Computer Networks (Dec.2017/Jan 2018)

Solutions

Module-1

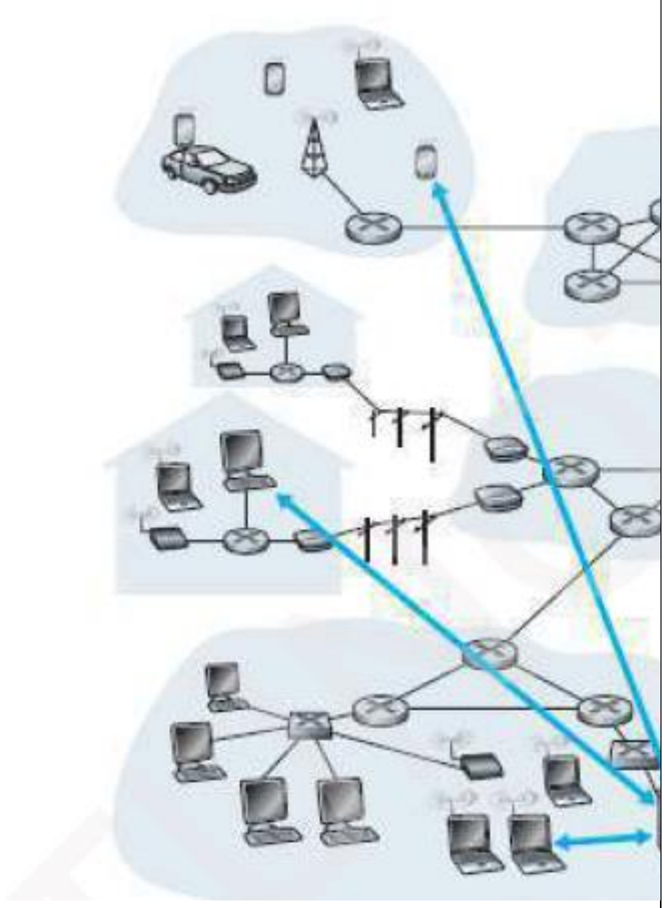
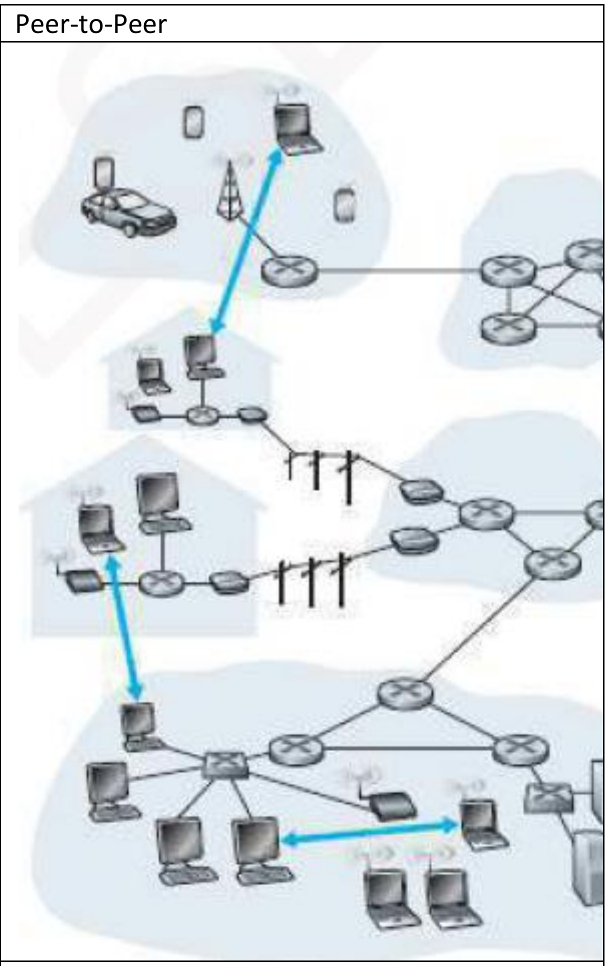
1. a) Compare Client server and Peer-to-Peer architecture (05)

b)Describe HTTP with Persistent and non-persistent connections (08)

c) What are the services provided by DNS? (03)

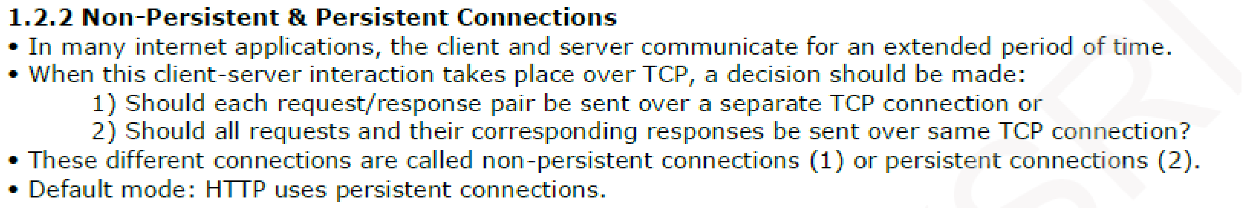
Answer:

a) Comparison between Client server and Peer-to-Peer architecture

Client server	Peer-to-Peer
	
There will be always-on dedicated Server with fixed IP Address	No dedicated server will be there
Client has to initiate the communication with server	But here as no dedicated server will be there peer can ask any other peer to provide service, then the asking peer will be the client and asked peer will be the server
Clients can't communicate with each other directly	Peers can communicate with each other directly
FTP, Telnet,Email uses the client server architechture	BitTorrent, Skype uses peer to peer architecture

There is no self scalability feature	Self scalability is an important feature

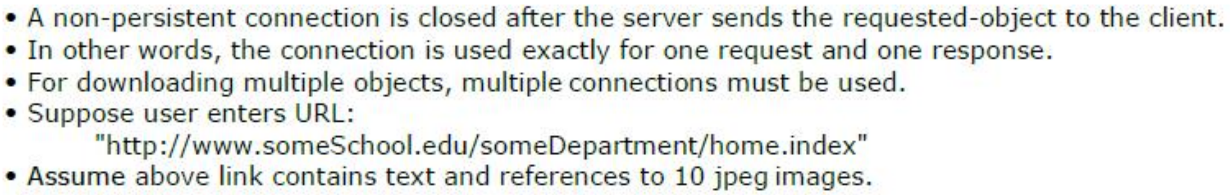
b)



1.2.2 Non-Persistent & Persistent Connections

- In many internet applications, the client and server communicate for an extended period of time.
- When this client-server interaction takes place over TCP, a decision should be made:
  1) Should each request/response pair be sent over a separate TCP connection or
  2) Should all requests and their corresponding responses be sent over same TCP connection?
- These different connections are called non-persistent connections (1) or persistent connections (2).
- Default mode: HTTP uses persistent connections.

Http with non persistent connection



- A non-persistent connection is closed after the server sends the requested-object to the client.
- In other words, the connection is used exactly for one request and one response.
- For downloading multiple objects, multiple connections must be used.
- Suppose user enters URL:
  "http://www.someSchool.edu/someDepartment/home.index"
- Assume above link contains text and references to 10 jpeg images.

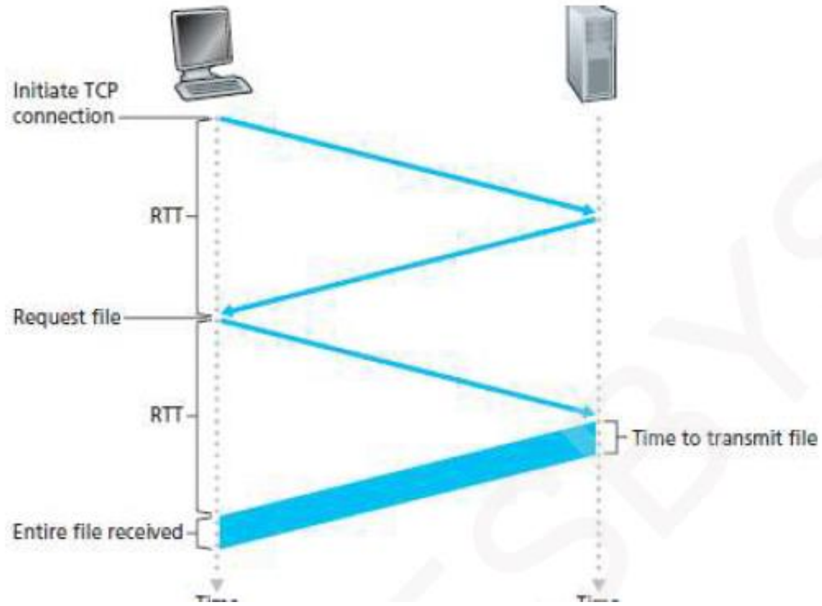
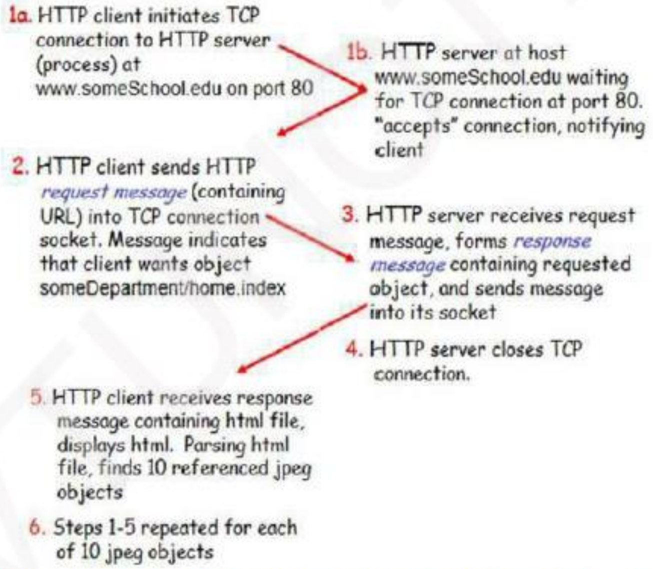


Figure 1.4: Back-of-the-envelope calculation for the time needed to request and receive an HTML file

- Here is how it works:



1a. HTTP client initiates TCP connection to HTTP server (process) at www.someSchool.edu on port 80

1b. HTTP server at host www.someSchool.edu waiting for TCP connection at port 80. "accepts" connection, notifying client

2. HTTP client sends HTTP *request message* (containing URL) into TCP connection socket. Message indicates that client wants object someDepartment/home.index

3. HTTP server receives request message, forms *response message* containing requested object, and sends message into its socket

4. HTTP server closes TCP connection.

5. HTTP client receives response message containing html file, displays html. Parsing html file, finds 10 referenced jpeg objects

6. Steps 1-5 repeated for each of 10 jpeg objects

- RTT is the time taken for a packet to travel from client to server and then back to the client.
- The total response time is sum of following (Figure 1.4):

  i) One RTT to initiate TCP connection (RTT → Round Trip Time).

  ii) One RTT for HTTP request and first few bytes of HTTP response to return.

  iii) File transmission time.

i.e. Total response time = (i) + (ii) + (iii) = 1 RTT+ 1 RTT+ File transmission time
= 2(RTT) + File transmission time

Http with Persistent connection

- Problem with Non-Persistent Connections:

  1) A new connection must be established and maintained for each requested-object.

  ➢ Hence, buffers must be allocated and state info must be kept in both the client and server.

  ➢ This results in a significant burden on the server.

  2) Each object suffers a delivery delay of two RTTs:

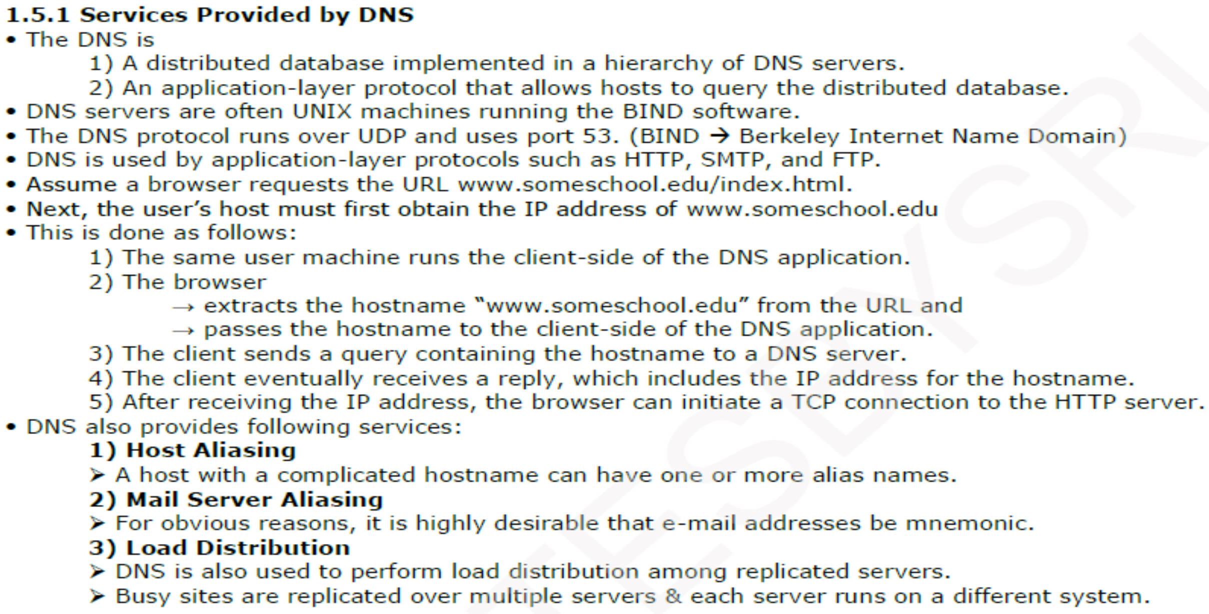
  i) One RTT to establish the TCP connection and

  ii) One RTT to request and receive an object.
- Solution: Use persistent connections.
- With persistent connections, the server leaves the TCP connection open after sending responses.
- Hence, subsequent requests & responses b/w same client & server can be sent over same connection
- The server closes the connection only when the connection is not used for a certain amount of time.
- Default mode of HTTP: Persistent connections with pipelining.
- Advantages:

  1) This method requires only one RTT for all the referenced-objects.

  2) The performance is improved by 20%.

c)



1.5.1 Services Provided by DNS

- The DNS is
  1) A distributed database implemented in a hierarchy of DNS servers.
  2) An application-layer protocol that allows hosts to query the distributed database.
- DNS servers are often UNIX machines running the BIND software.
- The DNS protocol runs over UDP and uses port 53. (BIND → Berkeley Internet Name Domain)
- DNS is used by application-layer protocols such as HTTP, SMTP, and FTP.
- Assume a browser requests the URL www.someschool.edu/index.html.
- Next, the user's host must first obtain the IP address of www.someschool.edu
- This is done as follows:
  1) The same user machine runs the client-side of the DNS application.
  2) The browser
     → extracts the hostname "www.someschool.edu" from the URL and
     → passes the hostname to the client-side of the DNS application.
  3) The client sends a query containing the hostname to a DNS server.
  4) The client eventually receives a reply, which includes the IP address for the hostname.
  5) After receiving the IP address, the browser can initiate a TCP connection to the HTTP server.
- DNS also provides following services:

  1) Host Aliasing
  ➢ A host with a complicated hostname can have one or more alias names.

  2) Mail Server Aliasing
  ➢ For obvious reasons, it is highly desirable that e-mail addresses be mnemonic.

  3) Load Distribution
  ➢ DNS is also used to perform load distribution among replicated servers.
  ➢ Busy sites are replicated over multiple servers & each server runs on a different system.

2. a) Demonstrate socket implementations using TCP (08)

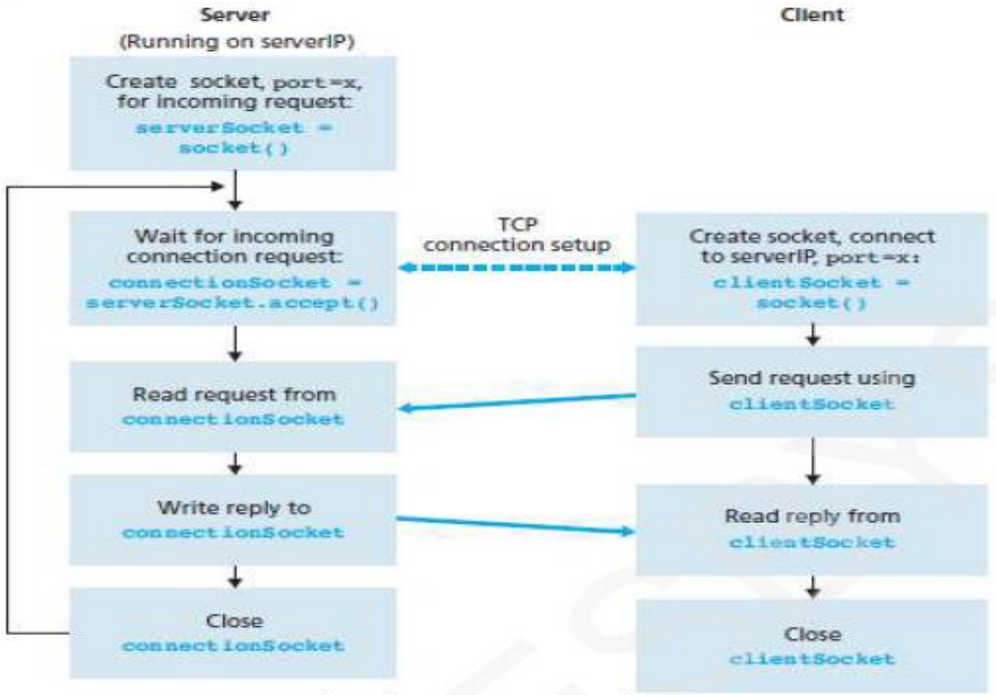
b)Write a note on Web Caching (04)

c) Illustrate the basic operation of SMTP with an example (04)

Answer:

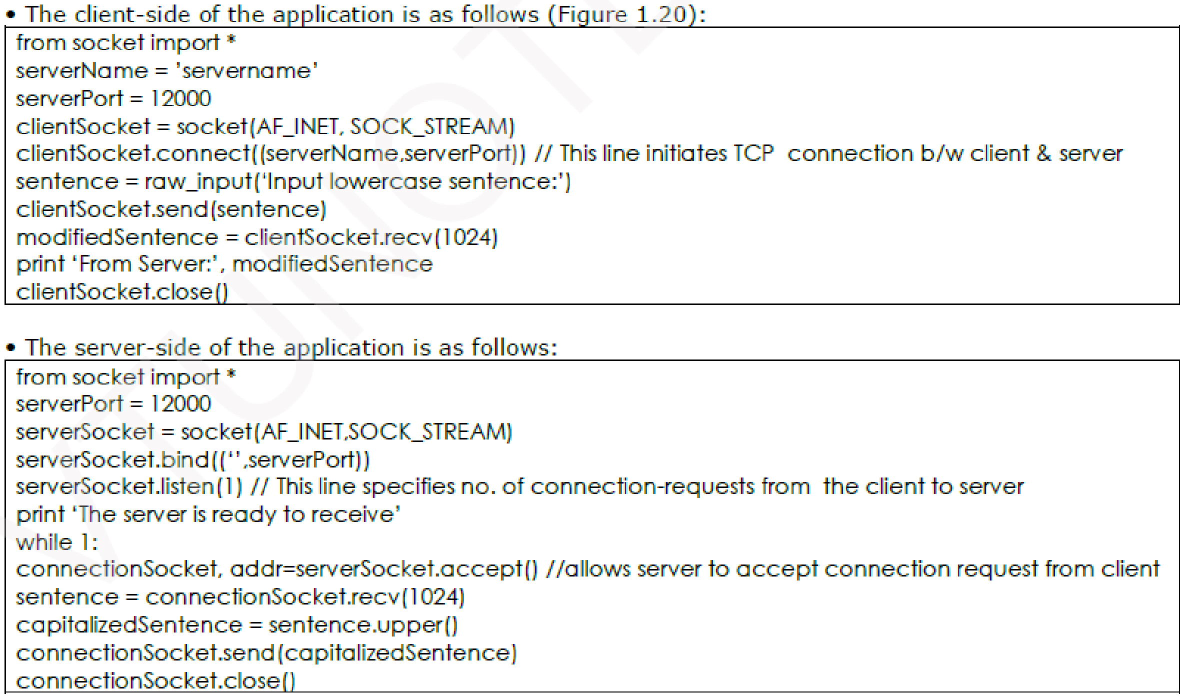
a)

Socket Programming with TCP



Server
(Running on serverIP)
Client
Create socket, port=x, for incoming request:
serverSocket = socket()
Wait for incoming connection request:
connectionSocket = serverSocket.accept()
TCP connection setup
Create socket, connect to serverIP, port=x:
clientSocket = socket()
Read request from connectionSocket
Send request using clientSocket
Write reply to connectionSocket
Read reply from clientSocket
Close connectionSocket
Close clientSocket

Figure 1.20: The client-server application using TCP



- The client-side of the application is as follows (Figure 1.20):

```
from socket import *
serverName = 'servername'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName,serverPort)) // This line initiates TCP  connection b/w client & server
sentence = raw_input('Input lowercase sentence:')
clientSocket.send(sentence)
modifiedSentence = clientSocket.recv(1024)
print 'From Server:', modifiedSentence
clientSocket.close()
```

- The server-side of the application is as follows:

```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET,SOCK_STREAM)
serverSocket.bind(('',serverPort))
serverSocket.listen(1) // This line specifies no. of connection-requests from  the client to server
print 'The server is ready to receive'
while 1:
connectionSocket, addr=serverSocket.accept() //allows server to accept connection request from client
sentence = connectionSocket.recv(1024)
capitalizedSentence = sentence.upper()
connectionSocket.send(capitalizedSentence)
connectionSocket.close()
```

b)

1.2.5 Web Caching

- A Web-cache is a network entity that satisfies HTTP requests on the behalf of an original Web-server.
- The Web-cache has disk-storage.
- The disk-storage contains copies of recently requested-objects.

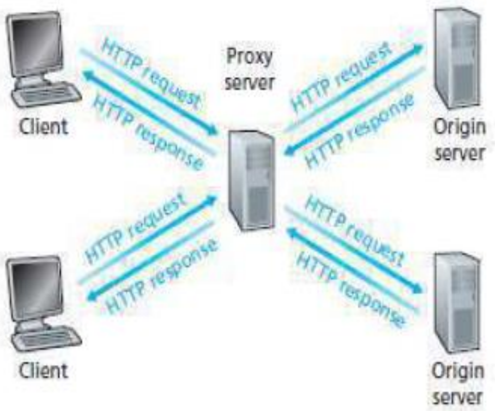


Figure 1.8: Clients requesting objects through a Web-cache (or Proxy Server)

- Here is how it works (Figure 1.8):
  1) The user's HTTP requests are first directed to the web-cache.
  2) If the cache has the object requested, the cache returns the requested-object to the client.
  3) If the cache does not have the requested-object, then the cache
     → connects to the original server and
     → asks for the object.
  4) When the cache receives the object, the cache
     → stores a copy of the object in local-storage and
     → sends a copy of the object to the client.
- A cache acts as both a server and a client at the same time.
  1) The cache acts as a server when the cache
     → receives requests from a browser and
     → sends responses to the browser.
  2) The cache acts as a client when the cache
     → requests to an original server and
     → receives responses from the origin server.
- Advantages of caching:
  1) To reduce response-time for client-request.
  2) To reduce traffic on an institution's access-link to the Internet.
  3) To reduce Web-traffic in the Internet.

c) SMTP

- SMTP is an application-layer protocol used for email.
- SMTP uses TCP to transfer mail from the sender's mail-server to the recipient's mail-server.
- SMTP has two sides:
  1) A client-side, which executes on the sender's mail-server.
  2) A server-side, which executes on the recipient's mail-server.
- Both the client and server-sides of SMTP run on every mail-server.
- When a mail-server receives mail from other mail-servers, the mail-server acts as a server.
  When a mail-server sends mail to other mail-servers, the mail-server acts as a client.

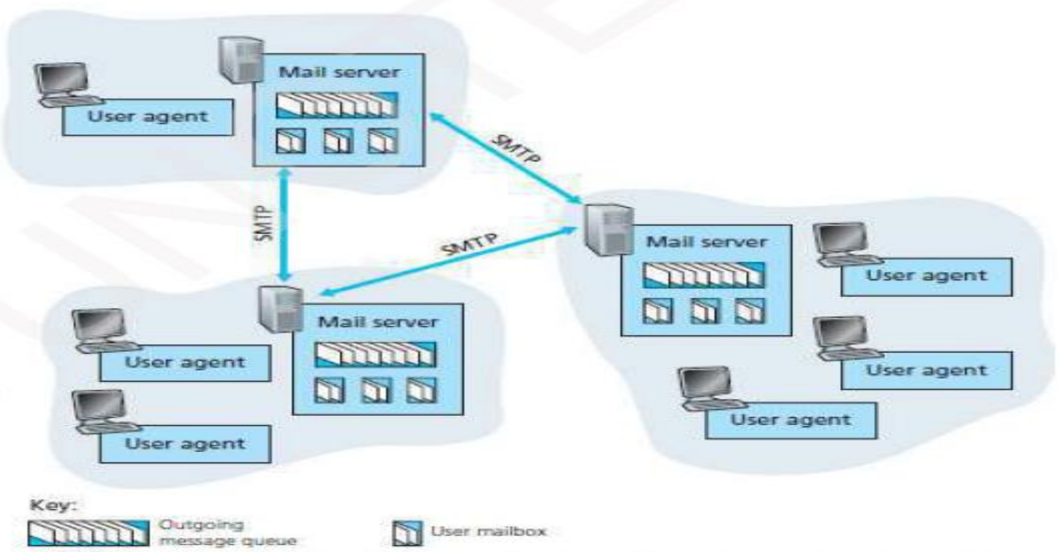


Figure 1.11: A high-level view of the Internet e-mail system

- SMTP is the most important protocol of the email system.
- Three characteristics of SMTP (that differs from other applications):
    1) Message body uses 7-bit ASCII code only.
    2) Normally, no intermediate mail-servers used for sending mail.
    3) Mail transmissions across multiple networks through mail relaying.
- Here is how it works:
    1) Usually, mail-servers are listening at port 25.
    2) The sending server initiates a TCP connection to the receiving mail-server.
    3) If the receiver's server is down, the sending server will try later.
    4) If connection is established, the client & the server perform application-layer handshaking.
    5) Then, the client indicates the e-mail address of the sender and the recipient.
    6) Finally, the client sends the message to the server over the same TCP connection.

Module-2

3)a) Elaborate the three way handshaking in TCP (05)

b) Discuss Go-Back N Protocol (06)

c)Explain the connection oriented multiplexing and demultiplexing (05)

Answer:

a) Three Way Handshaking

2.5.6.1 Connection Setup & Data Transfer

- To setup the connection, three segments are sent between the two hosts. Therefore, this process is referred to as a three-way handshake.
- Suppose a client-process wants to initiate a connection with a server-process.
- Figure 2.33 illustrates the steps involved:

Step 1: Client sends a connection-request segment to the Server

- ➢ The client first sends a connection-request segment to the server.
- ➢ The connection-request segment contains:
    1) SYN bit is set to 1.
    2) Initial sequence-number (client_isn).
- ➢ The SYN segment is encapsulated within an IP datagram and sent to the server.

Step 2: Server sends a connection-granted segment to the Client

- ➢ Then, the server
    - → extracts the SYN segment from the datagram
    - → allocates the buffers and variables to the connection and
    - → sends a connection-granted segment to the client.
- ➢ The connection-granted segment contains:
    1) SYN bit is set to 1.
    2) Acknowledgment field is set to client_isn+1.
    3) Initial sequence-number (server_isn).

Step 3: Client sends an ACK segment to the Server

- ➢ Finally, the client
    - → allocates buffers and variables to the connection and
    - → sends an ACK segment to the server
- ➢ The ACK segment acknowledges the server.
- ➢ SYN bit is set to zero, since the connection is established.

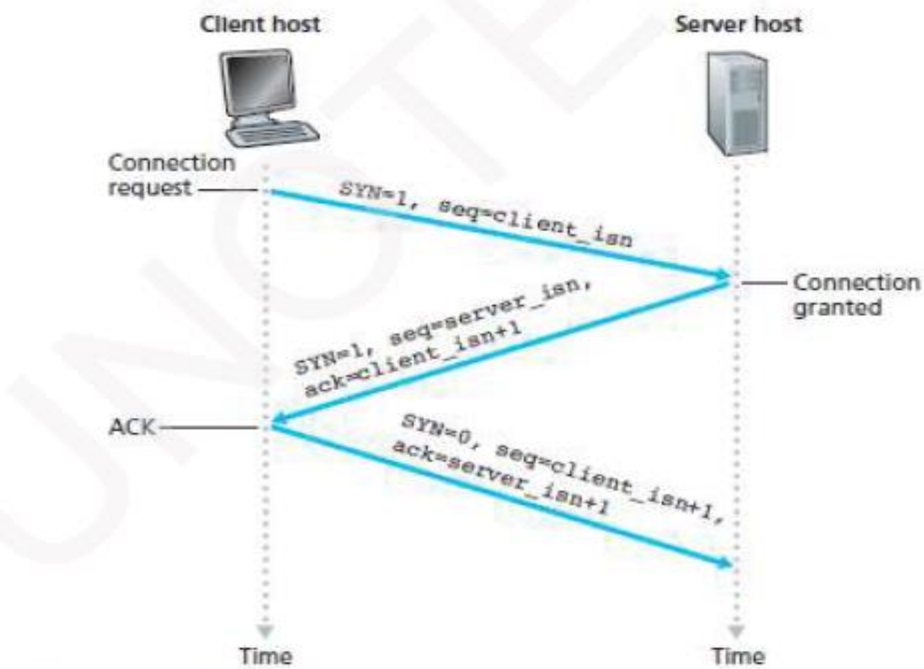


Figure 2.33: TCP three-way handshake: segment exchange

2.5.6.2 Connection Release

- Either of the two processes in a connection can end the connection.
- When a connection ends, the "resources" in the hosts are de-allocated.
- Suppose the client decides to close the connection.
- Figure 2.34 illustrates the steps involved:
  1) The client-process issues a close command.
     ¤ Then, the client sends a shutdown-segment to the server.
     ¤ This segment has a FIN bit set to 1.
  2) The server responds with an acknowledgment to the client.
  3) The server then sends its own shutdown-segment.
     ¤ This segment has a FIN bit set to 1.
  4) Finally, the client acknowledges the server's shutdown-segment.

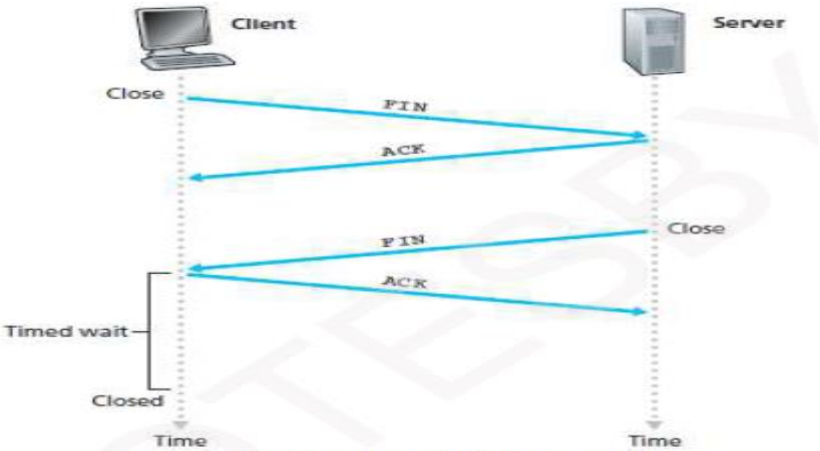


Figure 2.34: Closing a TCP connection

b)

2.4.3 Go-Back-N (GBN)

- The sender is allowed to transmit multiple packets without waiting for an acknowledgment.
- But, the sender is constrained to have at most N unacknowledged packets in the pipeline.
  Where N = window-size which refers maximum no. of unacknowledged packets in the pipeline
- GBN protocol is called a sliding-window protocol.
- Figure 2.17 shows the sender's view of the range of sequence-numbers.

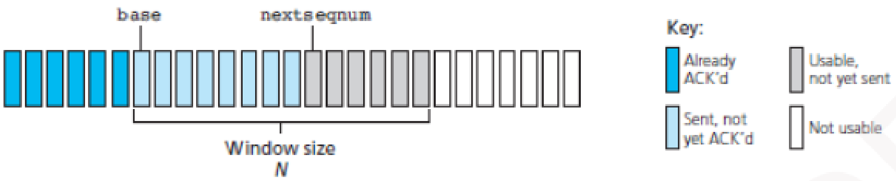
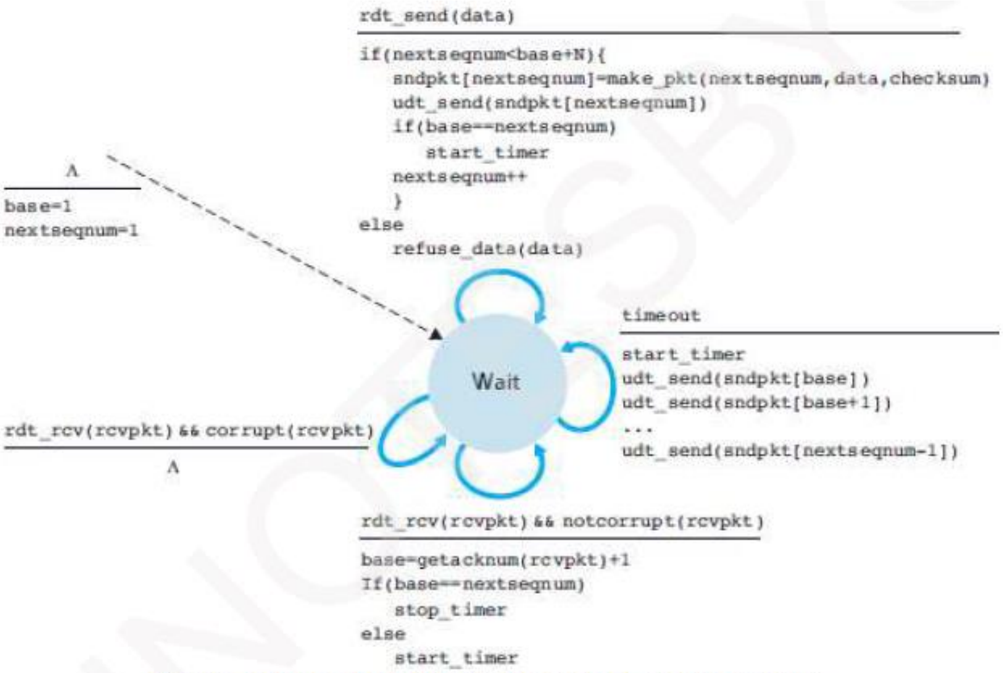


Figure 2.17: Sender's view of sequence-numbers in Go-Back-N

- Figure 2.18 and 2.19 give a FSM description of the sender and receivers of a GBN protocol.



```
rdt_send(data)
________________________________
if(nextseqnum<base+N){
   sndpkt[nextseqnum]=make_pkt(nextseqnum,data,checksum)
   udt_send(sndpkt[nextseqnum])
   if(base==nextseqnum)
      start_timer
   nextseqnum++
   }
else
   refuse_data(data)
```

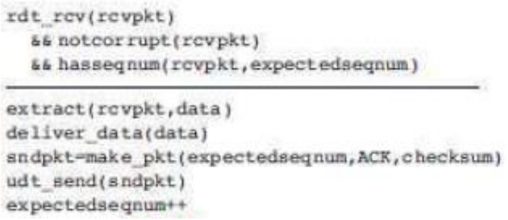
```
Λ
________________________________
base=1
nextseqnum=1
```

```
timeout
________________________________
start_timer
udt_send(sndpkt[base])
udt_send(sndpkt[base+1])
...
udt_send(sndpkt[nextseqnum-1])
```

```
rdt_rcv(rcvpkt) && corrupt(rcvpkt)
________________________________
Λ
```

```
rdt_rcv(rcvpkt) && notcorrupt(rcvpkt)
________________________________
base=getacknum(rcvpkt)+1
If(base==nextseqnum)
   stop_timer
else
   start_timer
```

Figure 2.18: Extended FSM description of GBN sender



```
rdt_rcv(rcvpkt)
  && notcorrupt(rcvpkt)
  && hasseqnum(rcvpkt,expectedseqnum)
________________________________
extract(rcvpkt,data)
deliver_data(data)
sndpkt=make_pkt(expectedseqnum,ACK,checksum)
udt_send(sndpkt)
expectedseqnum++
```

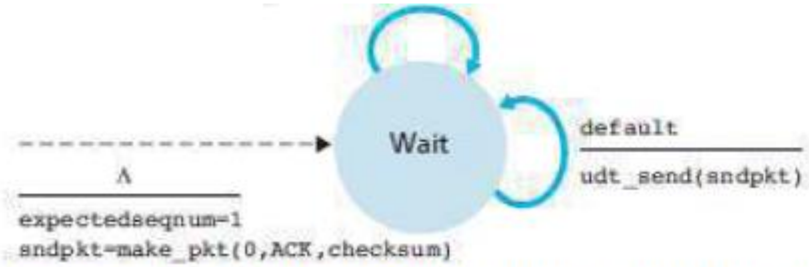


Figure 2.19: Extended FSM description of GBN receiver

2.4.3.1 GBN Sender

- The sender must respond to 3 types of events:

1) Invocation from above.

➢ When rdt_send() is called from above, the sender first checks to see if the window is full i.e. whether there are N outstanding, unacknowledged packets.

i) If the window is not full, the sender creates and sends a packet.

ii) If the window is full, the sender simply returns the data back to the upper layer. This is an implicit indication that the window is full.

2) Receipt of an ACK.

➢ An acknowledgment for a packet with sequence-number n will be taken to be a cumulative acknowledgment.

➢ All packets with a sequence-number up to n have been correctly received at the receiver.

3) A Timeout Event.

➢ A timer will be used to recover from lost data or acknowledgment packets.

i) If a timeout occurs, the sender resends all packets that have been previously sent but that have not yet been acknowledged.

ii) If an ACK is received but there are still additional transmitted but not yet acknowledged packets, the timer is restarted.

iii) If there are no outstanding unacknowledged packets, the timer is stopped.

2.4.3.2 GBN Receiver

- If a packet with sequence-number n is received correctly and is in order, the receiver

→ sends an ACK for packet n and

→ delivers the packet to the upper layer.

- In all other cases, the receiver

→ discards the packet and

→ resends an ACK for the most recently received in-order packet.

2.4.3.3 Operation of the GBN Protocol

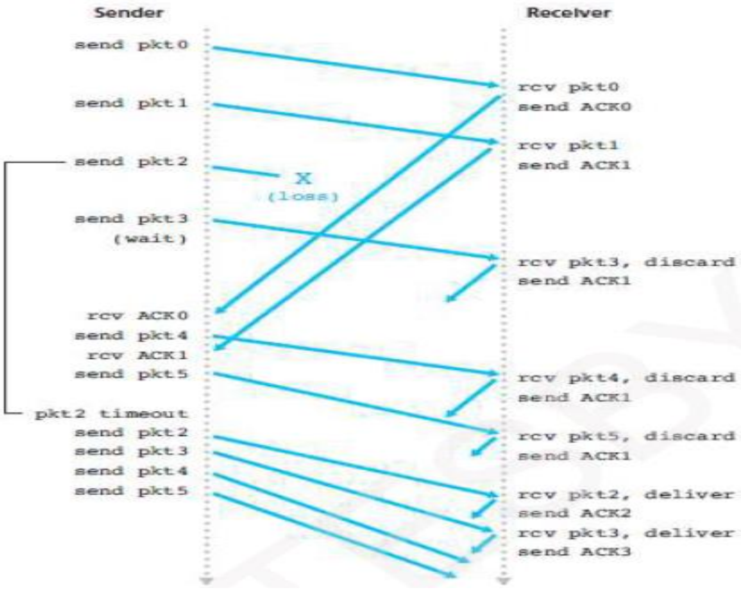


Figure 2.20: Go-Back-N in operation

- Figure 2.20 shows the operation of the GBN protocol for the case of a window-size of four packets.
- The sender sends packets 0 through 3.
- The sender then must wait for one or more of these packets to be acknowledged before proceeding.
- As each successive ACK (for ex, ACK0 and ACK1) is received, the window slides forward and the sender transmits one new packet (pkt4 and pkt5, respectively).
- On the receiver, packet 2 is lost and thus packets 3, 4, and 5 are found to be out of order and are discarded.

c)

2.2.3 Connection Oriented Multiplexing and Demultiplexing

- Each TCP connection has exactly 2 end-points. (Figure 2.4).
- Thus, 2 arriving TCP segments with different source-port-nos will be directed to 2 different sockets, even if they have the same destination-port-no.
- A TCP socket is identified by a four-tuple:
  1) Source IP address
  2) Source-port-no
  3) Destination IP address &
  4) Destination-port-no.

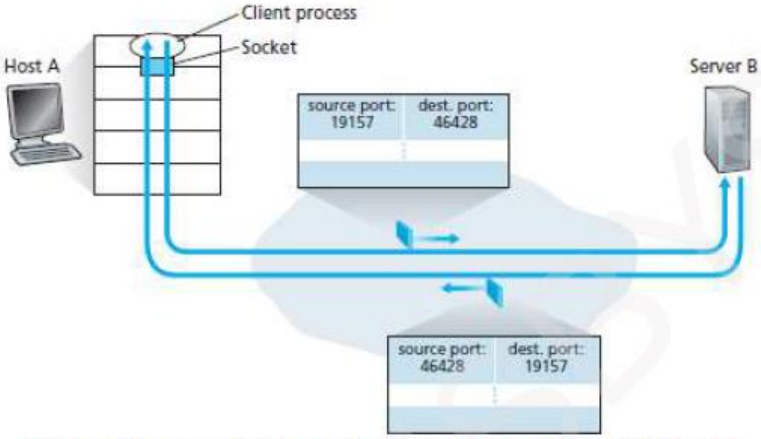


Figure 2.4: The inversion of source and destination-port-nos

- The server-host may support many simultaneous connection-sockets.
- Each socket will be
  → attached to a process.
  → identified by its own four tuple.
- When a segment arrives at the host, all 4 fields are used to direct the segment to the appropriate socket. (i.e. Demultiplexing).

4)a)State Congestion and discuss the cause of congestion (04)

b) With a neat diagram explain the TCP Segment structure (08)

c) Suppose that two measured sample RTT values are 106ms and 120ms. Compute:

i) Estimate RTT after each of these sample RTT value is obtained. Assume α=0.125 and estimated RTT is 100ms just before first of the samples obtained.

ii)Compute DevRTT,

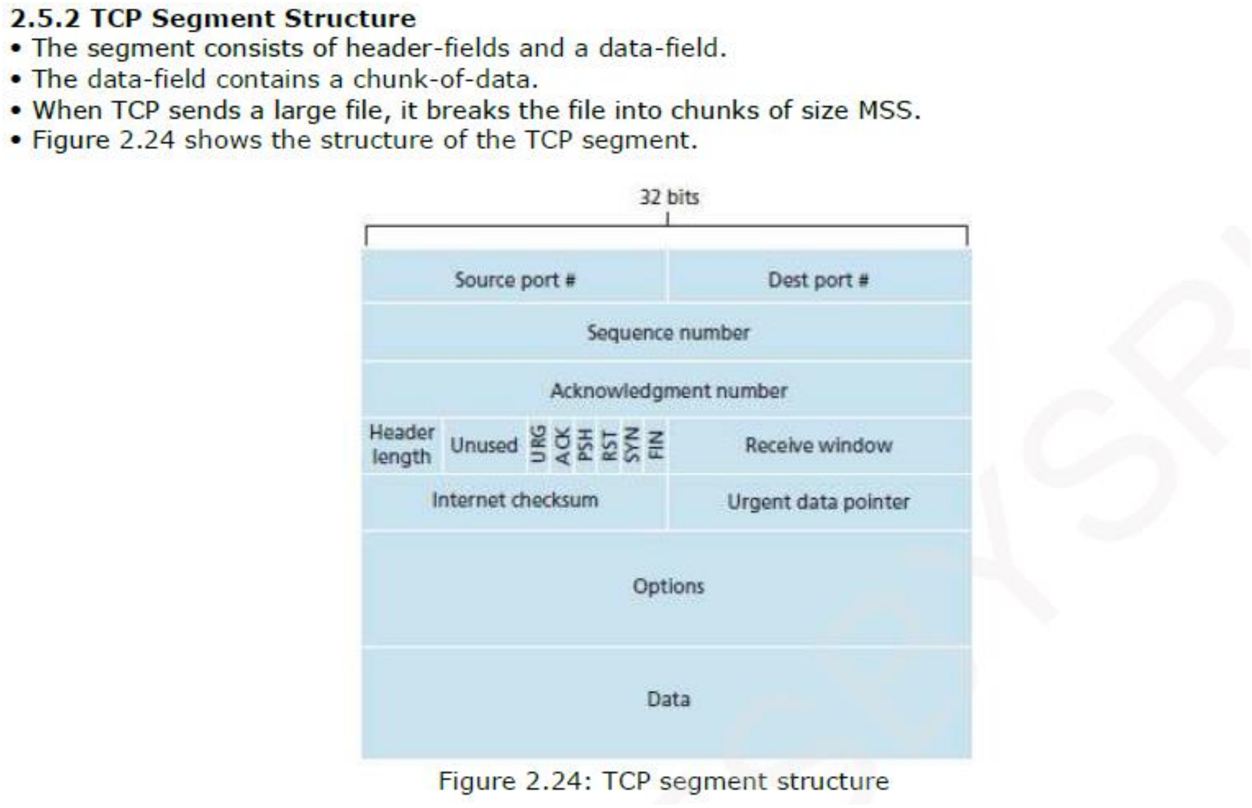
Assume β=0.25 and DevRTT was 5ms before first of these samples are obtained (04)

a) Network congestion in data networking and queueing theory is the reduced quality of service that occurs when a network node or link is carrying more data than it can handle. Typical effects include queueing delay, packet loss or the blocking of new connections.

Cause of congetion

- Too many hosts in broadcast domain. ...
- Broadcast Storms. ...
- Low Bandwidth. ...
- Adding Retransmitting Hubs
- Multicasting.
- Outdated Hardware. ...
- Bad Configuration Management.
- Rogue Adapter Broadcasts.

b) TCP Segment structure



2.5.2 TCP Segment Structure

- The segment consists of header-fields and a data-field.
- The data-field contains a chunk-of-data.
- When TCP sends a large file, it breaks the file into chunks of size MSS.
- Figure 2.24 shows the structure of the TCP segment.

Figure 2.24: TCP segment structure

- The fields of TCP segment are as follows:

1) Source and Destination Port Numbers

➢ These fields are used for multiplexing/demultiplexing data from/to upper-layer applications.

2) Sequence Number & Acknowledgment Number

➢ These fields are used by sender & receiver in implementing a reliable data-transfer-service.

3) Header Length

➢ This field specifies the length of the TCP header.

4) Flag

➢ This field contains 6 bits.

i) ACK

¤ This bit indicates that value of acknowledgment field is valid.

ii) RST, SYN & FIN

¤ These bits are used for connection setup and teardown.

iii) PSH

¤ This bit indicates the sender has invoked the push operation.

iv) URG

¤ This bit indicates the segment contains urgent-data.

5) Receive Window

➢ This field defines receiver's window size

➢ This field is used for flow control.

6) Checksum

➢ This field is used for error-detection.

7) Urgent Data Pointer

➢ This field indicates the location of the last byte of the urgent data.

8) Options

➢ This field is used when a sender & receiver negotiate the MSS for use in high-speed networks.

2.5.2.1 Sequence Numbers and Acknowledgment Numbers

Sequence Numbers

- The sequence-number is used for sequential numbering of packets of data flowing from sender to receiver.
- Applications:

1) Gaps in the sequence-numbers of received packets allow the receiver to detect a lost packet.

2) Packets with duplicate sequence-numbers allow the receiver to detect duplicate copies of a packet.

Acknowledgment Numbers

- The acknowledgment-number is used by the receiver to tell the sender that a packet has been received correctly.
- Acknowledgments will typically carry the sequence-number of the packet being acknowledged.

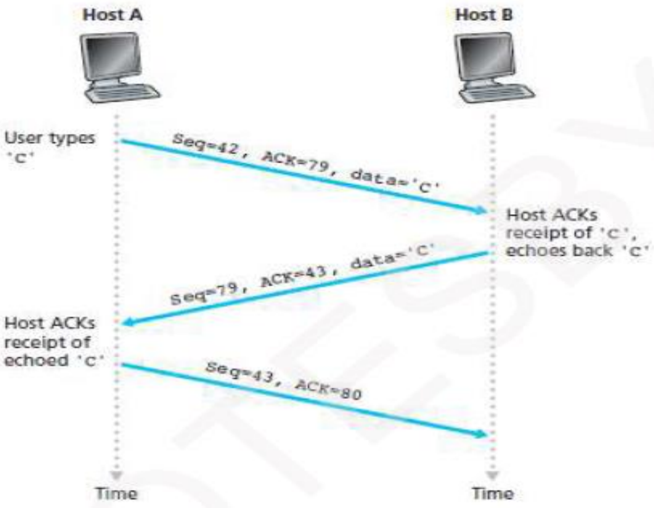


Figure 2.25: Sequence and acknowledgment-numbers for a simple Telnet application over TCP

c) $\text{EstimatedRTT} = (1 - \alpha) \cdot \text{EstimatedRTT} + \alpha \cdot \text{SampleRTT}$

i) For first sample where sample RTT = 106ms the Estimated RTT is:

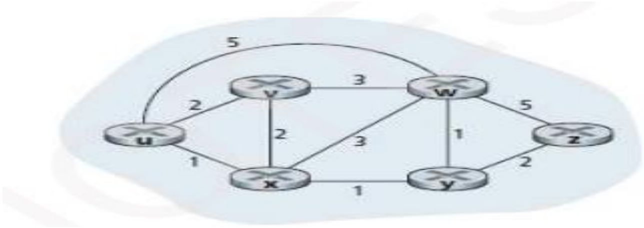
Estimated RTT= (1-0.125)100 + 113 = 200.5 ms (as α =.125 and Estimated RTT= 100ms) (astaken avg of two sample values (120+106)/2= 113)

ii) $\text{DevRTT} = (1 - \beta) \cdot \text{DevRTT} + \beta \cdot | \text{SampleRTT} - \text{EstimatedRTT} |$

=(1-0.25).5 + 0.25 | 113-200.5| = 3.75 + 21.875 = 25.625

Module-3

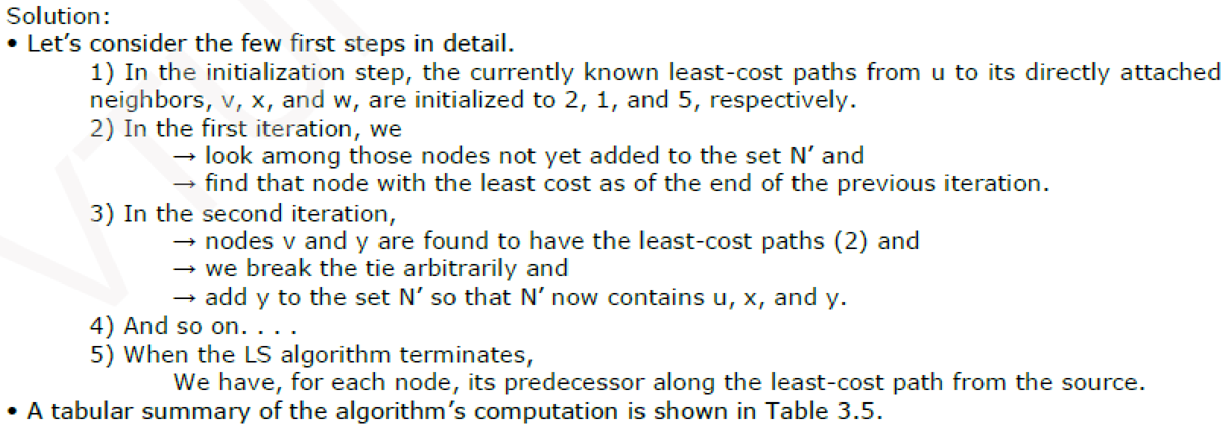
5. a) Write the Link State routing algorithm. Solve the following graph using Link-state algorithm with source node 'u'. (08)



b) What is Routing? Explain the structure of Router (08)

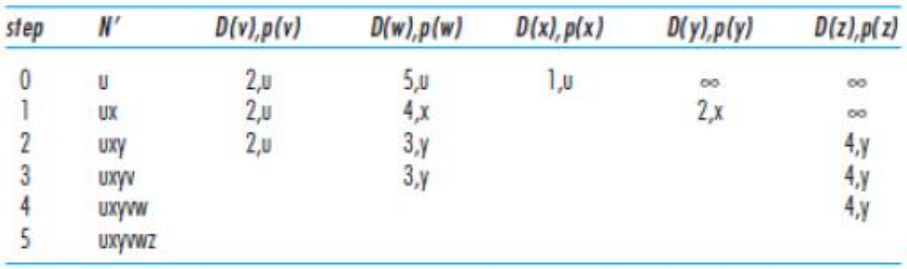
Answer:

a) Link State routing algorithm : The link-state protocol is performed by every switching node in the network (i.e., nodes that are prepared to forward packets; in the Internet, these are called routers). The basic concept of link-state routing is that every node constructs a map of the connectivity to the network, in the form of a graph, showing which nodes are connected to which other nodes. Each node then independently calculates the next best logical path from it to every possible destination in the network. Each collection of best paths will then form each node's routing table.



Solution:

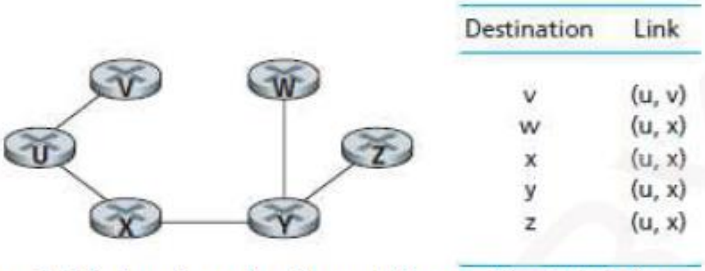
- Let's consider the few first steps in detail.
  1) In the initialization step, the currently known least-cost paths from u to its directly attached neighbors, v, x, and w, are initialized to 2, 1, and 5, respectively.
  2) In the first iteration, we
     → look among those nodes not yet added to the set N' and
     → find that node with the least cost as of the end of the previous iteration.
  3) In the second iteration,
     → nodes v and y are found to have the least-cost paths (2) and
     → we break the tie arbitrarily and
     → add y to the set N' so that N' now contains u, x, and y.
  4) And so on. . . .
  5) When the LS algorithm terminates,
     We have, for each node, its predecessor along the least-cost path from the source.
- A tabular summary of the algorithm's computation is shown in Table 3.5.



step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

Table 3.5: Running the link-state algorithm on the network in Figure 3.20

Figure 3.23 shows the resulting least-cost paths for u for the network in Figure 3.22.



Destination	Link
v	(u, v)
w	(u, x)
x	(u, x)
y	(u, x)
z	(u, x)

Figure 3.23: Least cost path and forwarding-table for node u

b)

- ➢ Routing means determining the path taken by packets from a sender to a receiver.
- ➢ Routing is a network-wide process.

3.3 What's Inside a Router?

- The router is used for transferring packets from an incoming-links to the appropriate outgoing-links.

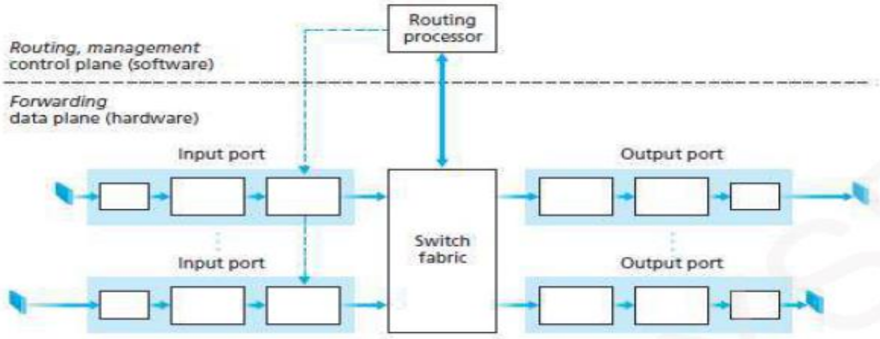

Routing processor
Routing, management control plane (software)
Forwarding data plane (hardware)
Input port
Output port
Switch fabric
Input port
Output port

Figure 3.5: Router architecture

- Four components of router (Figure 3.5):

1) Input Ports

- An input-port is used for terminating an incoming physical link at a router (Figure 3.6).
- It is used for interoperating with the link layer at the other side of the incoming-link.
- It is used for lookup function i.e. searching through forwarding-table looking for longest prefix match.
- It contains forwarding-table.
- Forwarding-table is consulted to determine output-port to which arriving packet will be forwarded.
- Control packets are forwarded from an input-port to the routing-processor.
- Many other actions must be taken:
  i) Packet's version number, checksum and time-to-live field must be checked.
  ii) Counters used for network management must be updated.

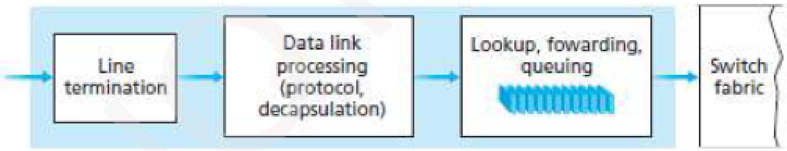

Line termination
Data link processing (protocol, decapsulation)
Lookup, fowarding, queuing
Switch fabric

Figure 3.6: Input port processing

) Switching Fabric

The switching fabric connects the router's input-ports to its output-ports.
In fabric, the packets are switched (or forwarded) from an input-port to an output-port.
In fact, fabric is a network inside of a router.
A packet may be temporarily blocked if packets from other input-ports are currently using the fabric.
A blocked packet will be queued at the input-port & then scheduled to send at a later point in time.

) Output Ports

An output-port

→ stores packets received from the switching fabric and
→ transmits the packets on the outgoing-link.

For a bidirectional link, an output-port will typically be paired with the input-port.

) Routing Processor

The routing-processor

→ executes the routing protocols
→ maintains routing-tables & attached link state information and
→ computes the forwarding-table.

It also performs the network management functions.

3.3.2 Output Processing

- Output-port processing
  → takes the packets stored in the output-port's memory and
  → transmits the packets over the output link (Figure 3.8).
- This includes
  → selecting and dequeueing packets for transmission and
  → performing the linklayer and physical-layer transmission functions.

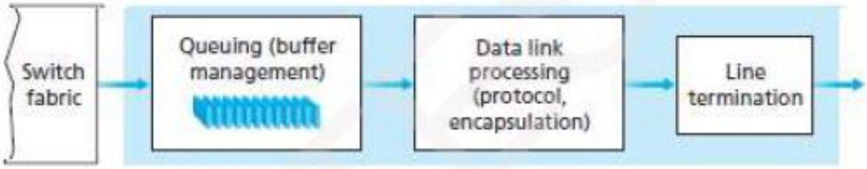

Switch fabric
Queuing (buffer management)
Data link processing (protocol, encapsulation)
Line termination

Figure 3.8: Output port processing

6. a)Discuss the IPV6 packet format (05)

b)Elaborate the path attributes in BGP and steps to select the BGP routes (05)

c)List the broadcast routing algorithm. Explain any one of them. (06)

a)

3.4.5.2 IPv6 Datagram Format

- The format of the IPv6 datagram is shown in Figure 3.18.

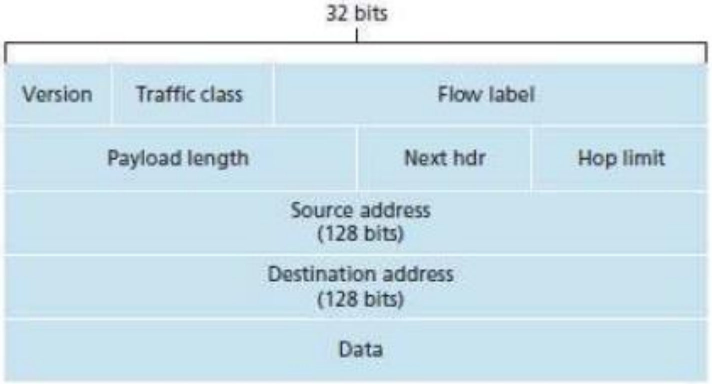


Figure 3.18: IPv6 datagram format

- The following fields are defined in IPv6:

1) Version

➤ This field specifies the IP version, i.e., 6.

2) Traffic Class

➤ This field is similar to the TOS field in IPv4.
➤ This field indicates the priority of the packet.

3) Flow Label

➤ This field is used to provide special handling for a particular flow of data.

4) Payload Length

➤ This field shows the length of the IPv6 payload.

5) Next Header

➤ This field is similar to the options field in IPv4 (Figure 3.19).
➤ This field identifies type of extension header that follows the basic header.

6) Hop Limit

➤ This field is similar to TTL field in IPv4.
➤ This field shows the maximum number of routers the packet can travel.
➤ The contents of this field are decremented by 1 by each router that forwards the datagram.
➤ If the hop limit count reaches 0, the datagram is discarded.

7) Source & Destination Addresses

➤ These fields show the addresses of the source & destination of the packet.

8) Data

➤ This field is the payload portion of the datagram.
➤ When the datagram reaches the destination, the payload will be
→ removed from the IP datagram and
→ passed on to the upper layer protocol (TCP or UDP).

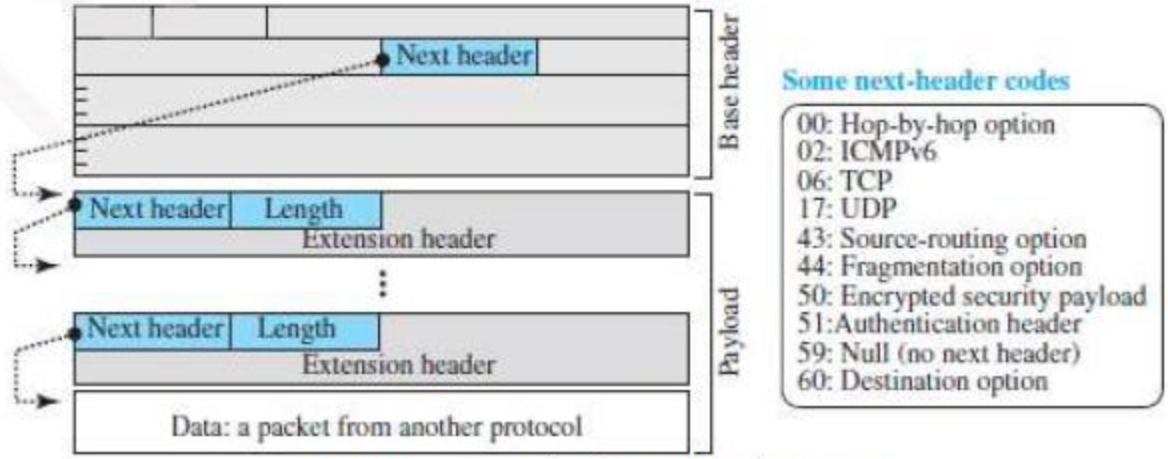


Figure 3.19: Payload in IPv6 datagram

b) path attributes in BGP

- An autonomous-system is identified by its globally unique ASN (Autonomous-System Number).
- A router advertises a prefix across a session.
- The router includes a number of attributes with the prefix.
- Two important attributes: 1) AS-PATH and 2) NEXT-HOP

  1) AS-PATH

  ➢ This attribute contains the ASs through which the advertisement for the prefix has passed.
  ➢ When a prefix is passed into an AS, the AS adds its ASN to the ASPATH attribute.
  ➢ Routers use the AS-PATH attribute to detect and prevent looping advertisements.
  ➢ Routers also use the AS-PATH attribute in choosing among multiple paths to the same prefix.

  2) NEXT-HOP

  ➢ This attribute provides the critical link between the inter-AS and intra-AS routing protocols.
  ➢ This attribute is the router-interface that begins the AS-PATH.
- BGP also includes

  → attributes which allow routers to assign preference-metrics to the routes.
  → attributes which indicate how the prefix was inserted into BGP at the origin AS.
- When a gateway-router receives a route-advertisement, the gateway-router decides

  → whether to accept or filter the route and
  → whether to set certain attributes such as the router preference metrics.

3.6.3.3 Route Selection

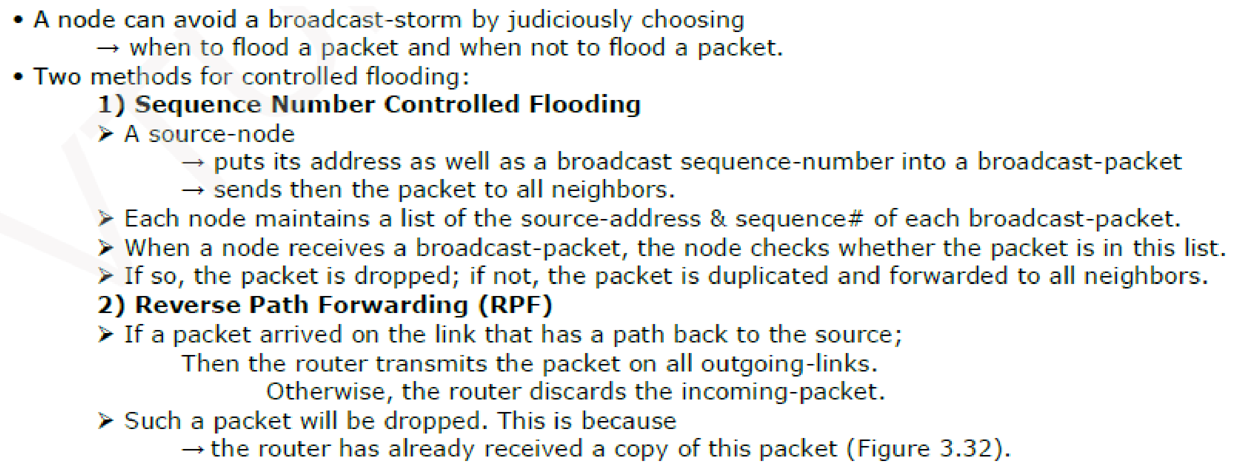
- For 2 or more routes to the same prefix, the following elimination-rules are invoked sequentially:

  1) Routes are assigned a local preference value as one of their attributes.
  2) The local preference of a route
     → will be set by the router or
     → will be learned by another router in the same AS.
  3) From the remaining routes, the route with the shortest AS-PATH is selected.
  4) From the remaining routes, the route with the closest NEXT-HOP router is selected.
  5) If more than one route still remains, the router uses BGP identifiers to select the route.

c) Broadcast routing algorithm

i) N-Way Unicast ii) Uncontrolled flooding iii)Controlled flooding iv) Spanning tree broadcast

Controlled flooding



- A node can avoid a broadcast-storm by judiciously choosing
  → when to flood a packet and when not to flood a packet.
- Two methods for controlled flooding:

1) Sequence Number Controlled Flooding

- A source-node
  → puts its address as well as a broadcast sequence-number into a broadcast-packet
  → sends then the packet to all neighbors.
- Each node maintains a list of the source-address & sequence# of each broadcast-packet.
- When a node receives a broadcast-packet, the node checks whether the packet is in this list.
- If so, the packet is dropped; if not, the packet is duplicated and forwarded to all neighbors.

2) Reverse Path Forwarding (RPF)

- If a packet arrived on the link that has a path back to the source;
  Then the router transmits the packet on all outgoing-links.
  Otherwise, the router discards the incoming-packet.
- Such a packet will be dropped. This is because
  →the router has already received a copy of this packet (Figure 3.32).

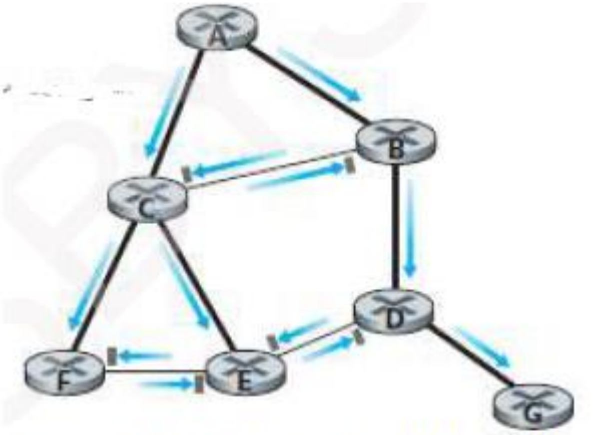


Figure 3.32: Reverse path forwarding

<u>Module-4</u>

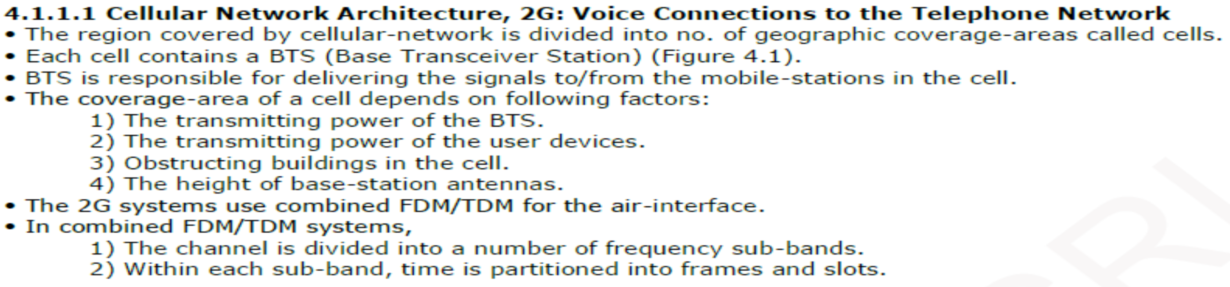
7. a) show the components GSM 2G Cellular network architecture with a diagram (07)

b)Illustrate the steps involved in Mobile IP registration with home agent (05)

c) Write a note on Mobile IP (04)

Answer:

a)



4.1.1.1 Cellular Network Architecture, 2G: Voice Connections to the Telephone Network

- The region covered by cellular-network is divided into no. of geographic coverage-areas called cells.
- Each cell contains a BTS (Base Transceiver Station) (Figure 4.1).
- BTS is responsible for delivering the signals to/from the mobile-stations in the cell.
- The coverage-area of a cell depends on following factors:
  1) The transmitting power of the BTS.
  2) The transmitting power of the user devices.
  3) Obstructing buildings in the cell.
  4) The height of base-station antennas.
- The 2G systems use combined FDM/TDM for the air-interface.
- In combined FDM/TDM systems,
  1) The channel is divided into a number of frequency sub-bands.
  2) Within each sub-band, time is partitioned into frames and slots.

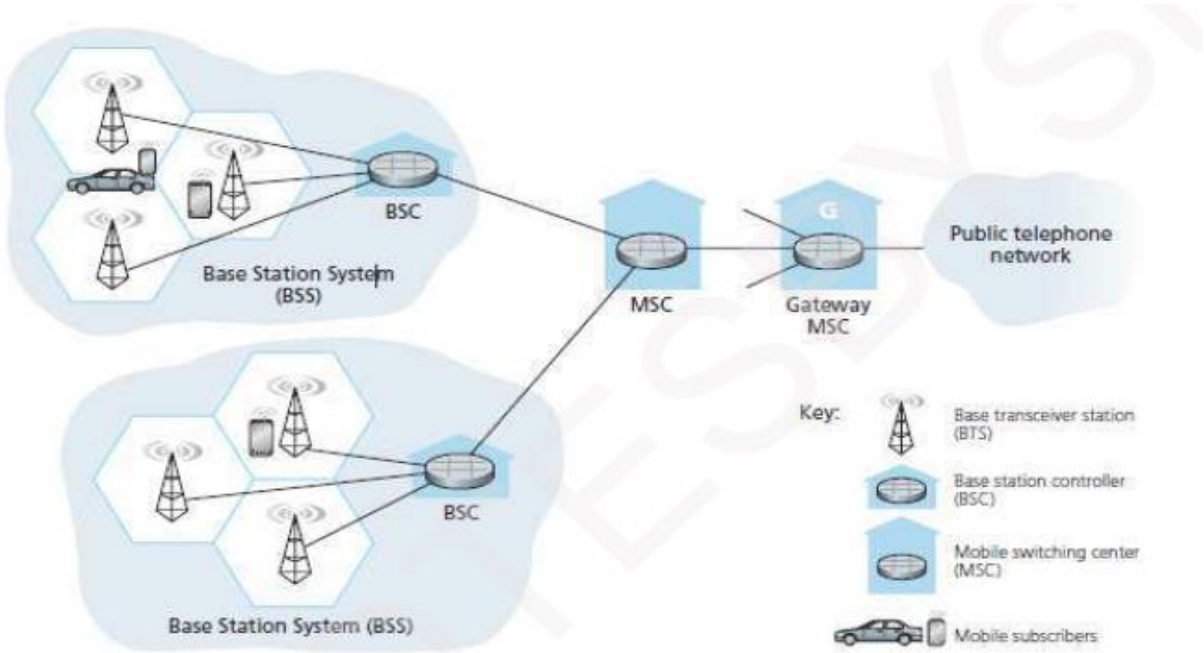


Figure 4.1: Components of the GSM 2G cellular network architecture

- The GSM network contains many BSCs (Base Station Controllers).
- Main responsibilities of the BSC:
  1) Providing service to many BTSs.
  2) Allocating radio-channels to mobile-users.
  3) Performing paging.
  4) Performing handoff of mobile-users.
- BSS (Base Station System) contains the BSC and its controlled BTSs.
- A MSC (Mobile Switching Center) contains upto 5 BSCs. This results in approx 200K subscribers/MSC.
- Main responsibilities of the MSC:
  1) User authorization & accounting
  2) Call establishment & teardown and
  3) Handoff.
- A cellular-provider's network will have a number of special MSCs known as gateway MSCs.
- Gateway MSCs are used to connect the provider's cellular-network to the public telephone-network.

b)

4.3.2 Registration with the Home Agent

- Address must be registered with the home-agent. This can be done in 2 ways:
  1) Via the foreign-agent who then registers the COA with the home-agent.
  2) By the mobile IP node itself.

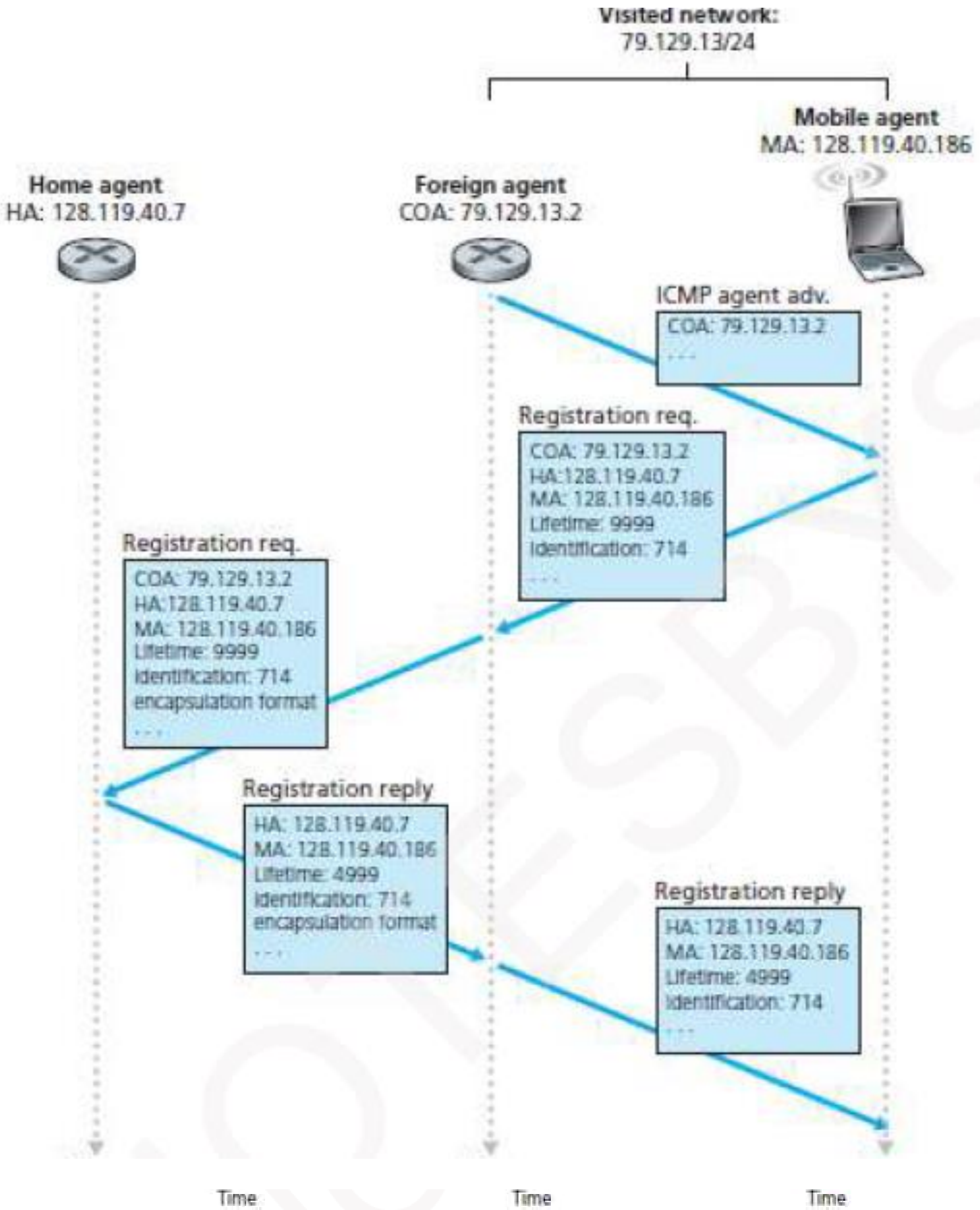


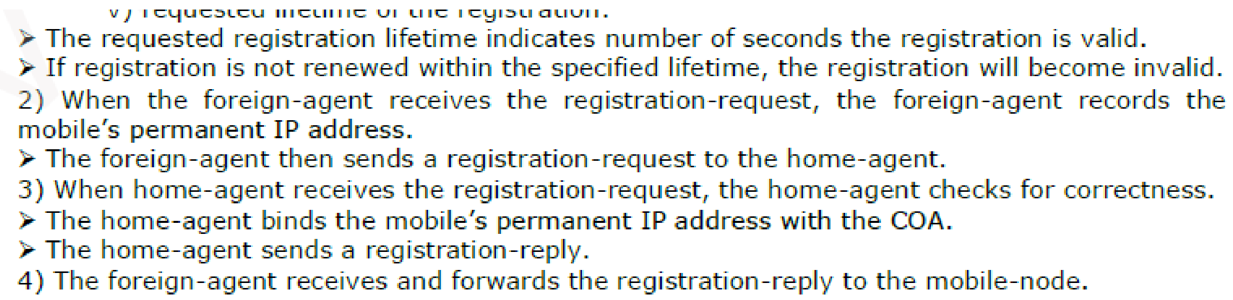
Figure 4.9: Agent advertisement and mobile IP registration

- Four steps are involved. Figure 4.9 illustrates the 4 steps.

  1) When a mobile receives a foreign-agent advertisement, the mobile sends a registration-request to the foreign-agent.

  ➢ The registration-request contains

    i) COA advertised by the foreign-agent
    ii) address of the home-agent (HA)
    iii) permanent-address of the mobile (MA)
    iv) registration identification and
    v) requested lifetime of the registration.



v) requested lifetime of the registration.

➢ The requested registration lifetime indicates number of seconds the registration is valid.

➢ If registration is not renewed within the specified lifetime, the registration will become invalid.

2) When the foreign-agent receives the registration-request, the foreign-agent records the mobile's permanent IP address.

➢ The foreign-agent then sends a registration-request to the home-agent.

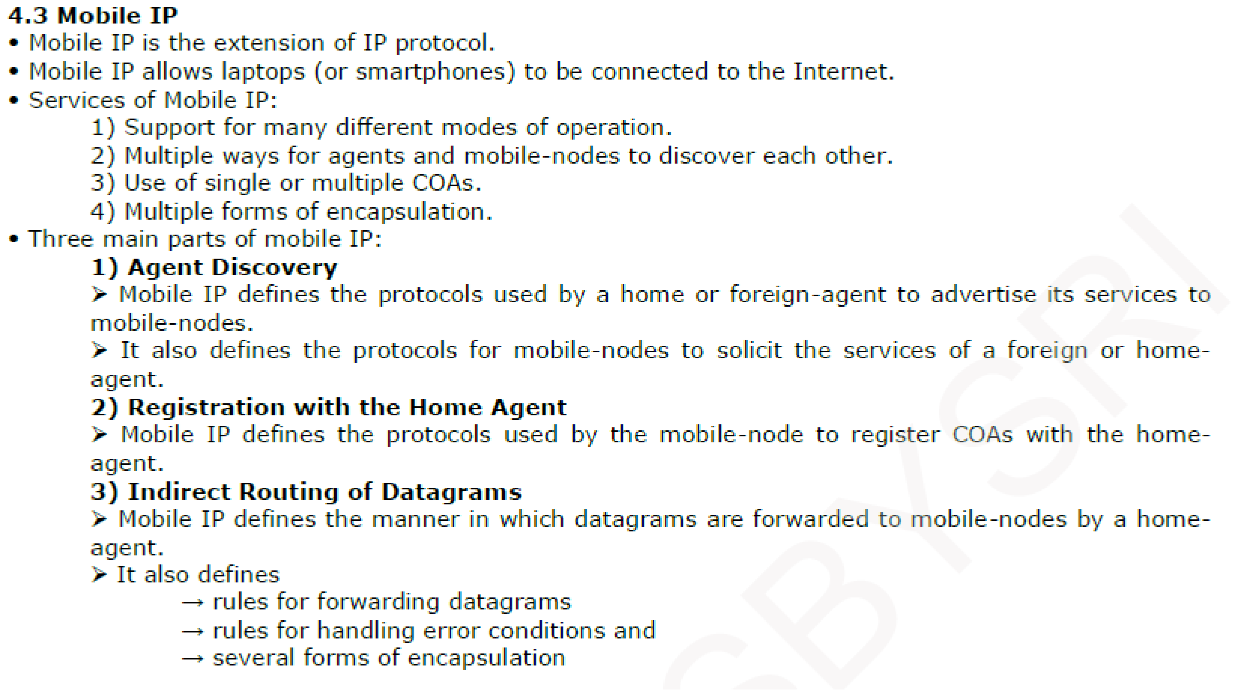
3) When home-agent receives the registration-request, the home-agent checks for correctness.

➢ The home-agent binds the mobile's permanent IP address with the COA.

➢ The home-agent sends a registration-reply.

4) The foreign-agent receives and forwards the registration-reply to the mobile-node.

c)



4.3 Mobile IP

- Mobile IP is the extension of IP protocol.
- Mobile IP allows laptops (or smartphones) to be connected to the Internet.
- Services of Mobile IP:
  1) Support for many different modes of operation.
  2) Multiple ways for agents and mobile-nodes to discover each other.
  3) Use of single or multiple COAs.
  4) Multiple forms of encapsulation.
- Three main parts of mobile IP:

1) Agent Discovery

➢ Mobile IP defines the protocols used by a home or foreign-agent to advertise its services to mobile-nodes.

➢ It also defines the protocols for mobile-nodes to solicit the services of a foreign or home-agent.

2) Registration with the Home Agent

➢ Mobile IP defines the protocols used by the mobile-node to register COAs with the home-agent.

3) Indirect Routing of Datagrams

➢ Mobile IP defines the manner in which datagrams are forwarded to mobile-nodes by a home-agent.

➢ It also defines

→ rules for forwarding datagrams
→ rules for handling error conditions and
→ several forms of encapsulation

8) a)Define Handoff. Explain the steps accomplishing a hand off (07)

b) Bring out the mechanism for direct routing to mobile node in mobility management (06)

c) Compare the 4G LTE standard to 3G systems. (03)

Answer:

a) In a cellular telephone network, handoff is the transition for any given user of signal transmission from one base station to a geographically adjacent base station as the user moves around.

4.4.2 Handoffs in GSM

- A handoff occurs when a mobile-station moves from one base-station to another during a call.
- As shown in Figure 4.11,
    1) Before handoff, a call is initially routed to the mobile through old base-station.
    2) After handoff, the call is routed to the mobile through another new base-station.
- Two reasons for handoff**:**
    1) The Call may be Dropped
    ➢ Because the signal between the current base-station and the mobile may have weakened.
    2) To reduce Congestion
    ➢ Because a cell may be overloaded because of handling a large number of calls.
    ➢ This congestion may be reduced by handing off mobiles to less congested cells.

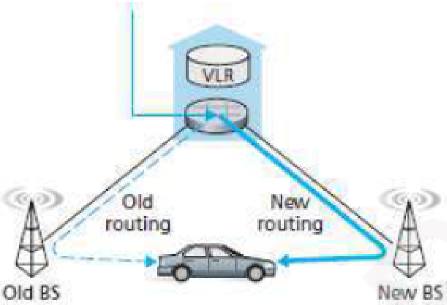


Figure 4.11: Handoff scenario between base stations with a common MSC

- Eight steps are involved. Figure 4.12 illustrates the steps involved when a hand off occurs.
    1) Old base-station (BS) informs both visited MSC & new BS that a handoff is about to happen.
    2) The visited MSC performs following tasks:
        i) Initiates path setup to the new BS.
        ii) Allocates the resources needed to carry the rerouted call.
        iii) Signals the new BS that a handoff is about to occur.
    3) The new BS allocates and activates a radio-channel for the mobile.
    4) The new BS informs both visited MSC and old BS that the new path is set up.
    5) The mobile is informed to perform a handoff.
    6) The mobile & new BS exchange signaling messages to fully activate the new channel.
    7) The mobile sends a handoff complete message to the new BS.
        ¤ This message is then forwarded to the visited MSC.
        ¤ The visited MSC then reroutes the ongoing-call to the mobile via the new BS.
    8) The resources allocated along the path to the old BS are released.

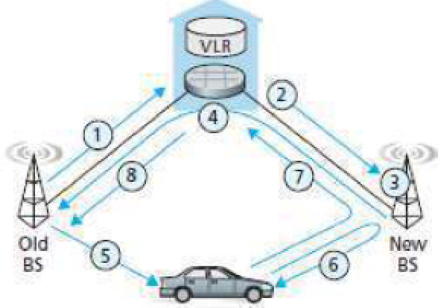


Figure 4.12: A handoff between base stations with a common MSC

b)

4.2.2.2 Direct Routing to a Mobile Node

- Four steps are involved. Figure 4.6 illustrates the 4 steps.

Steps 1 & 2

➢ A correspondent-agent in the correspondent's n/w first learns the COA of the mobile-node.

➢ This can be done by having the correspondent-agent query the home-agent.

Steps 3 & 4

➢ Then, the correspondent-agent forwards datagrams directly to the mobile-node's COA.

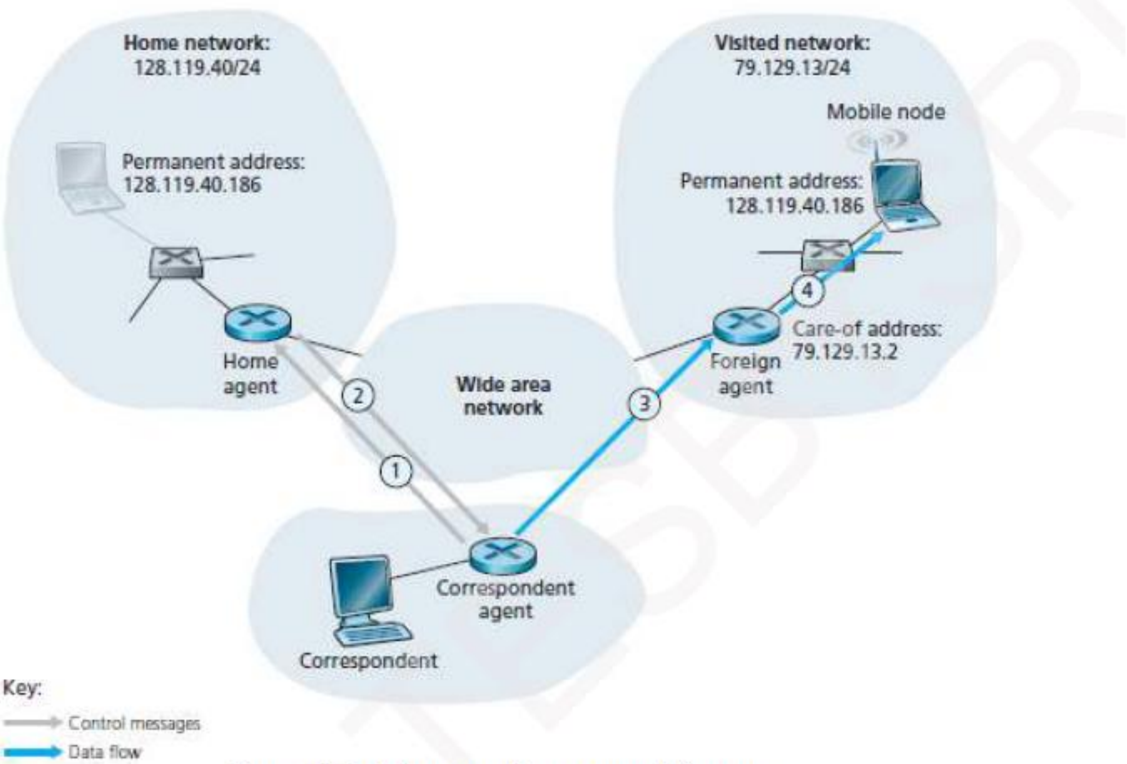


Figure 4.6: Direct routing to a mobile user

4.2.2.2.1 Challenges in Direct Routing

- Two additional challenges:

1) The correspondent-agent needs a mobile location protocol to query the home-agent to obtain the mobile-node's COA (steps 1 & 2 in Figure 4.6).

2) Problem: When the mobile-node moves from one foreign-network to another, how will data be forwarded to the new foreign-network?

Solution: Use anchor foreign-agent.

➢ An anchor foreign-agent refers to a foreign-agent in the foreign-network where the mobile-node was first found. (step 1 in Figure 4.7).

➢ When the mobile-node moves to a new foreign-network (step 2), the mobile-node registers with the new foreign-agent (step 3).

➢ The new foreign-agent provides the anchor foreign-agent with the mobile-node's new COA (step 4).

➢ When the anchor foreign-agent receives an encapsulated-datagram, the anchor re-encapsulates and forwards the datagram to the mobile-node (step 5).

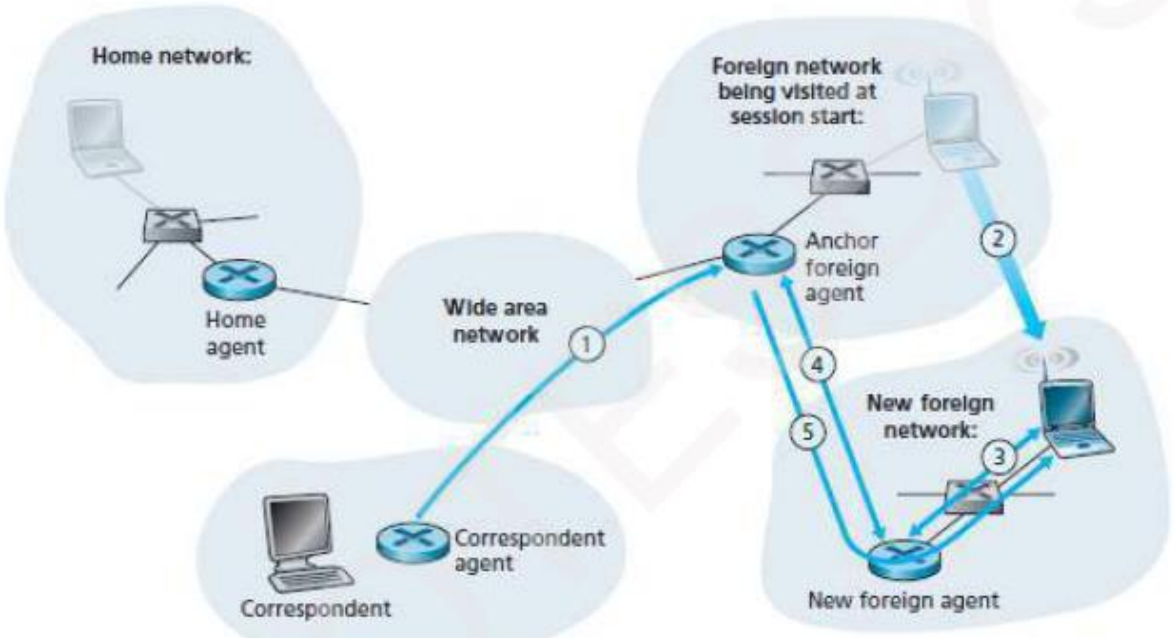


Figure 4.7: Mobile transfer between networks with direct routing

c) 4G LTE standard to 3G systems

Comparison chart

3G versus 4G comparison chart

	3G	4G
Data Throughput	Up to 3.1Mbps with an average speed range between 0.5 to 1.5 Mbps	Practically speaking, 2 to 12 Mbps (Telstra in Australia claims up to 40 Mbps) but potential estimated at a range of 100 to 300 Mbps.
Peak Upload Rate	5 Mbps	500 Mbps
Switching Technique	packet switching	packet switching, message switching
Network Architecture	Wide Area Cell Based	Integration of wireless LAN and Wide area.
Services And	CDMA 2000, UMTS,	Wimax2 and LTE-Advance

3G versus 4G comparison chart

	3G	4G
Applications	EDGE etc	
Forward error correction (FEC)	3G uses Turbo codes for error correction.	Concatenated codes are used for error corrections in 4G.
Peak Download Rate	100 Mbps	1 Gbps
Frequency Band	1.8 – 2.5 GHz	2 – 8 GHz

Module-5

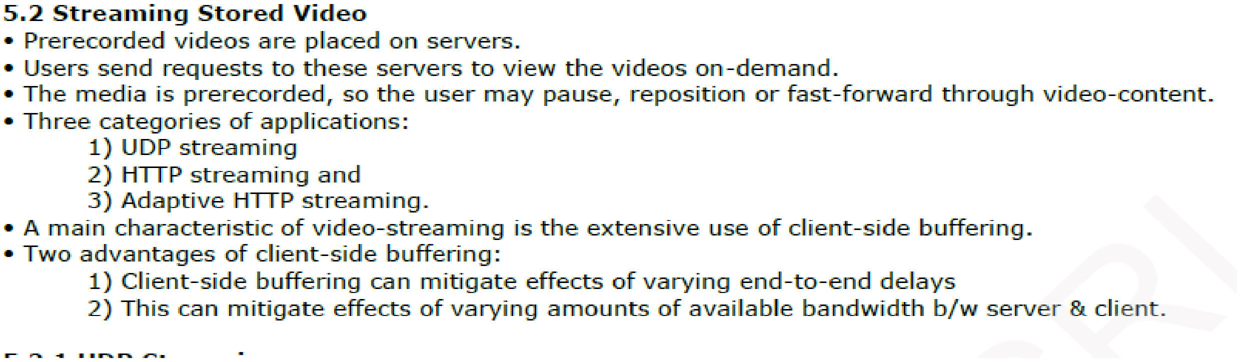
9) a) Elaborate the features of streaming stored video (03)

b)With a neat diagram, explain CDN operation (08)

c) Summarize the limitations of Best-effort IP service (05)

Answer:

a)



5.2 Streaming Stored Video

- Prerecorded videos are placed on servers.
- Users send requests to these servers to view the videos on-demand.
- The media is prerecorded, so the user may pause, reposition or fast-forward through video-content.
- Three categories of applications:
  1) UDP streaming
  2) HTTP streaming and
  3) Adaptive HTTP streaming.
- A main characteristic of video-streaming is the extensive use of client-side buffering.
- Two advantages of client-side buffering:
  1) Client-side buffering can mitigate effects of varying end-to-end delays
  2) This can mitigate effects of varying amounts of available bandwidth b/w server & client.

5.2.1 UDP Streaming

b)

5.2.4.2 CDN Types

- A CDN

  → manages servers in multiple geographically distributed locations
  → stores copies of the videos in its servers, and
  → attempts to direct each user-request to a CDN that provides the best user experience.

- The CDN may be a private CDN or a third-party CDN.

  1) Private CDN
  ➢ A private CDN is owned by the content provider itself.
  ➢ For example:
  Google's CDN distributes YouTube videos

  2) Third Party CDN
  ➢A third-party CDN distributes content on behalf of multiple content providers CDNs.
  ➢ Two approaches for server placement:

  i) Enter Deep
  ¤ The first approach is to enter deep into the access networks of ISPs.
  ¤ Server-clusters are deployed in access networks of ISPs all over the world.
  ¤ The goal is to get close to end users.
  ¤ This improves delay/throughput by decreasing no. of links b/w end user & CDN cluster

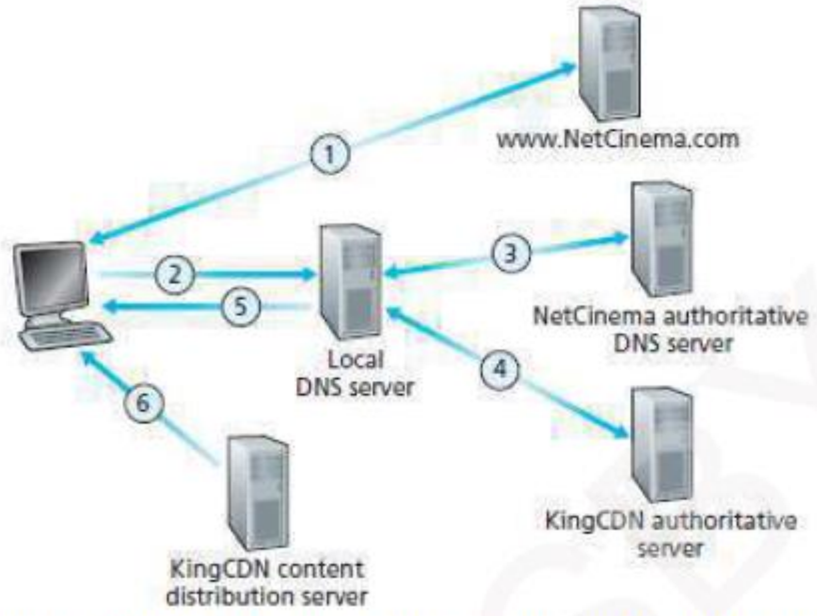
  ii) Bring Home
  ¤ The second approach is to bring the ISPs home.
  ¤ Large clusters are built at a smaller number of key locations.
  ¤ These clusters are connected using a private high-speed network.
  ¤ Typically, clusters are placed at a location that is near the PoPs of many tier-1 ISPs.
  For example: within a few miles of both Airtel and BSNL PoPs in a major city.
  ¤ Advantage:
  Lower maintenance and management overhead.
  ¤ Disadvantage:
  Higher delay and lower throughput to end users.

5.2.4.3 CDN Operation

- When a browser wants to retrieve a specific video, the CDN intercepts the request.
- Then, the CDN

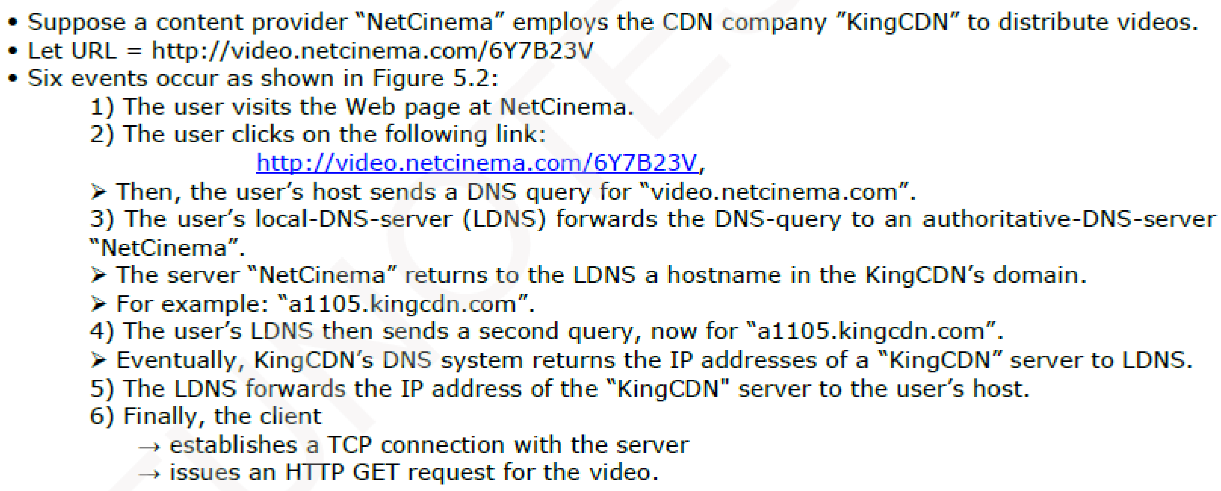
  1) determines a suitable server-cluster for the client and
  2) redirects the client's request to the desired server.

- Most CDNs take advantage of DNS to intercept and redirect requests.
- CDN operation is illustrated in Figure 5.2.



1
www.NetCinema.com
2
5
3
NetCinema authoritative DNS server
Local DNS server
4
6
KingCDN authoritative server
KingCDN content distribution server

Figure 5.2: DNS redirects a user's request to a CDN server



- Suppose a content provider "NetCinema" employs the CDN company "KingCDN" to distribute videos.
- Let URL = http://video.netcinema.com/6Y7B23V
- Six events occur as shown in Figure 5.2:

  1) The user visits the Web page at NetCinema.

  2) The user clicks on the following link:

  http://video.netcinema.com/6Y7B23V,

  ➢ Then, the user's host sends a DNS query for "video.netcinema.com".

  3) The user's local-DNS-server (LDNS) forwards the DNS-query to an authoritative-DNS-server "NetCinema".

  ➢ The server "NetCinema" returns to the LDNS a hostname in the KingCDN's domain.

  ➢ For example: "a1105.kingcdn.com".

  4) The user's LDNS then sends a second query, now for "a1105.kingcdn.com".

  ➢ Eventually, KingCDN's DNS system returns the IP addresses of a "KingCDN" server to LDNS.

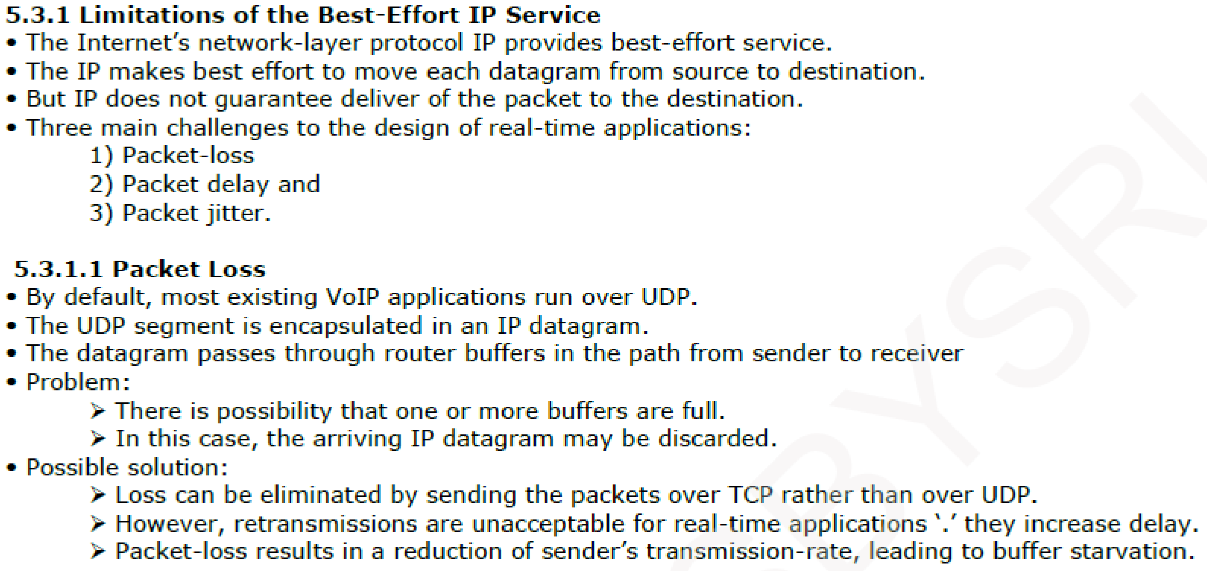
  5) The LDNS forwards the IP address of the "KingCDN" server to the user's host.

  6) Finally, the client

  → establishes a TCP connection with the server

  → issues an HTTP GET request for the video.

c)



5.3.1 Limitations of the Best-Effort IP Service

- The Internet's network-layer protocol IP provides best-effort service.
- The IP makes best effort to move each datagram from source to destination.
- But IP does not guarantee deliver of the packet to the destination.
- Three main challenges to the design of real-time applications:

  1) Packet-loss

  2) Packet delay and

  3) Packet jitter.

5.3.1.1 Packet Loss

- By default, most existing VoIP applications run over UDP.
- The UDP segment is encapsulated in an IP datagram.
- The datagram passes through router buffers in the path from sender to receiver
- Problem:

  ➢ There is possibility that one or more buffers are full.

  ➢ In this case, the arriving IP datagram may be discarded.
- Possible solution:

  ➢ Loss can be eliminated by sending the packets over TCP rather than over UDP.

  ➢ However, retransmissions are unacceptable for real-time applications '.' they increase delay.

  ➢ Packet-loss results in a reduction of sender's transmission-rate, leading to buffer starvation.

5.3.1.2 End-to-End Delay

- End-to-end delay is the sum of following delays:
  1) Transmission, processing, and queuing delays in routers.
  2) Propagation delays in links and
  3) Processing delays in end-systems.
- For VoIP application,
  → delays smaller than 150 msecs are not perceived by a human listener.
  → delays between 150 and 400 msecs can be acceptable but are not ideal and
  → delays exceeding 400 msecs can seriously hinder the interactivity in voice conversations.
- Typically, the receiving-side will discard any packets that are delayed more than a certain threshold.
- For example: more than 400 msecs.

5.3.1.3 Packet Jitter

- Jitter refers to varying queuing delays that a packet experiences in the network's routers.
- If the receiver
  → ignores the presence of jitter and
  → plays out audio-chunks,
  then the resulting audio-quality can easily become unintelligible.
- Jitter can often be removed by using sequence numbers, timestamps, and a playout delay

10.a)Explain the diffserv internet architecture (05)

b)Describe the Leaky bucket policing mechanism (06)

c) Discuss the round-robin and waited fair queuing scheduling mechanism (05)

Answer:

a)

5.5.3 DiffServ

- This provides QoS support to a broad class of applications.
- This provides service differentiation.
- Differentiation is defined as
  "The ability to handle different classes of traffic in different ways within the Internet".

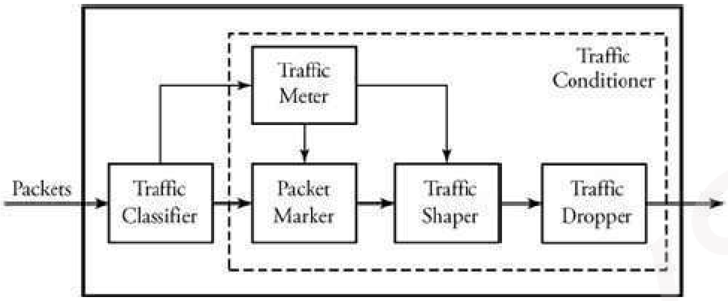


Figure 5.21: Overview of DiffServ operation

- The Diffserv architecture consists of 2 functional elements:

1) Packet Classification & Traffic Conditioning

- The traffic-classifier routes packets to specific outputs, based on the values of one or more header-fields.
- The traffic-profile contains a limit on the peak-rate of the flow.
- The traffic-conditioner detects and responds if any packet has violated the negotiated traffic-profile.
- The traffic-conditioner has 4 major components:

i) Meter

➢ The meter measures the traffic to make sure that packets do not exceed their traffic profiles

ii) Marker

➢ The marker marks or unmarks packets in order to keep track of their situations in the Diffserv node.

iii) Shaper

➢ The shaper delays any packet that is not complaint with the traffic-profile

iv) Dropper

➢ The dropper discards any packet that violates its traffic-profile

2) Core Function: Forwarding

- The per-hop behavior (PHB) is performed by Diffserv-capable routers.
- A router forwards marked-packet onto its next hop according to the PHB (per-hop behavior).
- PHB influences how network-resources are shared among the competing classes of traffic.
- Two types of PHB are: i) expedited forwarding and ii) assured forwarding.

i) Expedited Forwarding (EF) PHB

➢ This specifies that the departure rate of a class of traffic from a router must equal or exceed a configured rate.

ii) Assured Forwarding (AF) PHB

➢ This divides traffic into 3 classes: good, average and poor.

➢ Here, each class is guaranteed to be provided with some minimum amount of bandwidth and buffering.

PHB is defined as

"A description of the externally observable forwarding behavior of a Diffserv node applied to a particular Diffserv behavior aggregate"

From above definition, we have 3 considerations:

i) A PHB can result in different classes of traffic receiving different performance.

ii) A PHB does not dictate any particular mechanism for differentiating performance (behavior) among classes.

iii) Differences in performance must be observable and hence measurable.

b)

5.5.2.3 Policing: The Leaky Bucket

- Policing is an important QoS mechanism
- Policing means the regulation of the rate at which a flow is allowed to inject packets into the network.
- Three important policing criteria:

1) Average Rate

➢ This constraint limits amount of traffic that can be sent into n/w over a long period of time.

2) Peak Rate

➢ This constraint limits maximum no. of packets that can be sent over a short period of time

3) Burst Size

➢ This constraint limits the maximum no. of packets that can be sent into n/w over a very short period of time.

5.5.2.3.1 Leaky Bucket Operation

- Policing-device can be implemented based on the concept of a leaky bucket.
- Tokens are generated periodically at a constant rate.
- Tokens are stored in a bucket.
- A packet from the buffer can be taken out only if a token in the bucket can be drawn.
- If the bucket is full of tokens, additional tokens are discarded.
- If the bucket is empty, arriving packets have to wait in the buffer until a sufficient no. of tokens is generated.

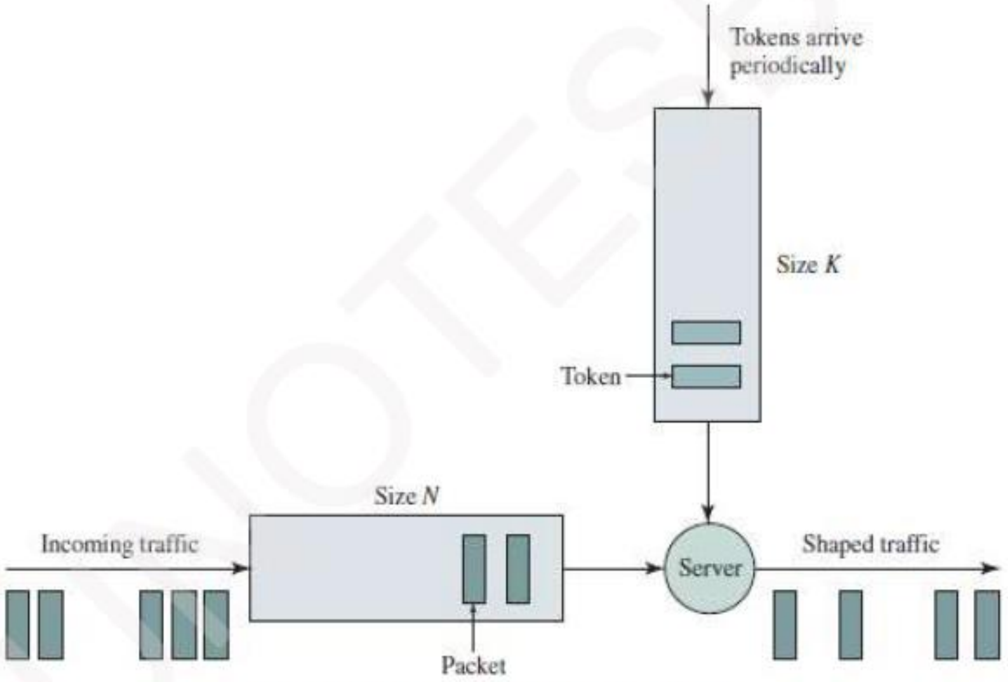


Figure 5.19: The leaky bucket policer

c)

5.5.2.2.3 RRQ

- RRQ (Round Robin Queuing) is illustrated in Figure 5.16 & Figure 5.17.
- The transmission bandwidth is divided equally among the buffers.
- Each user flow has its own logical buffer.
- Round-robin scheduling is used to service each non-empty buffer one bit at a time.
- In the simplest form, a class 1 packet is transmitted, followed by a class 2 packet, followed by a class 1 packet, followed by a class 2 packet, and so on.
- RRQ is a work-conserving queuing discipline.
- Thus, RRQ will immediately move on to the next class when it finds an empty queue.
- Disadvantage: Extensive processing at the destination.

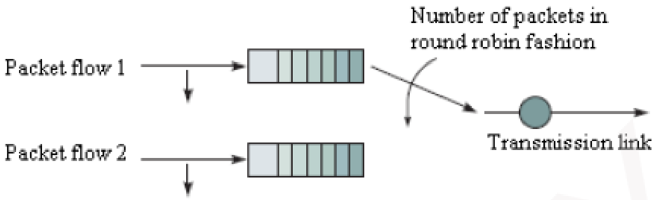


Figure 5.16: Round-robin queuing

Figure 5.16: Round robin queuing

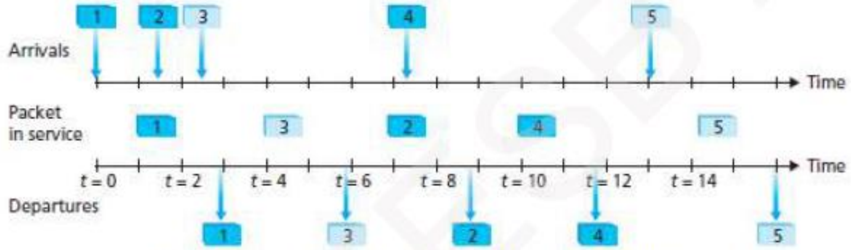


Figure 5.17: Operation of the two-class round robin queue

5.5.2.2.4 WFQ

- WFQ (Weighted Fair Queuing) is illustrated in Figure 5.18.
- Each user flow has its own buffer, but each user flow also has a weight that determines its relative share of the bandwidth.
- If buffer 1 has weight 1 and buffer 2 has weight 3, then buffer 1 will receive 1/4 of the bandwidth and buffer 2 will receive 3/4 of the bandwidth.
- In each round, each non-empty buffer would transmit a number of packets proportional to its weight.
- WFQ systems are means for providing QoS guarantees.
- WFQ is also a work-conserving queuing discipline.
- Thus, WFQ will immediately move on to the next class when it finds an empty queue.

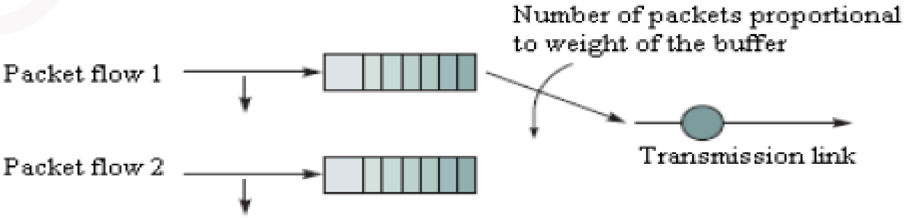


Figure 5.18: Weighted fair queuing