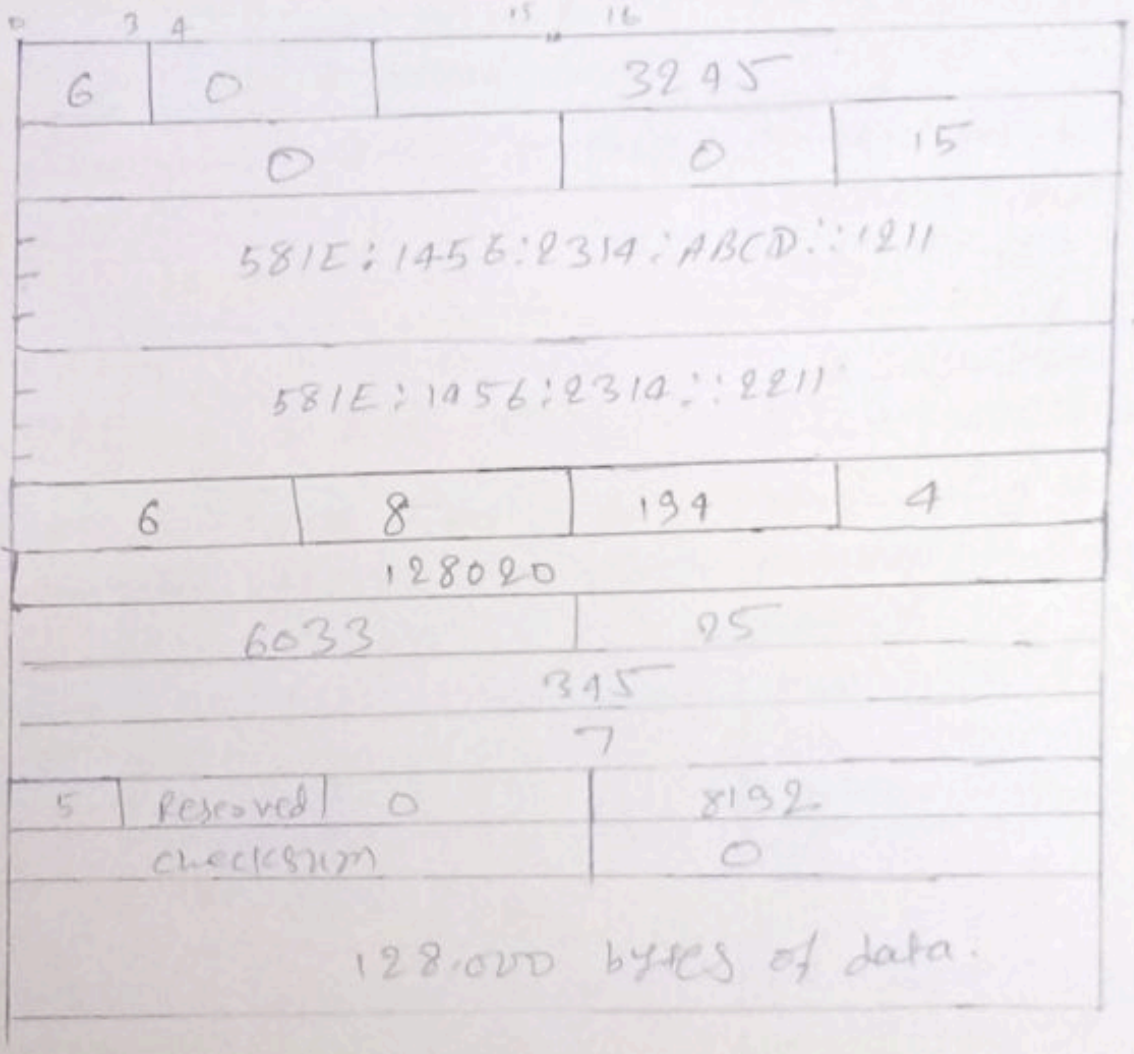
Now, I will show the IPV6 packet that consists of the base header and TCP segment where the length of data is 320 bytes.

- version : version of IP packet which is 6 (IPV6). or (1110).
- source address and destination address of 128 bits. We can write in Hexadecimal

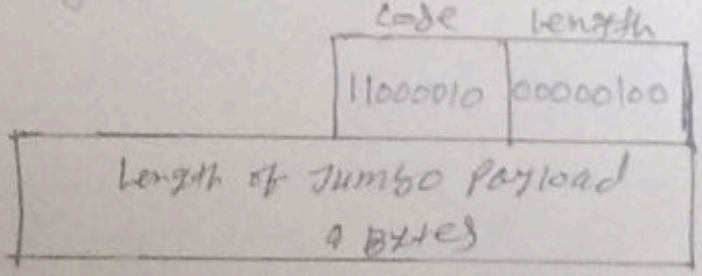

0
3 4
12
31
6
0
3215
340
6
15
581E:1456:2314:ABCD::1211
581E:1456:2314:ABCD::2211
60333
13
345
7
5
Reserved
0
8192
0
Checksum
320 bytes of data.

8.25 An IPV6 packet consists of a base header and a TCP segment. The length of data is 128,000 bytes (Jumbo payload). Show the packet and enter a value for each field.

Ans:



The length of the payload in the IP datagram can be a maximum of 65,535 bytes. However, if for any reason a longer payload is required, we can use the Jumbo payload option to define this longer length.



Q-3) Make a table to compare and contrast inverse neighbor discovery message in V4.

Ans: Inverse Neighbor Discovery (IND) - IPV4 Vs IPV6.

Feature	IPV4 (ARP: Address Resolution Protocol	IPV6 (IND)
Protocol	Separate protocol	Extension of Neighbors discovery protocol (NDP)
Message Type	ARP Request & ARP Reply	IND Solicitation & IND Advertisement
Address Resolution	MAC address for a known IPV4 address	IPV6 address for a known link-layer address
Unicast vs Multicast	Unicast to target device	Multicast to all neighbours on the link.
Standardization	RFC 826	RFC 3122
Target specificity	Specific target device	Neighbouring devices on the link.
Efficiency	Less efficient - Requires separate message.	More efficient - use existing NDP infrastructure
Scalability	Less scalable with large network	More scalable due to multicast nature

8.4) Write the IPV6 address of the following IPV4 address.

a) 192.168.99.1

Soln:- Convert decimal value of each octet in IPV4 into Hexadecimal.

$(192)_{10} = (C0)_{16}$

$(168)_{10} = (A8)_{16}$

$(99)_{10} = 63$

$(1)_{10} = (01)_{16}$

		remainders
16	192	0
	12	C

↑ Bottom-up

16	168	8
	10	A

So, the IPV4 address of 192.168.99.1 represented in the IPV6 address portion would be C0A8:6301

b) 138.14.85.210

$(138)_{10} = (8A)_{16}$

$(14)_{10} = (0E)_{16}$

$(85)_{10} = (55)_{16}$

$(210)_{10} = (D2)_{16}$

16	138	10=A
	8	

138=(8A)

16	14	14=E
	0	

(14) = (0E)

So, the IPV6 address portion would be 8A0E:55D2

Q.5) Write the IPV4 address of the following IPV6 address.

a) 8a0e:55d2

Soln: 8A0E : 55 D2 Hex

8bit 8bit

↓ convert into Decimal from Hex.

$16^0 \times (A=10) + 16^1 \times 8$

$: 1 \times 10 + 128$

$(8A)_{16} = (138)_{10}$

$(0E)_{16} = 16^0 \times (E=14) + (16^1 \times 0) = (14)_{10}$

$(55)_{16} = 16^0 \times 5 + 16^1 \times 5 = 5 + 85 = (90)_{10}$

$(D2)_{16} = 16^0 \times 2 + (D=13) \times 16 = 2 + 208 = (210)_{10}$

∴ IPV4 : 138.14.90.210

b) C0A8:6301

$(C0)_{16} = 16^0 \times 0 + 16^1 \times (12) = (192)_{10}$

$(A8)_{16} = 16^0 \times 8 + 16 \times (10) = (168)_{10}$

$(63)_{16} = 16^0 \times 3 + 16^1 \times 6 = (99)_{10}$

$(01)_{16} = 16^0 \times 1 + 16^1 \times 0 = 1$

∴ IPV4 address : 192.168.99.1

Q.6) Explain the compressed format for writing IPV6 address and write the following IPV6 address into their shortest compressed form.

Ans:

Zero compression.

i) Double colon (::)

- It can replace any single, contiguous string of one or more 16 bit segments consisting of all 0s.
- It can only be used once per IPV6 address.

ii) Leading zeros.

- Any leading 0s in any 16-bit section or hextet can be omitted.

  Eg 0 0A → 10A

a) 2001:0DB8:0000:1470:0000:0000:0000:0200

Shortest compression.

2001:DB8:0:1470::200

b) F380:0000:0000:0000:0123:4567:89AB:CDEF

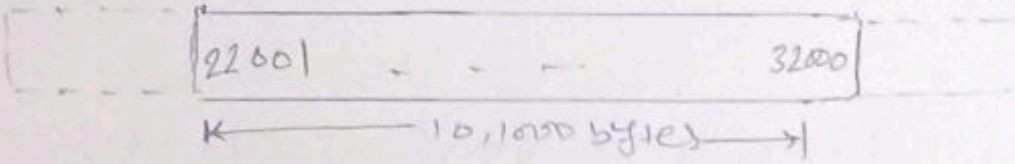
Shortest form

F380::123:4567:89AB:CDEF

A.7) TCP connection is using a window size of 10,000 bytes and the previous acknowledgment number was 22,001. It receives a segment with Acknowledgment number 24,001 and window size advertisement of 12,000. Draw a diagram to show the situation of the window before and after.

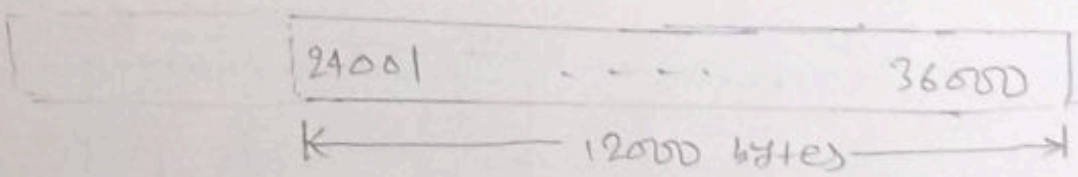
Soln

22001 + 10000 - 1
= 32000



Window Before Sliding

- Previous ack 22001, ie The receiver's next expected sequence number.



Window after sliding

Q.8) A window hold bytes 2001 to 5000. The next byte to be sent is 3001. Draw a figure to show the situation of the window after the following two events.

a) An ACK segment with the acknowledgement number 2500 and window size advertisement 4000 is received.

b) A segment carrying 1,000 byte is sent.

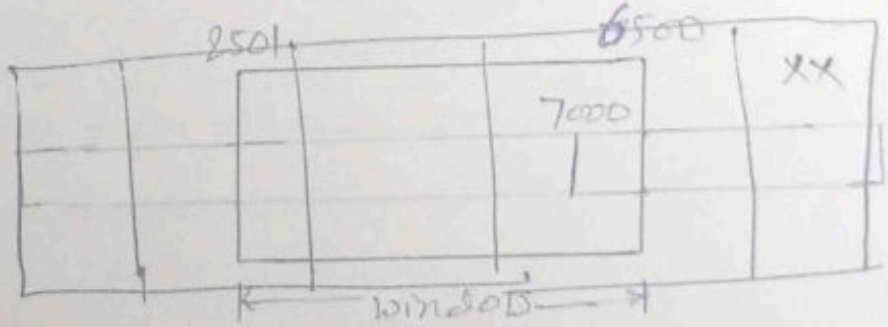
Soln

- Initial window hold 2001 to 500 bytes.
- ACK segment with acknowledgement no. 2500 window size advertisement ~~5000~~ 4000

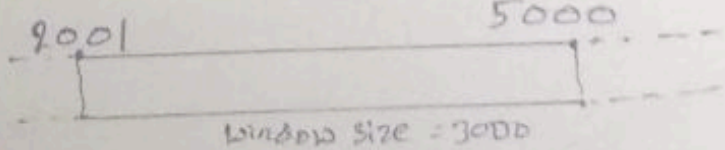
So, window hold

2501 to 2501 + 4000 - 1

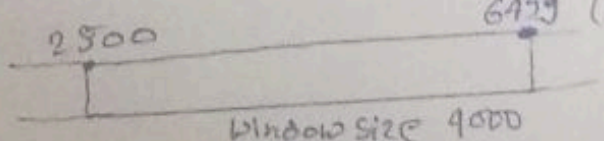
i.e 2501 to 6500



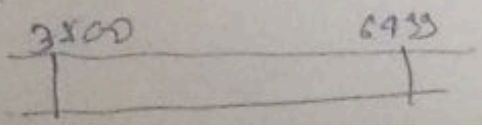
- Initially



- After ACK = 2500 i.e next byte to receive 2500 and window size = 4000



- After 1000 byte sent



Q.9) An SCTP client opens an association using an initial tag of 806, an initial TSN of 14534 and a window size of 20000. The server responds with an initial tag of 2000, an initial TSN of 670, and a window size of 14000. Show the contents of all four packets exchanged during association establishment. Ignore the value of the cookie.

Soln:

1. INIT (from client)
   a) Source Port: Assigned by OS.
   b) Destination Port: Assigned by application
   c) Verification tag: 806
   d) Advertised Receiver window credit: 20000
   e) 23203103
   f) Number of outbound streams: As needed
   g) Initial TSN: 14534
   h) Checksum: calculated

2. INIT-ACK (from server)
   a) Source Port: Assigned by OS
   b) Destination Port: Assigned by application
   c) Verification Tag: Randomly chosen by server
   d) Initiation Tag: 2000
   e) Advertized Receiver window credit: 14000
   f) Nos of outbound streams: As needed.
   g) Initial TSN: 670
   h) Checksum: calculated.

3. COOKIE-ECHO (from client):

a. Source Port: Assigned by OS
b. Destination Port: Assigned by application
c. Verification Tag: 806
d. Initial Tag: 806
e) Advertised Receiver Window Credit: 20000
f) Nos of outbound streams: As needed
g) Initial TSN: 19534
h) Cookie: Ignored

4. COOKIE-ACK (from server):

a. Source Port: Assigned by OS
b) Destination Port: Assigned by application
c) Verification Tag: 2000
d) Checksum: Calculated.