Subject: CN
Faculty: N.Shirisha
Topic:

Class Notes

Unit No: IV
Lecture No: L01
Link to Session Planner (SP): S.No. 43 of SP
Book Reference: T1
Date Conducted: 18/9/19
Page No.: 1

UNIT – IV

The Transport Layer

Its task is to provide reliable, cost-effective data transport from the source machine to the destination machine.

The Transport Service:

Services provided to the upper layers:

The ultimate goal of Transport layer is to provide, efficient, reliable and cost-effective service to its users, normally processes in the application layer. The hardware and/or software with in the transport layer. The hardware and/or software with in the transport layer that does the work is called the transport entity.

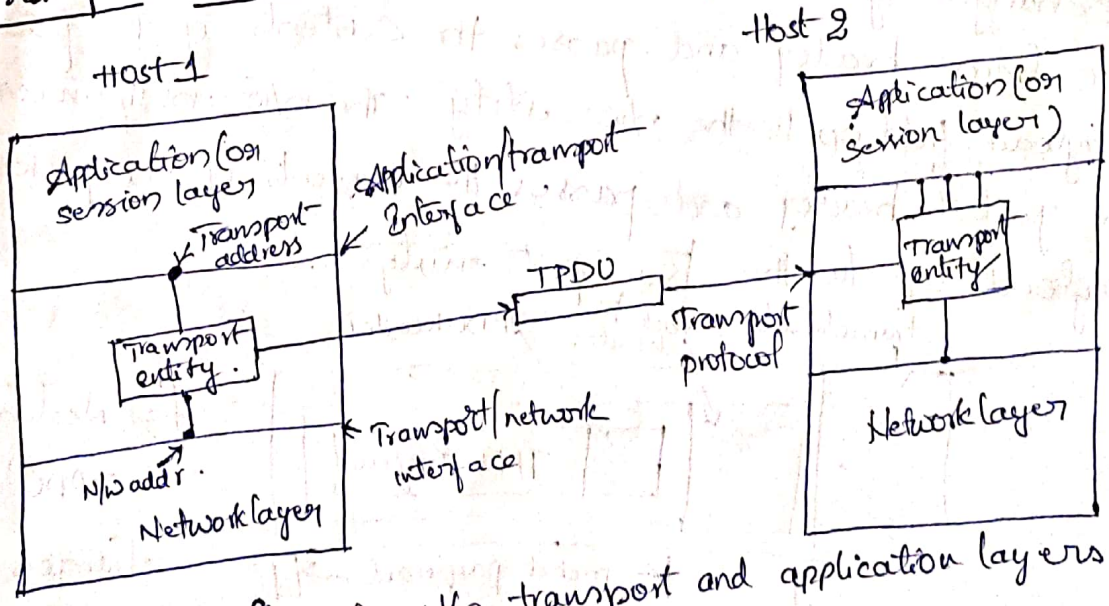


fig:- the N/w, transport and application layers.

Transport Service Primitives :.

To allow users to access the transport service, the transport layer must provide some operations to application programs, ie., a transport service interface.

primitive	Packet sent	Meaning
LISTEN	(none)	- Block until some process tries to connect
CONNECT	CONNECTION REQ.	- Actively attempt to establish a connection
SEND	DATA	- Send information
RECEIVE	(none)	- Block until a DATA packet arrives
DISCONNECT	DISCONNECTION REQ.	- This side wants to release a connection

fig: primitives for a simple transport service.

→TPDU (Transport protocol Data Unit) is used for messages sent from transport entity to transport entity. Thus, TPDUs are contained in packets. In turn, packets are contained in frames. (exchanged by dll). When a frame arrives, the dll processes the frame header and passes the contents of the frame payload field up to the n/w entity. The n/w entity processes the packet header and passes the contents of the packet payload up to the transport entity.

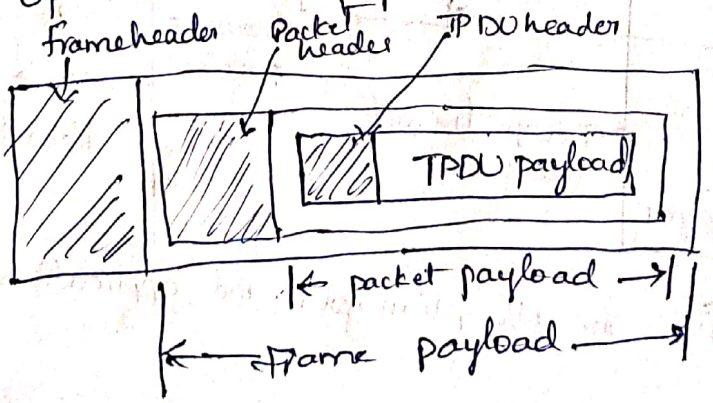


fig: Nesting of TPDU's, Packets, Frames.

Subject: CN
Faculty: N.Shirisha
Topic:

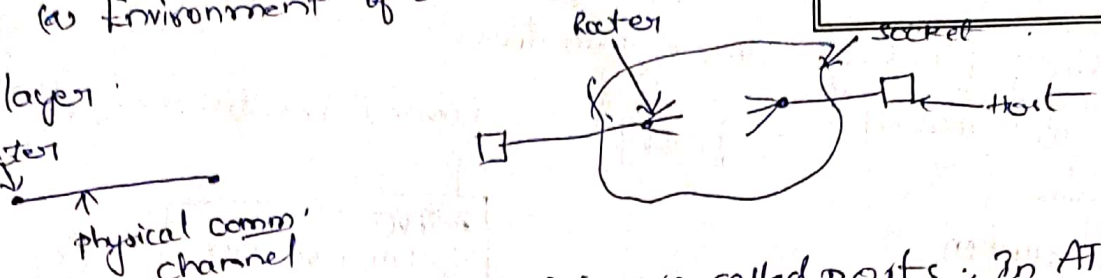
Class Notes

Unit No: IV
Lecture No: L 02
Link to Session Planner (SP): S.No. 44 of SP
Book Reference: T1
Date Conducted: 19/9/19
Page No: 2

Elements of Transport Protocols :-

The transport service is implemented by a transport protocol used between the two transport entities.

fig: (a) Environment of dll layer (b) Environment of the transport layer



Addressing :- In the Internet, end points are called ports; In ATM n/w, they are called AAL-SAPS; TSAP (Transport service Access point) & NSAP (n/w layer-NSAPs). TSAPs, NSAPs, and transport Connections.

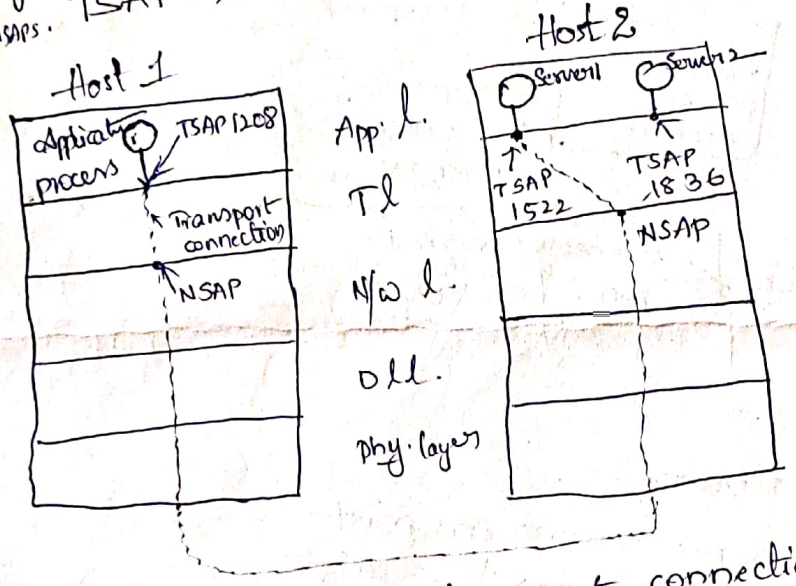


fig: TSAPs, NSAPs and transport connections.

A possible scenario for a transport connection is as follows:

1. A time of day server process on host 2 attached itself to TSAP 1522 to wait for an incoming call.
2. An application process on host 1 wants to find out the time-of-a-day so it issues a CONNECT request specifying TSAP 1208 as the source and TSAP 1522 as the destination. This action ultimately results in a transport connection being established b/w the app. process on host 1 and server1 on host 2.
3. The app. process then sends over a request for the time.
4. The time server process responds with the current time.
5. The transport connection is then released.

A state diagram for connection establishment and release for these simple primitives.

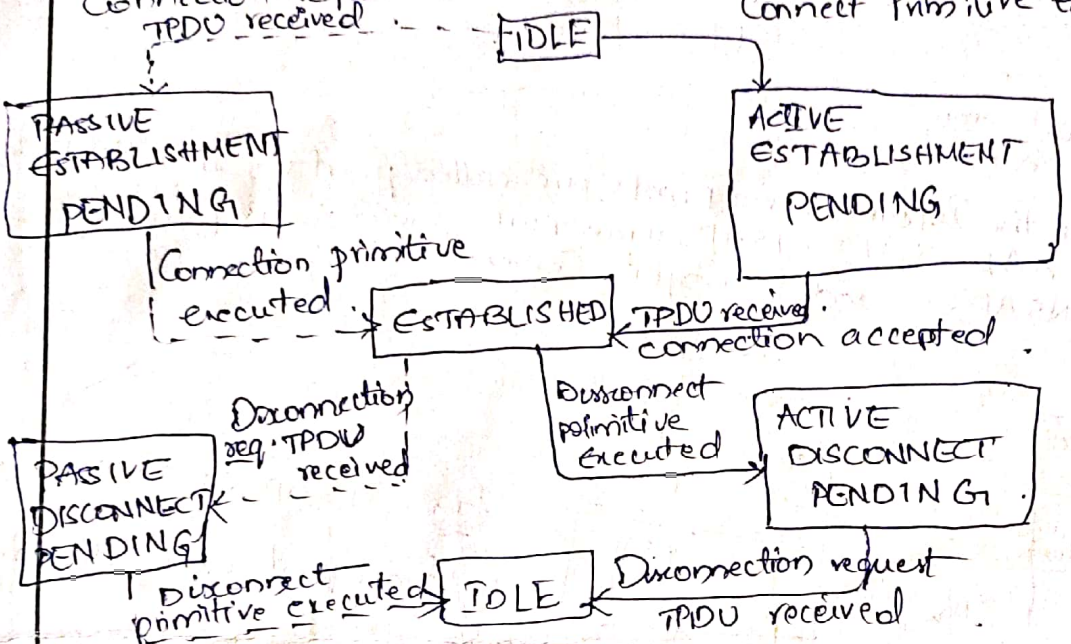


fig :- A state diagram for a simple connection management scheme.

Berkey Sockets :

primitive	Meaning
SOCKET	Create a new comm. end point
BIND	Attach a local address to a socket
LISTEN	announcing willingness to accept connections; give queue size
ACCEPT	Block the caller until a connects - attempt arrives
CONNECT	Actively attempt to establish a connection
SEND	Send some data over the connection
RECEIVE	Receive some data from the connection
CLOSE	Release the connection.

These socket primitives used in Berkeley UNIX for TCP. These primitive are widely used for Internet programming

Subject: CN
Faculty: N.Shirisha
Topic: Elements of Transport protocol

Class Notes

Unit No: IV
Lecture No: L02
Link to Session L04
Planner (SP): S.No.... of SP
Book Reference: T1
Date Conducted: 19/8/19
Page No: 3

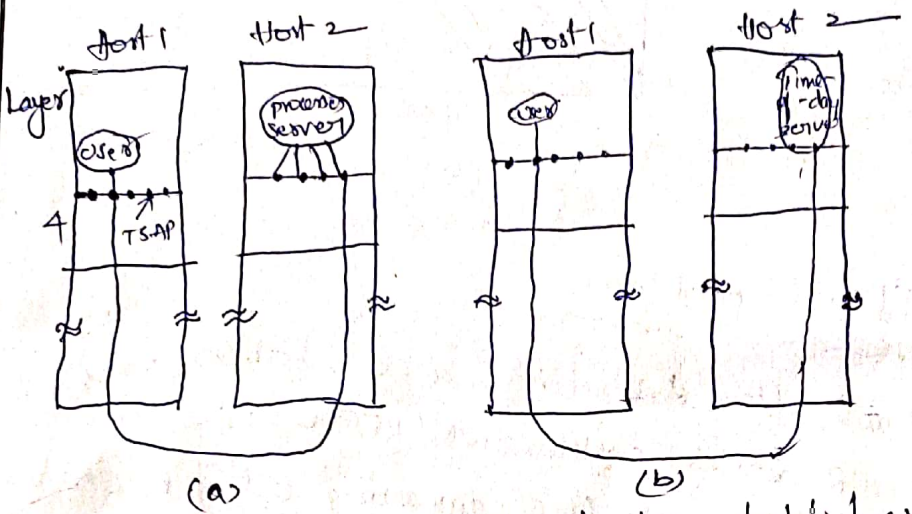


Fig: How a user process in host 1 establishes a connection with a time-of-day server in host 2.

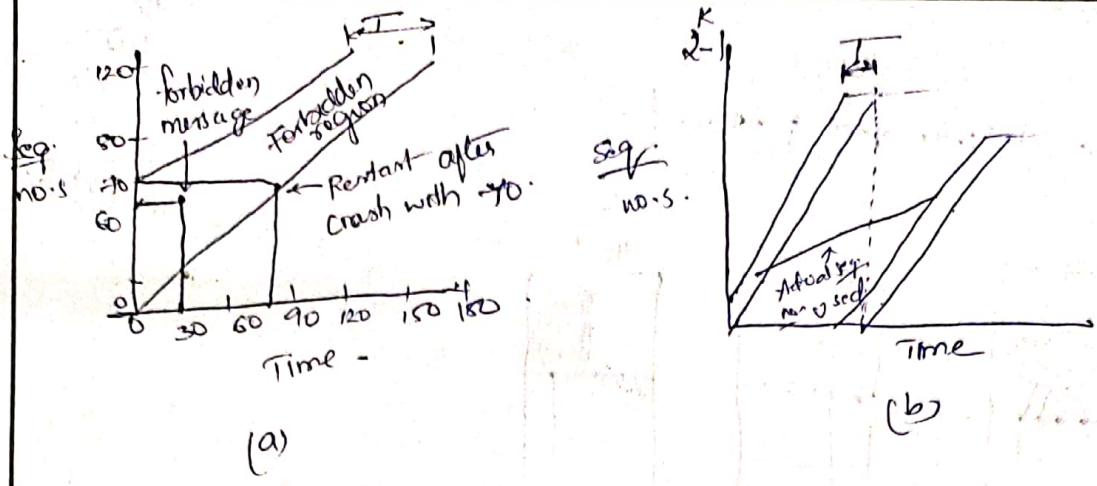
Connection Establishment:

Another possibility is to give each connection a connection Identifier (i.e., a sequence number incremented for each connection established) chosen by the initiating party and put in each TPDU, including the one requesting the connection.

Instead, we need to make a different tack. Rather than allowing packets to live forever within the subnet, we must devise a mechanism to kill off aged packets that are still hobbling about.

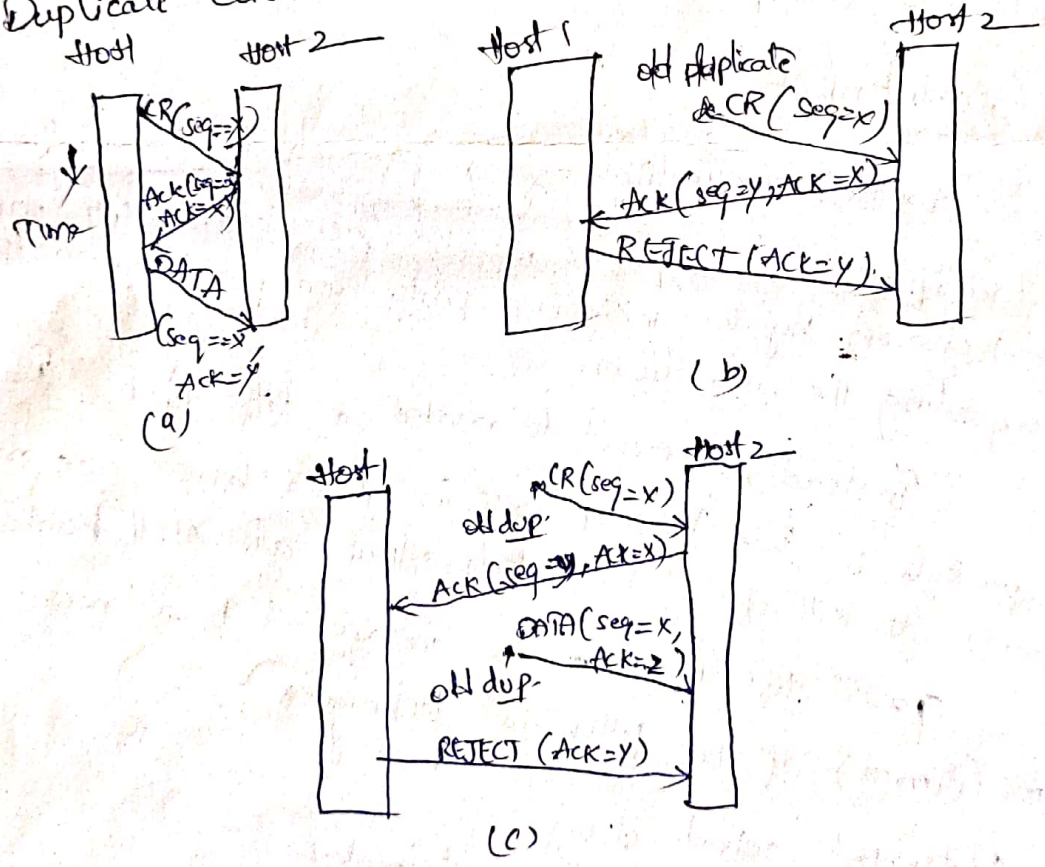
Packet lifetime can be restricted to a known maximum using one (or more) of the following techniques.

1. Restricted subnet design
2. Putting a hop counter in each packet
3. Time Stamping each packet.



Three-way handshake problem :

Three protocol scenarios for establishing a connection using a 3-way handshake. CR denotes CONNECTION REQUEST. (a) Normal operation (b) Old duplicate CONN. REQ appearing out of nowhere. (c) Duplicate CONN. REQ. and DUP. Ack.



↳ Expense / learning

Skill or will → Attitude

19-01-21

Subject: CN
Faculty: N.Shirisha
Topic: Elements of transport protocol

Class Notes

Unit No: IV
Lecture No: L23
Link to Session
Planner (SP): S.No. 45 of SP
Book Reference: T1
Date Conducted: 23/9/19
Page No: 4

CONNECTION RELEASE

There are two ways of terminating a connection:

(i) Asymmetric release and
(ii) Symmetric release.

- Asymmetric release is the way the telephone system works: when one party hangs up, the connection is broken.
- Symmetric release treats the connection as two separate unidirectional connections and requires each one to be released separately.

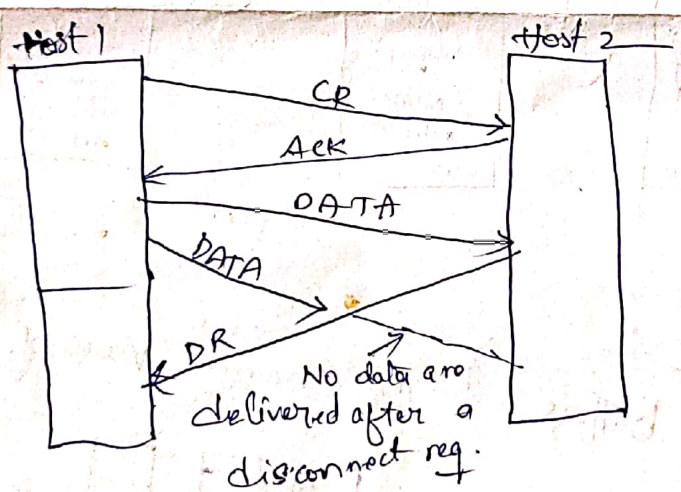


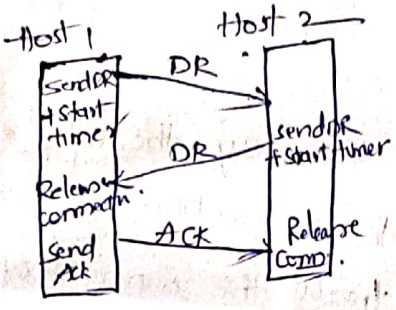
fig: Abrupt disconnection with loss of data.

Eg: Two-army problem:

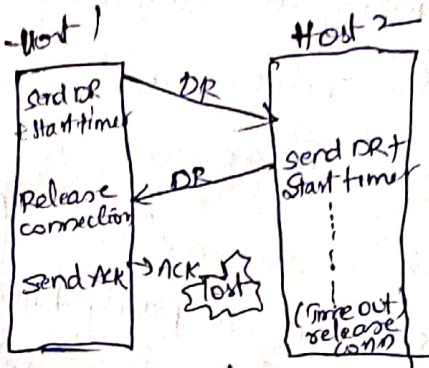
The blue armies want to synchronize their attacks. However, their only comm. medium is to send messengers on foot down into the valley, where they might be captured and the message lost.

fig: four protocol scenarios for releasing a connection.

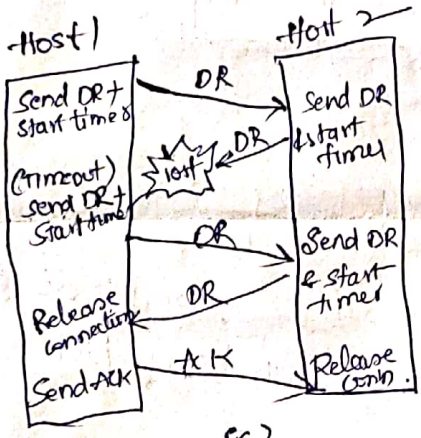
(a) Normal case of three-way handshake
(b) final ACK lost
(c) Response lost
(d) Response lost and subsequent DRs lost.



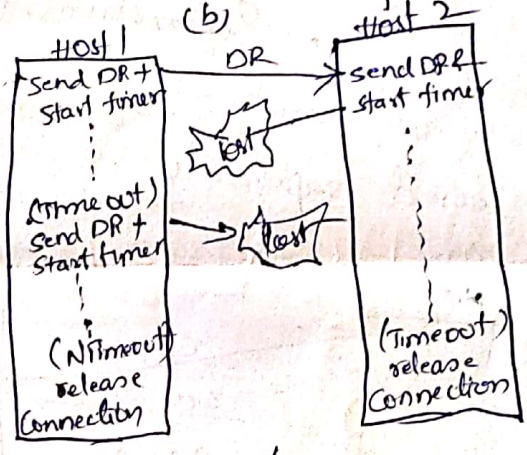
(a)



(b)



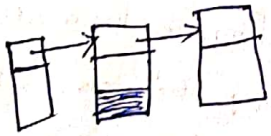
(c)



(d)

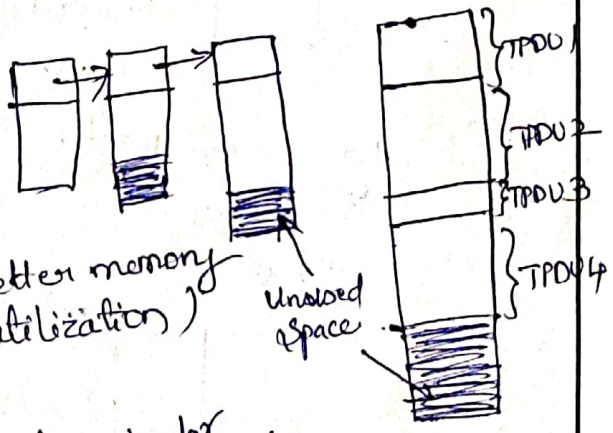
Flow Control and Buffering:

(a) Chained fixed-size buffers



(a)

(b) Chained Variable-sized buffers



(c) one large circular buffer per connection.

(Good use of memory, provided all connections are heavily loaded, but is poor if some are lightly loaded.)

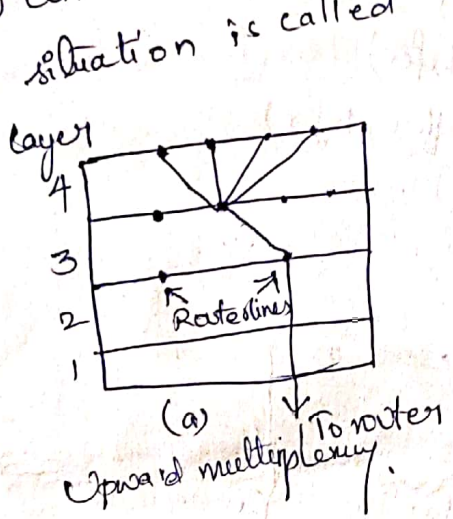
Subject: CN
Faculty: N.Shirisha
Topic: Elements of transport protocol

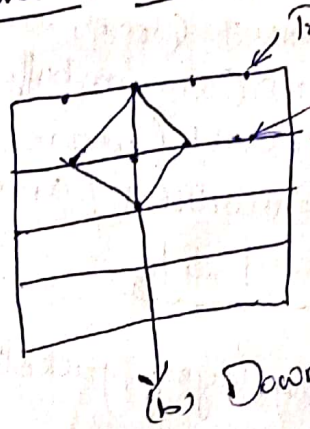
Class Notes

Unit No: IV
Lecture No: 1 & 2
Link to Session
Planner (SP): S.No. 46 of SP
Book Reference: T1
Date Conducted: 29/9/19
Page No: 5 host, all

Multiplexing :

If only one n/w address is available on a host, all transport connections on that machine have to use it. When a TPDU comes in, some way is needed to tell which process to give it to. This situation is called Upward Multiplexing.





If a subnet uses virtual circuits internally and imposed a maximum data rate on each one. If a user needs more bandwidth than one virtual circuit can provide, a way out is to open multiple n/w connections and distribute the traffic among them on a round-robin basis, is called downward multiplexing.

Crash Recovery :

If the hosts and routers are subject to crashes, recovery from these crashes becomes an issue.

The server can be programmed in one of two ways: ack first write first. The client can do it in one of four ways!- ① always retransmit the last TPDU ② never retransmit ③ retransmit only in state S1. Three events are possible at the server sending an Ack (A), writing to the O/P process (W), & crashing (C), and the three events can occur in six diff. orderings: AC(W), AWC, C(AW), C(WA), WAC, WC(A), where parentheses are used to indicate that neither A nor W can follow C.

A Simple Transport Protocol:

- Example Service Primitives :-
  1. Connect
  2. Listen
  3. disconnect
  4. send
  5. receive

The parameters for the service primitives and library procedures are as follows:

```
connum = LISTEN (local)
connum = CONNECT (local, remote)
status = SEND (connum, buffer, bytes)
status = RECEIVE (connum, buffer, bytes)
status = DISCONNECT (connum)
```

- Example Transport Entity :-

The n/w layer packets used in our ex.

N/w packet	Meaning
CALL REQUEST	send to establish a connection
CALL ACCEPTED	Response to CALL REQUEST
CLEAR REQUEST	sent to release a connection
CLEAR CONFIRMATION	Response to CLEAR REQUEST
DATA	used to transport data
CREDIT	Control packet for managing the window.

- Example as a finite state machine:

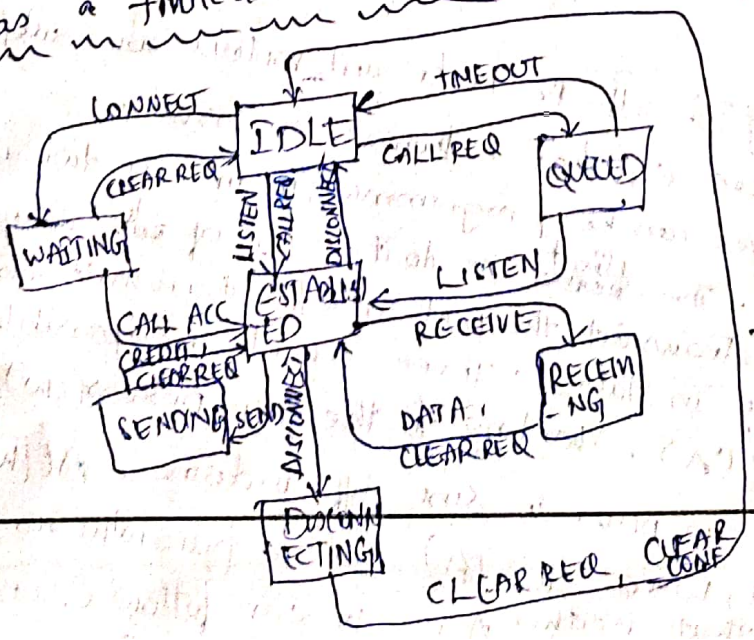


Fig:- Example protocol in graphical form. Transitions that leave the connection state unchanged have been omitted for simplicity.

A Simple Transport protocol (contd..)

Subject: CN
Faculty: N.SHIRISHA
Topic: Crash recovery (contd.:)

Class Notes

Unit No: IV
Lecture No: L003
Link to Session
Planner (SP): S No. 46 of SP
Book Reference: J1
Date Conducted: 25/9/19
Page No: 1

Strategy used by Sending host	First Ack, then Write			First write, then Ack		
	A(W)	AWC	C(AW)	C(W A)	WAC	W C(A)
Always retransmit	OK	DUP	OK	OK	DUP	DUP
Never retransmit	LOST	OK	LOST	LOST	OK	OK
Retransmit in S0	OK	DUP	LOST	LOST	DUP	OK
Retransmit in S1	LOST	OK	OK	OK	OK	DUP

OK = protocol functions Correctly

DUP = protocol generates a duplicate message

LOST = protocol loses a message.

fig: Diff. Comb. of Client and Server Strategy.

Subject: CN
Faculty: N.Shirisha
Topic:

Class Notes

Unit No: IV
Lecture No: L05
Link to Session 47
Planner (SP): S.No.... of SP
Book Reference: T1
Date Conducted: 24/9/19
Page No: 7

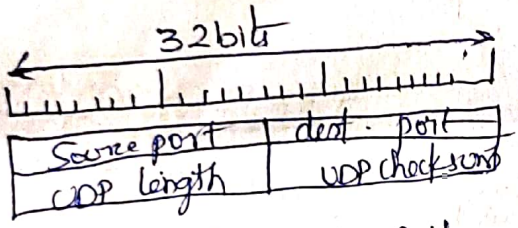
INTERNET TRANSPORT PROTOCOLS: UDP :

The Internet has two main protocols in the transport layer, a connection-less, which is UDP and Connection-oriented which is a TCP.

Introduction to UDP :

UDP (user datagram protocol). UDP provides a way for applications to send encapsulated IP datagrams and send them without having a establish a connection. UDP is described in RFC768.

- UDP transmits segments consisting of an 8-byte header followed by the payload.



Remote Procedure Call :

When a procces on m/c 1 calls a procedure on m/c 2, the calling process on 1 is suspended and execution of the called procedure takes place on 2. Info. can be transported from the caller to the callee in the parameters and can come back in the procedure result. No Message passing is visible to the programmer. This technique is known as RPC.

In Simplest form, to call a remote procedure, the client program must be bound with a small library procedure, called the Client Stub, that represents the server procedure in the Client's address space. Silly, the Server is bound with a procedure called the Server Stub.

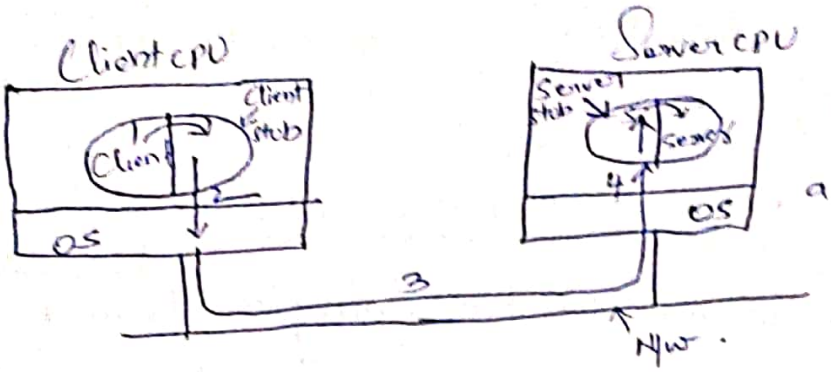
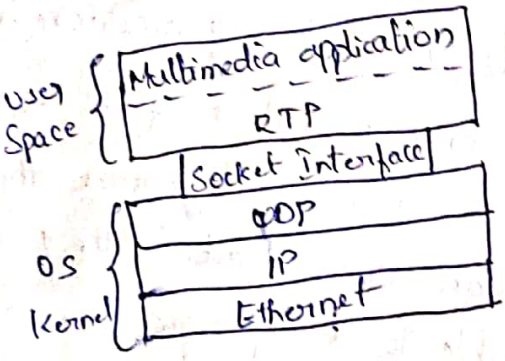


fig: Steps in making a RPC. The stubs are shaded.

Real-time Transport Protocol:

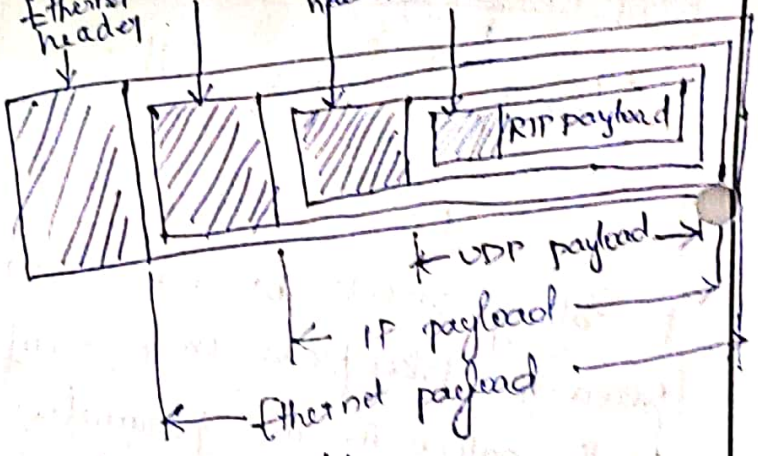
Client-server RPC is one area in which UDP is widely used. Another one is real-time multi-media applications. In particular, as Internet radio, Internet telephony, music-on-demand, videoconferencing, video-on-demand, & other multi-media applications became more common place, people discovered that each application was inventing more or less the same real-time transport protocol. Therefore, RTP() born. It is described in RFC 1889.

fig: (a) The Position of RTP in the protocol stack.



(a)

(b) Packet nesting



(b)

The basic function of RTP is to multiplex several real-time data streams onto a single stream of UDP packets. The UDP stream can be sent to a single-destination (unicasting) or to multicasting (multiple destinations).

Subject: CN
Faculty: N.Shirisha
Topic: Internet transport protocol UDP.

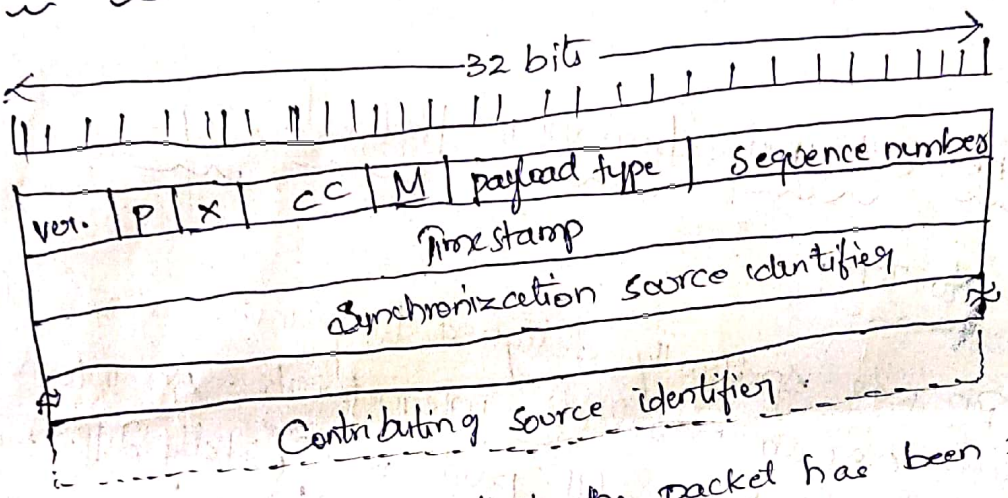
Class Notes

Unit No: IV
Lecture No: L 015
Link to Session
Planner (SP): S.No. of SP 48
Book Reference: T1
Date Conducted: 25/9/19
Page No: 8

RTP (contd..)

- RTP has no flow control, no error control, no acknowledgement no mechanism to request retransmission.

RTP header:



P- The P bit indicates that the packet has been padded to a multiple of 4 bytes. The last padding byte tells how many bytes were added.

X- X bit indicates that an extension header is present, the first word of the extension gives the length. This is an escape hatch for any unforeseen requirements.

CC- field tells how many contributing sources are present, from 0 to 15.

M- It is an application-specific marker bit. It can be used to mark the start of the video frame.

Payload type - field tells which encoding algo: has been used.

Sequence number - is just a counter that is incremented on each RTP packet sent. It is used to detect lost packets.

The Synchronization source identifier tells which stream the packet belongs to. It is the method used to multiplex and demultiplex multiple data streams onto a single stream of UDP packets.

- The Contributing source identifiers, if any, are used when mixers are present in the studio.

- RTP has a little sister protocol - RTCP (Real time control protocol. It handles feedback, synchronization, and the user interface but does not transport any data.

The Internet Transport Protocols: TCP

Introduction to TCP:

TCP was specifically designed to provide a reliable end-to-end byte stream over an unreliable internetwork. An internetwork differs from a single n/w b'coz diff. parts may have wildly diff. topologies, bandwidths, delays, packet sizes, intern/w and to be robust in the face of many kinds of failures.

TCP was formally defined in RFC 793.

TCP Service Model:

- TCP Service is obtained by both the sender and receiver creating end points, called Sockets.

- Each socket has a Socket no. consisting of the IP address of the host and a 16-bit no. local to that host, called a port.

- A port is a TCP name for the TSAP. For TCP service to be obtained, a connection must be explicitly established b/w a socket on the sending m/c & a socket on receiving m/c.

Subject: CN
Faculty: N.Shirisha
Topic: ITP (TCP) :

Class Notes

Unit No: IV
Lecture No: L66
Link to Session Planner (SP): S.No. 48 of SP
Book Reference: T1
Date Conducted: 25/9/19
Page No: 9

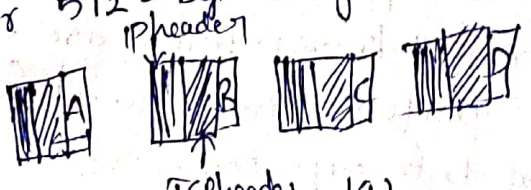
The TCP Service model. (contd..)

- Posit numbers below 1024 are called well-known ports and are reserved for standard services.

port	protocol	use
21	FTP	file transfer
23	Telnet	Remote login
25	SMTP	E-mail
69	TFTP	Trivial
79	Finger	lookup info. about a user
80	HTTP	www
110	POP-3	Remote-email access
119	NNTP	USENET news

All TCP connections are full duplex and point-to-point.
full duplex means that traffic can go in both directions at the same time
Point-to-point means that each connection has exactly two end pts.
TCP does not support multicasting or broadcasting.

A TCP connection is a byte stream, not a message stream.
Message boundaries are not preserved end-to-end.
Eg: (a) four 512-byte segments sent as separate IP datagrags.



(a)

A B C D

(b)

(b) The 2048 bytes of data delivered to the application in a single READ call.

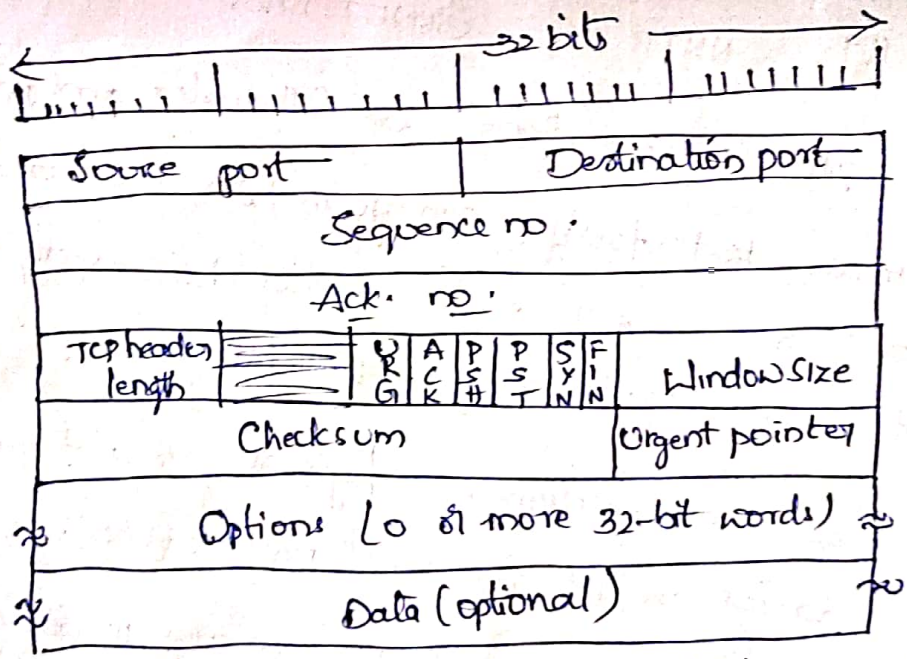
- The TCP protocol :

- The key feature of TCP, and one which dominates the protocol design, is that every byte on a TCP connection has its own 32-bit seq. no.

- The sending and receiving the TCP entities exchange data in the form of segments. A TCP segment consists of a fixed 20-byte header followed by zero or more data bytes.

- The basic protocol used by TCP entities is the sliding window protocol. When a sender transmits a segment, it also starts a timer.

The TCP Segment header :



- The source port and Destination port fields identify the local end points of the connection. The well-known ports are defined.

- Seq. no. & ack no. fields perform their usual functions. The latter specifies the next byte expected.

Subject: CN
Faculty: N.Shirisha
Topic:

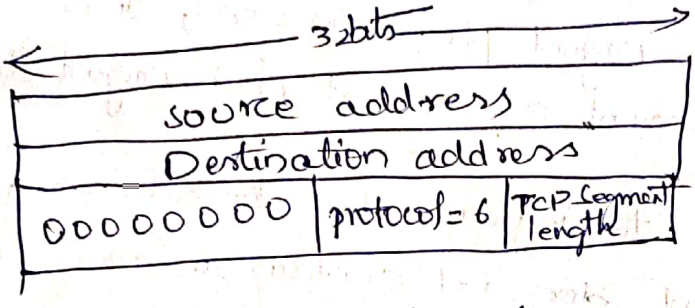
Class Notes

Unit No: IV
Lecture No: L 47
Link to Session
Planner (SP): S.No. 46 (Sep)
Book Reference:
Date Conducted: 9/9/19
Page No: 10

The TCP segment Header (contd..)

- The TCP header length tells how many 32-bit words are contained in the TCP header. This info is needed bcoz the options field is of variable length.
- URG is set to 1 if the urgent ptr. is in use, it is to indicate a byte offset from the current seq. no. at which urgent data are to be found.
- The Ack bit is set to 1 to indicate that the Ack no. is valid. If Ack is 0, the segment does not contain an ack. so the Ack. no. field is ignored.
- The PSH bit indicates pushed data. The receiver is hereby kindly requested to deliver the data to the application upon arrival and not buffer it until a full buffer has been received.
- The RST bit is used to reset a connection that has become confused due to host crash.
- The SYN bit is used to establish connections. The connection request has SYN = 1 and ACK = 0 to indicate that the piggyback ack. field is not in use. The connection reply does bear an ack. so it has SYN = 1 and ack = 1. In essence, the SYN bit is used to denote CONNECTION REQUEST and CONNECTION ACCEPTED, with the ACK bit used to distinguish b/w those two possibilities.
- The FIN bit is used to release a connection. It specifies that the sender has no more data to transmit. Both SYN and FIN segments have seq. nos. and are thus guaranteed to be processed in the correct order.
- Flow control in TCP is handled using variable-sized sliding window.

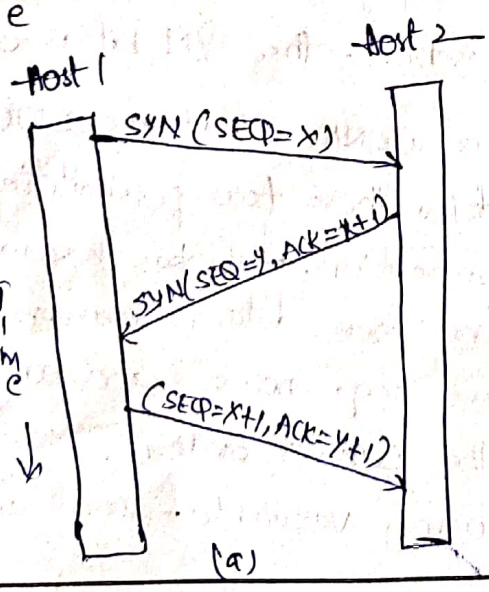
- The window-size - tells how many bytes may be sent starting at the byte acknowledged.
- A Window size field of 0 is legal and says that the bytes up to and including Acknowledgement no. - 1 have been received.
- Checksum is also provided for extra reliability. It checksums the header, the data, & the conceptual pseudo header.

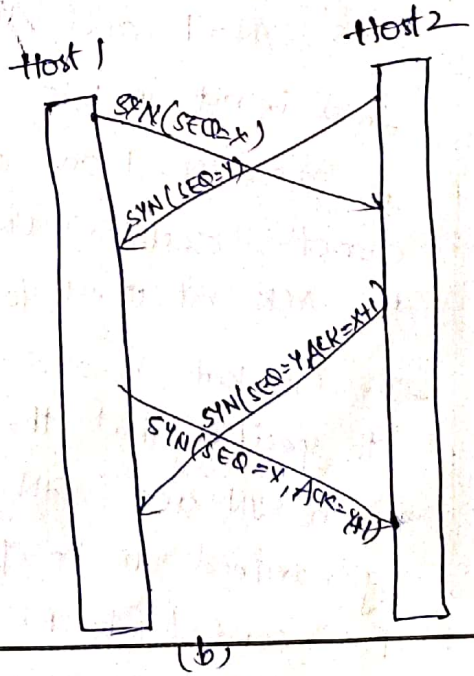


TCP Connection Establishment :

- Connections are established in TCP by means of the three-way handshake.
- To establish a connection, one side, server, passively waits for an incoming connection by executing the LISTEN and ACCEPT primitives.

(a) TCP Connection establishment in the normal case (b) Call collision.





Subject: CN
Faculty: N.Shirisha
Topic:

Class Notes

Unit No: IV
Lecture No:
Link to Session: 2 3-way
Planner (SP): S.No. 49 SP
Book Reference: hake 5
Date Conducted: 25/9/19
Page No: 11

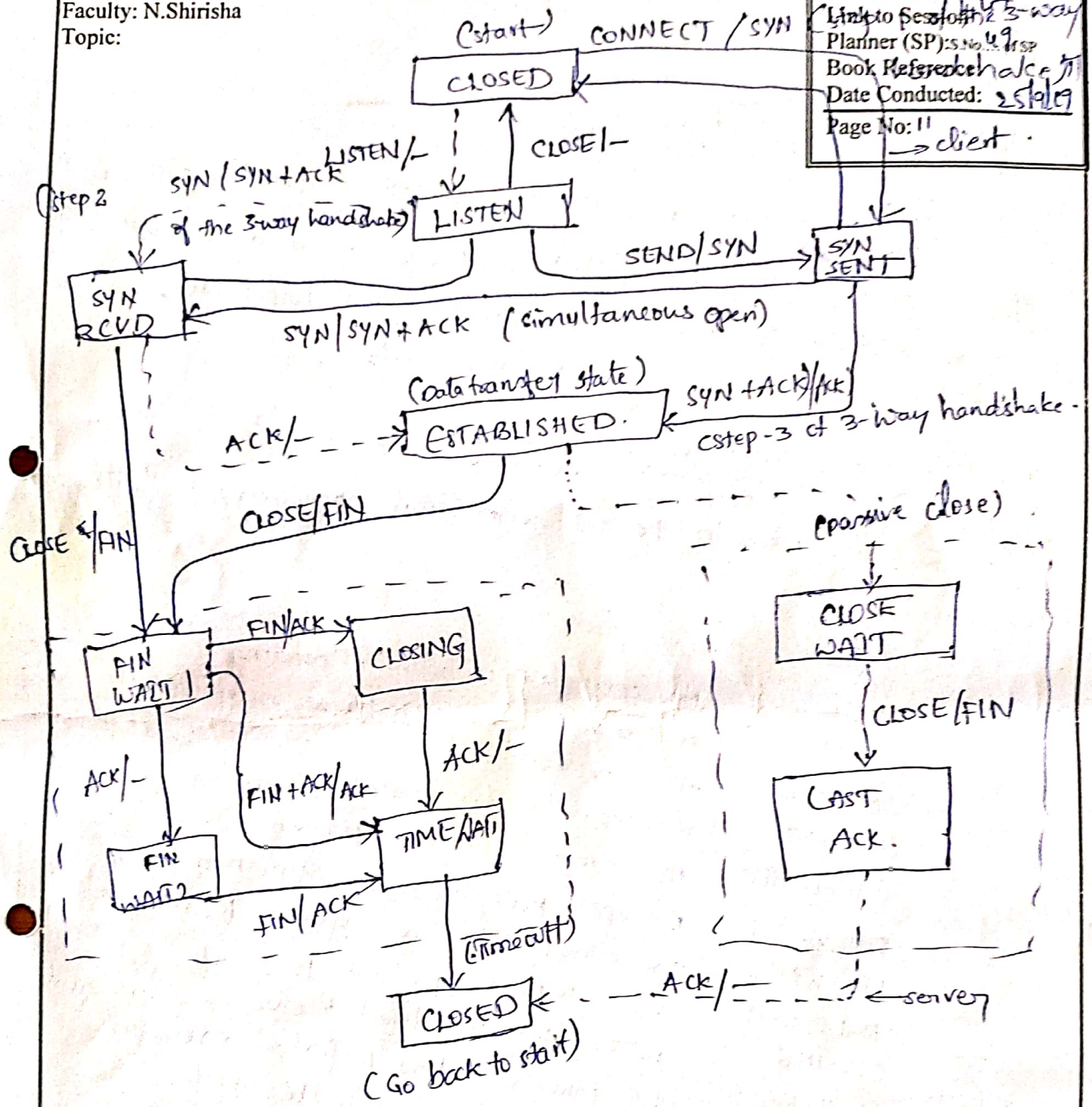


fig: TCP connection mngmnt finite state m/c.

TCP Connection Release:

To release a connection, either party can send a TCP segment with the FIN bit set, which means that it has no more data to transmit. When a FIN bit is acknowledged, that direction is shut down for new data.

TCP Connection Management Modeling:

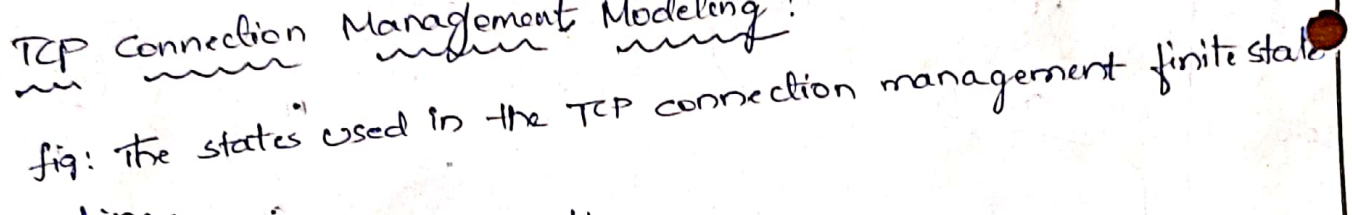


fig: The states used in the TCP connection management finite state machine.

State	Description
CLOSED	No connection is active or pending.
LISTEN	The server is waiting for an incoming call
SYN RCVD	A connection request has arrived; wait for ACK
SYN SENT	The application has started to open a connection
ESTABLISHED	The normal data transfer state.
FIN WAIT 1	The app. has said it is finished.
FIN WAIT 2	The other side has agreed to release
TIMED WAIT	Wait for all packets to die off.
CLOSING	Both sides have tried to close simultaneously
CLOSE WAIT	The other side has initiated a release
LAST ACK	Wait for all packets to die off.