

## Module - IV

### Transport Layer

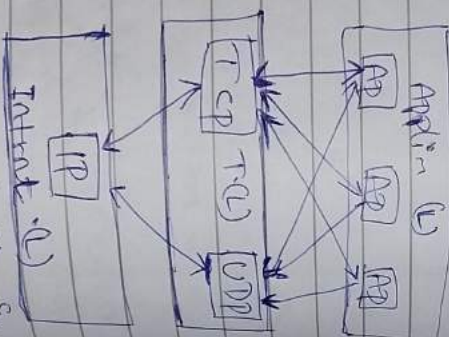
- \* T. layer located b/w app'n (L) & n/w (L).
- \* The ultimate goal of transport (L) is to provide efficient, reliable & cost-effective process-to-process data transmission to process in the app'n (L).
- \* It makes use of services provided by the n/w (L).

\* Main role of T. (L) is to provide common services directly to the app'n (Ps) running on different hosts.

#### \* Services -

##### 1) End-to-end delivery:

T. (L) transmits the entire msg to the destination.  
i.e., it ensures the E-T-E delivery of entire msg from S to destination.



##### 2) Reliable delivery:

Provides reliability services by retransmitting the lost & damaged packets.

##### 3) Flow control:

If the receiver is overloaded with too much data, then the receiver discards the packets & asking for the retransmission of packets.

1) Multiplexing: uses the multiplexing to improve transmission efficiency.

=> T. (L) protocols = 4

- \* Represented by protocols - TCP & UDP
- \* IP protocol in n/w (L) delivers a datagram from a source host to the destination host.
- \* msg-oriented transport (L) -> SCTP (Stream Control Transmission Protocol) that combines the features of UDP & TCP.

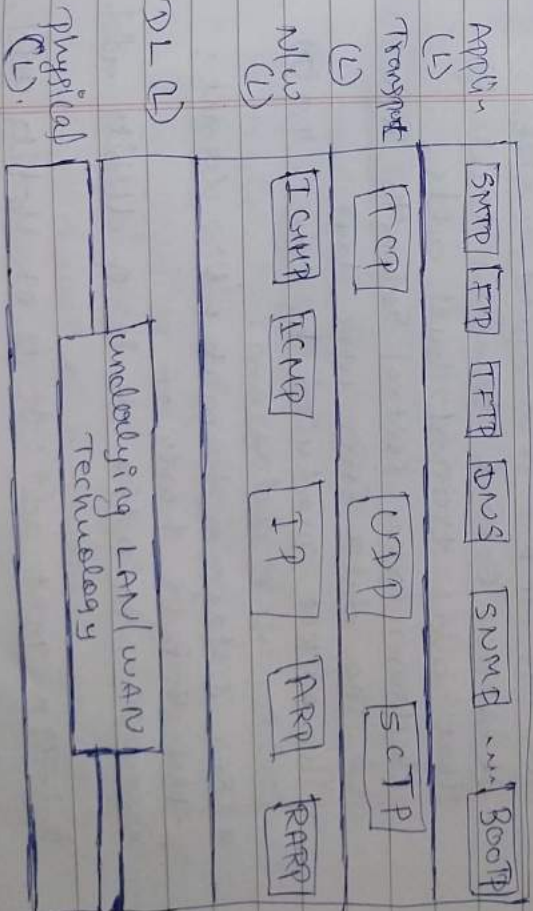
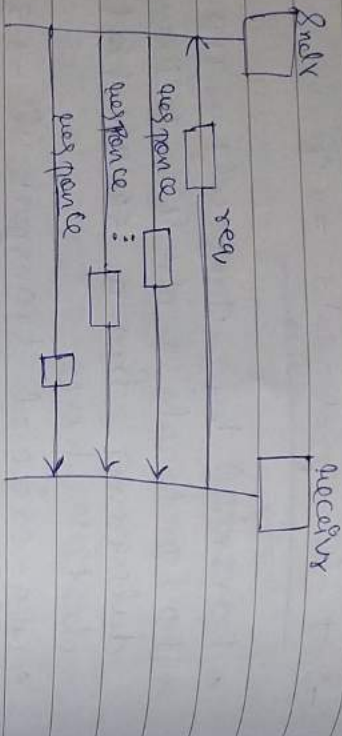


fig: T. (L) 4



# 1 UDP [user Datagram (P)] :

## Functioning of UDP



- \* It is a conn<sup>n</sup> less (P) & this type of (P) is used when reliability & security are less imp than speed & size
- \* It is an e-to-e transport level (P) that adds transport level adds, checksum error control & length info to the data from upper layer.
- \* The packet produced by the UDP (P) → a user datagram.
- \* UDP Datagram Format : / UDP Segment :
  - flag 8-byte header —
  - source port add : It is of 16 bits add
  - Dest n port add : It is of 16 bits add

## 2 checksum :

16-bit field which is used in error correc<sup>n</sup>

### Length :

16-bit field that defines the total length of user datagram, header plus data.

### \* Services —

- 1 (P) - to - (P) connection (2) connection services
- 3) Multiplexing & demultiplexing (4) flow control (5) error control.
- \* Adv → fast delivery of msgs
- \* Dis → Lack of error control.

## II TCP [Transmission Control (P)] :

- \* provides t. (P) services to appl<sup>n</sup> & is a conn<sup>n</sup>-oriented (P).
- \* A secured conn<sup>n</sup> is being established b/w the snder & receiver.
- \* For creating the conn<sup>n</sup>, TCP generates a virtual circuit b/w snder & receiver for the dur<sup>n</sup> of a trans<sup>n</sup>.

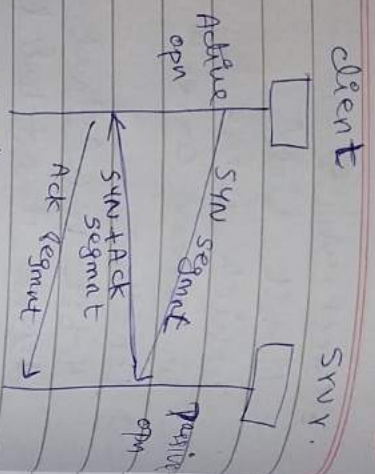
### UDP Header

8 bytes		UDP data	
source port	16 bits	desti <sup>n</sup> -port	16 bits
length	16 bits	checksum	16 bits

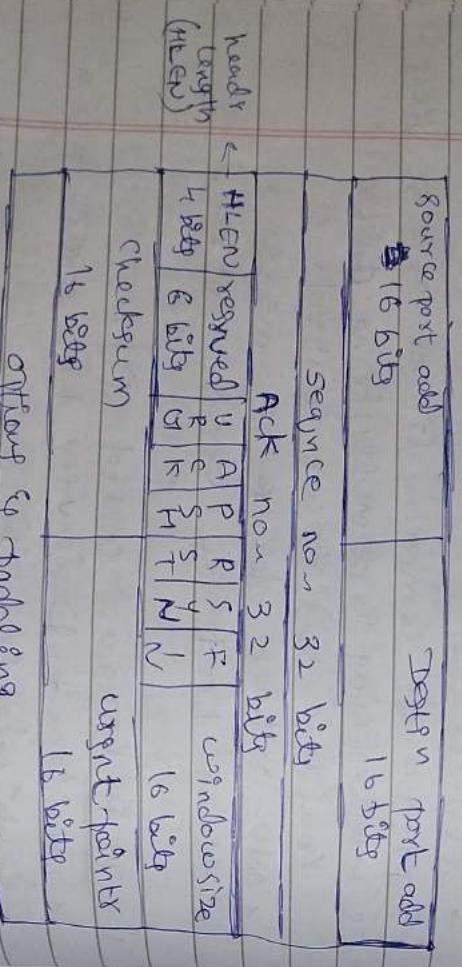


\* Features -

- reliable full duplex.
- order of data is maintained
- conn<sup>n</sup> oriented.



\* TCP segment - TCP Header :



- source port add → to define add of the app<sup>n</sup> prog<sup>n</sup> in a source comp.
- sequence num → represents the position of the data in an original data stream.
- Ack num → Acked data from other communicating devices.
- Control bits → URG, ACK, PSH, RST, SYN, FIN, urgent, window, pressing, window size, window scaling.

\* TCP conn<sup>n</sup> → conn<sup>n</sup> establishment

- Data transfer
- conn<sup>n</sup> termination.

III SCTP :

- \* It is a conn<sup>n</sup> oriented p<sup>r</sup> in comp<sup>n</sup> (N) which provides a full duplex association.
- \* It is a new T.C.P. designed to combine some features of UDP & TCP to create a better p<sup>r</sup> for multimedia comm<sup>n</sup>.

- \* It ensures reliable & in-sequence data trans<sup>n</sup>, so that data units sent over the net arrive completely.

\* SCTP Segment :



- \* SCTP conn<sup>n</sup> oriented also includes
  - Association establishment
  - Data transfer.
  - Association Termination.



⇒ (P)-to-(P) Commu =

\* Takeplace through client & srvr.

\* t(P) is an appln (L) entity that uses the service of the t. (L).

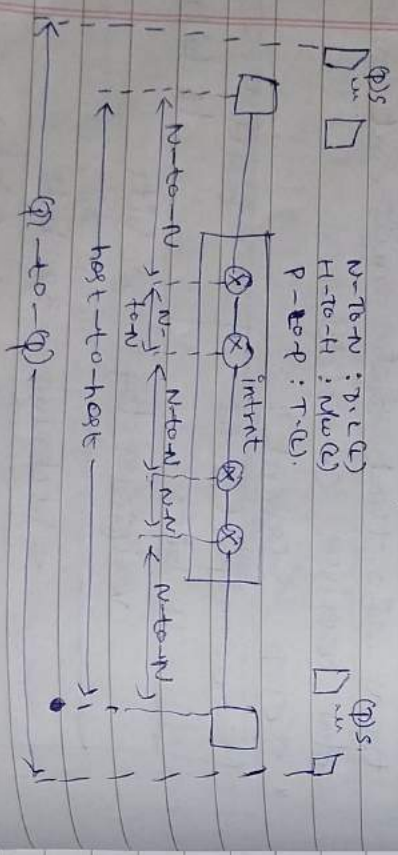
\* Node-to-node delivery →

D.L (L) is responsible for delivery of messages b/w 2 neighbouring nodes over a link

\* Host-to-host delivery →

N/w (L) is responsible for delivery of ~~the~~ datagrams b/w 2 hosts.

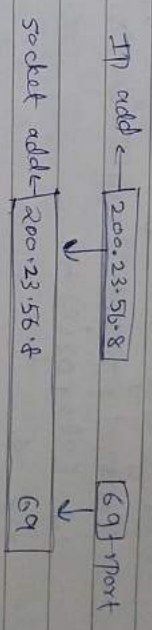
\* (P)-to-(P) delivery →



\* To deliver (P)s, we need General & distinctive  
→ port no. In TCP/IP (P), the port no. are int b/w 0 and 65,535 (16 bits)  
client prgm delivers itself with a port

now, chosen randomly by the t. (L).

\* Combin of IP add & port no. → socket add



I Encapsulation & Decapsulation:

\* Encapsu" hpy at Sndr site, when a (P) has a msg to Snd, it passes the msg to t. (L) along with pair of Socket add.

\* Decap8u" hpy at receiver site, when the msg arrives at destn t. (L), the header is stripped & t. (L) delivers the msg to the (P) running at the appln (L).

II Multiplexing & Demux:

\* Mux → stitching data from multiple appln (P)s at the Sndr, & sending them as a whole to the intended receiver.

\* Demux → delivering received segments at the receiver side to the correct app (L) (P)s.



→ Congestion Control !  
Refers to the techniques used to control / prevent congestion.

2 categories -



1) open loop c.c =

are applied to prevent congestion before it happens. The c.c is handled by either by the source / the destination.

Policies -

\* Retrans policy - If the sender feels that a sent packet is lost / corrupted, the packet needs to be retransmitted.

\* Window policy - The type of window (go-back-n, selective-repeat) at the sender's side may also affect the congestion.

\* Dropping P -

A good D.P adopted by the routers is that the routers may "prune congestion" i.e. at some time partially discard the congested sensitive packets.

\* Admission P

\* Ack P

2) closed loop c.c =

\* Backpressure -

Technique in which a congested node stops receiving packets from upstream node.

\* Choke packet technique -

Technique is applicable to both virtual as well as datagram subnets.

\* Implicit signaling -

Here, there is no communication b/w the congested nodes & the source.

\* Explicit signaling -

Here, if a node experiences congestion it can explicitly send a packet to the destination to inform about congestion.

Forward signaling } explicit & occurs.

→ Quality of service = (QoS)

\* Set of technologies that work on a network to guarantee its ability to dependably run high-priority applications.

\* Accomplish this by providing differentiated handling & capacity allocation to specific flows in network traffic.



\* Traffic types that qos measures -

- 1) Bandwidth → speed of a link.
  - 2) Delay → Time taken for a packet to go from its source to its destination.
  - 3) Loss → Amount of data lost as a result of packet loss.
  - 4) Jitters → Irregular speed of packets on a network as a result of congestion.
- \* Techniques to improve qos -
- 1) Scheduling → packets from different flows arrive at a router/switch for processing.
  - 2) Traffic shaping → mechanism to control the amount & rate of traffic sent to the network.
  - 3) Resource reservation → flow of data needs resources such as buffers, etc.
  - 4) Admission control → mechanism used by a switch/router to accept/reject a flow specification.

services → Addressing, Network virtual terminal  
→ file transfer, Access & management  
→ mail services & directory services.  
→ Authentication

⇒ Application (L) =

\* Exists at layer 7, ~~at~~ the top of the 7-layer OSI communication model.

\* Engineers can effectively communicate with other applications on different computer systems & networks.

\* It's protocols -

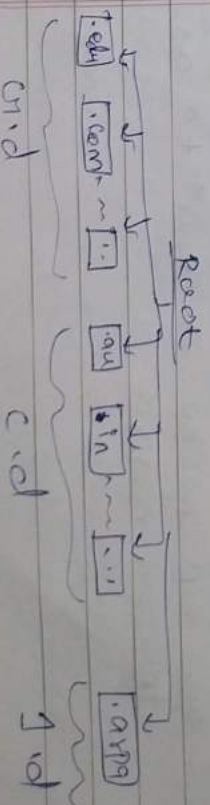
a) DNS :

\* Every host is identified by the IP address, but remembering now is very difficult for people.

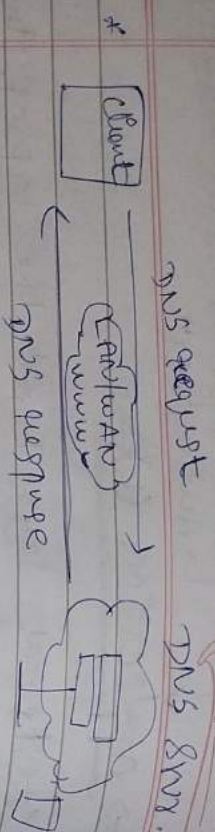
\* So DNS is used to convert the domain name of the websites to their numerical IP address.

\* Types of domains -

- 1) Generic id → .com, .edu, .mil (military)
- 2) Country id → in (india), .us, .uk.
- 3) Inverse id → .arpa.

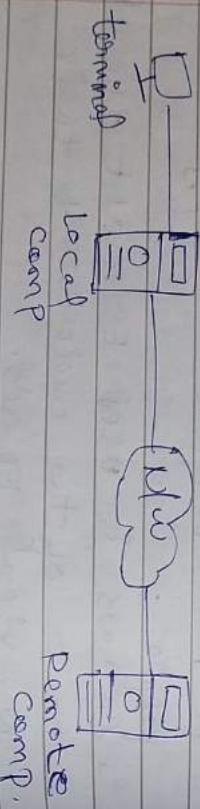






### b) Remote login:-

- \* process in which user can login into remote side.
- \* with the help of remote login a user is able to understand vlt of transposing & vlt of processing from the remote comp to the local comp.



- \* It is implemented using Telnet.
- \* It is a protocol that allows you to connect to remote comps over a tcp/ip net.
- \* using telnet client s/w on yr comp you can make a conn to a target s/w.

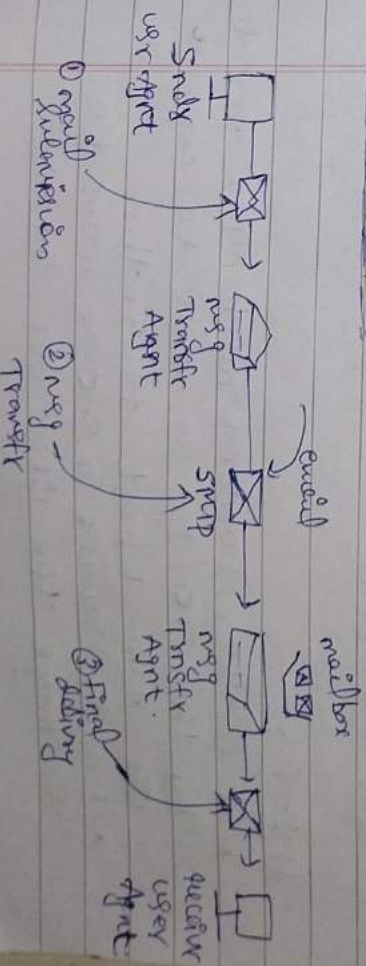
### c) Email = (e).

- \* A method of exchanging msgs over the internet.
- \* An email add is a unique identifier for each user in the format of @domain.com.
- \* An email client is a s/w pgrm used to send, receive & manage emails like email.
- \* An email s/w is a comp system responsible for storing & forwarding emails to their intended recipients.
- \* Services -> Com position, Reporting, Displaying, Transposing, Disposition.

### \* Components -> user Agent

- > msg Transf Agent
- > msg Access Agent.

### \* Architecture -





## \* Structure of (e) -

2 parts -

### ① Header :

generally contains the sender's add as well as the receiver's add & the subject of the msg

### ② Body :

Body of the msg containing the actual info that is meant for the receiver

\* (e) add consists of 2 parts -

### ① Local part :

used to define the name of the file file which is commonly a user mailbox.

It is the place where all the mails received for the user is stored for retrieval.

### ② Domain Name :

Second part of the add is Domain

Both local part & Domain are separated with the help of @.

## d) FTP :

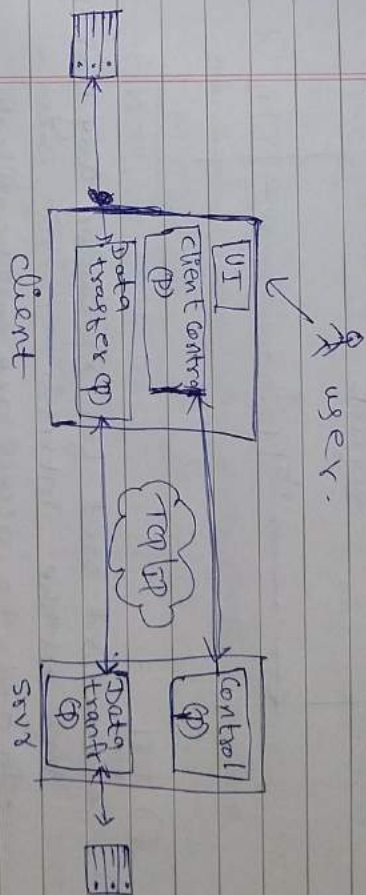
\* It is the standard mechanism provided by the TCP/IP in order to copy a file from 1 host to another.

\* It is the easiest & secure ways to exchange the files over internet.

\* It is different from the other client/server

app. as this protocol establishes 2 conn b/w the hosts - data conn & control conn.

\* It makes the use of 2 protocols - port 21 for the control conn & port 20 for data conn.



## e) www :

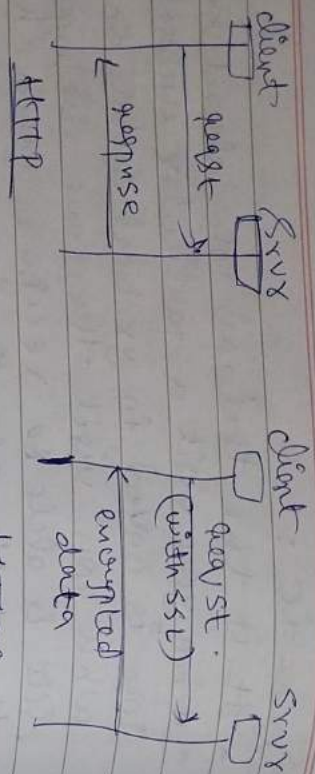
\* Basically a collection of info that is linked together from points all over the world.

\* provide flexibility, portability & user-friendly features.

\* consist of worldwide collection of electronic doc. (web pages)

\* The service that is provided is distributed over many different loc → 800s / websites.

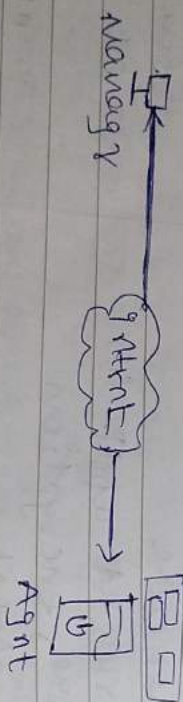




HTTP.

1) SNMP (Simple network management (P)):

- framework used for managing the device on the internet by using TCP/IP (P)
- It is an application (P) that was defined by the internet engineering task force.
- used to monitor the network, detect the faults in the network, also configure remote devices.



3 components in the architecture

1) SNMP manager → used to monitor & manage devices that are connected with the network.

2) SNMP agent → a software program that

is packaged within the network element.

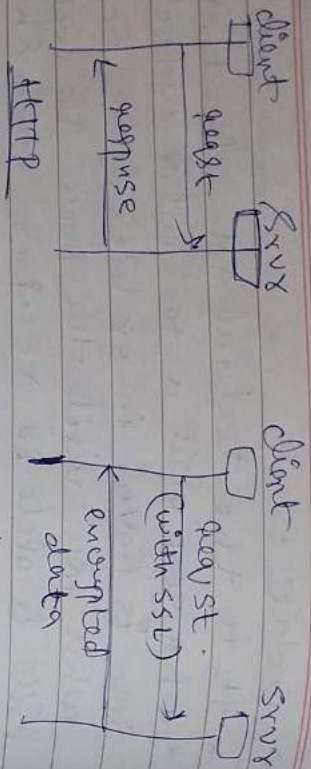
3) Management Information Base (MIB)

used to create a set of objects that are defined for each entity that is similar to the DB.

→ Network Management:

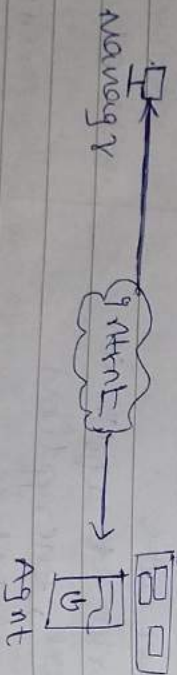
- process of monitoring, testing, configuring & troubleshooting network components to meet a set of requirements defined by an organization.
- ISO defines 5 areas of network management:
  - 1) configuration . M
  - 2) fault . M
  - 3) performance . M
  - 4) security . M
  - 5) Accounting . M.





### 1) SNMP (Simple network management (P)):

- Framework used for managing the device on the internet by using TCP/IP (P)
- It is an application (P) that was designed by the internet engineering task force.
- Used to monitor the network, detect the faults in the network, also configure remote devices.



### 2) Components in the architecture

- 1) SNMP manager → used to monitor & manage devices that are connected with the network.
- 2) SNMP agent → a software program that

is packaged within the network element.

### 3) Management Information Base (MIB)

used to create a set of objects that are defined for each entity that is similar to the DB.

### → Network Management:

- Process of monitoring, testing, configuring & troubleshooting network components to meet a set of requirements defined by an organization.
- ISO defines 5 areas of network management:
  - 1) configuration . M
  - 2) Fault . M
  - 3) performance . M
  - 4) security . M
  - 5) Accounting . M.