They are of two types.

1) Client - Server
2) Peer - to - Peer.
3) Hybrid.

1) Single Server to which the others connect and request services. Client Requests and Server Services.

2) Purely for file sharing.

3) Eg :- Chat.

10/07/2014

Effective Throughput = $\frac{\text{Transfer Size}}{\text{Transfer Time}}$

Eg :- For a data of 1MB with 1Gbps bandwidth with RDT = 100ms

Throughput = $\frac{8 \times 10^6}{10^9} = 8$ ms.

Effective Throughput = 100 + 8 = 108ms.

$\Rightarrow \frac{8 \times 10^6 \times 8}{108} = \frac{8}{108} \times 10^9 = 74$ Mbps.

Client process initiates communication. Server process is one that waits for communication.

HTTP always runs at port no. 80. SMTP at 25.

Application Performance Requirements.

→ Reliability. (messages should be properly received).

→ Bandwidth.

→ Timing (talks about Delay).

Multimedia applications are loss-tolerant. Applications that don't prefer losses in data are said to be loss-intolerant.

Based on bandwidth Apps can be bandwidth-sensitive or bandwidth insensitive.

Delay sensitive and Delay insensitive apps are based on delay.

Transport layer provides TCP and UDP.

TCP is a connection oriented protocol. UDP is a connection less protocol. TCP checks for the fidelity of the packets used in transmission. UDP doesnot provide for reliability. TCP provides congestion, flow control.

UDP does not provide any of the features of TCP. The only adv. of UDP is that it is simple to implement and faster than TCP.

HTTP :-

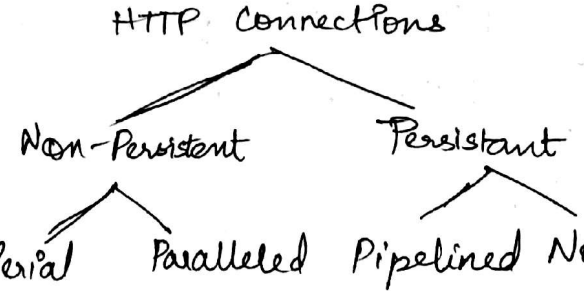
Stands for Hypertext Transfer Protocol. An URL has host name and path name. The system implements a client server relationship. A connection is requested which is responded with an HTTP message.

Control Packet - Packet without data.

On receiving the segment TCP sends an ACK and also in forthcoming sends. A request for a particular data can be done using ACK.

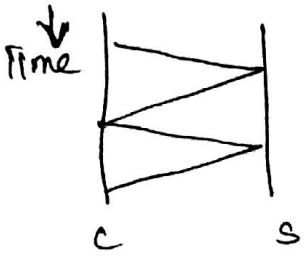
HTTP is called a stateless protocol because it does not store the state of the previous request.

12/07/2017

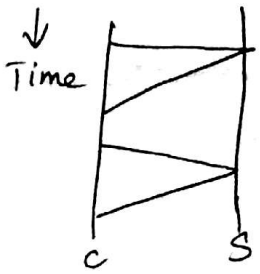


Non-persistant :- Atmost single object is sent over TCP and then the connection is closed.

Non Persistant



Persistant



Persistant :- the connection is kept open and it get closed only when inactivity is detected.

Serial :- It consists of connections that allow only one object to be requested at a time.

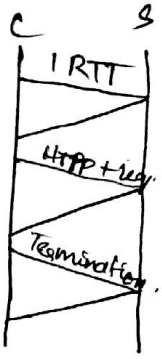
Parallel :- Requests the remaining objects in parallel. Multiple connections (till 10) are allowed by the browser.

Pipelined :- Parallel Requests are allowed. No need to wait for previous requests to be handled.

Non-Pipelined :- Only after reception of one request can another request can be made.

The 3-way handshake contains an initiation from client, ACK by server and confirmation & Request by client.

RTT - Round Trip time.

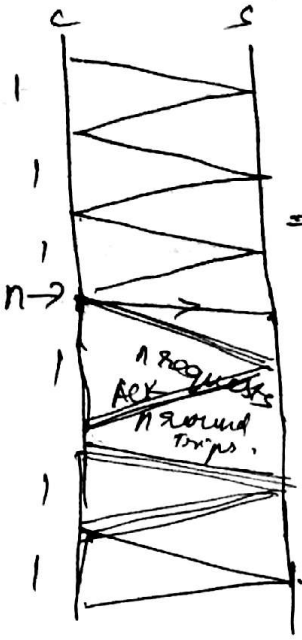


i) one RTT spent for initiation and ACK
ii) one RTT for HTTP request and first few bytes to return.
iii) one RTT for termination.

∴ Total RT for one object = 3RTT + file transmission time.

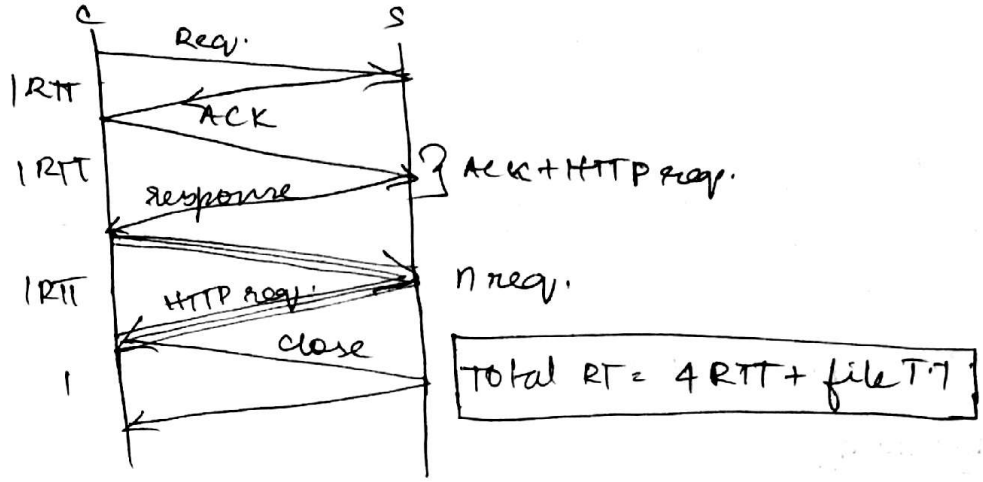
For n objects, RT = 3nRTT + 3RTT + Transmission time.
{non persistent? serial}.

Non-Persistent Parallel:-

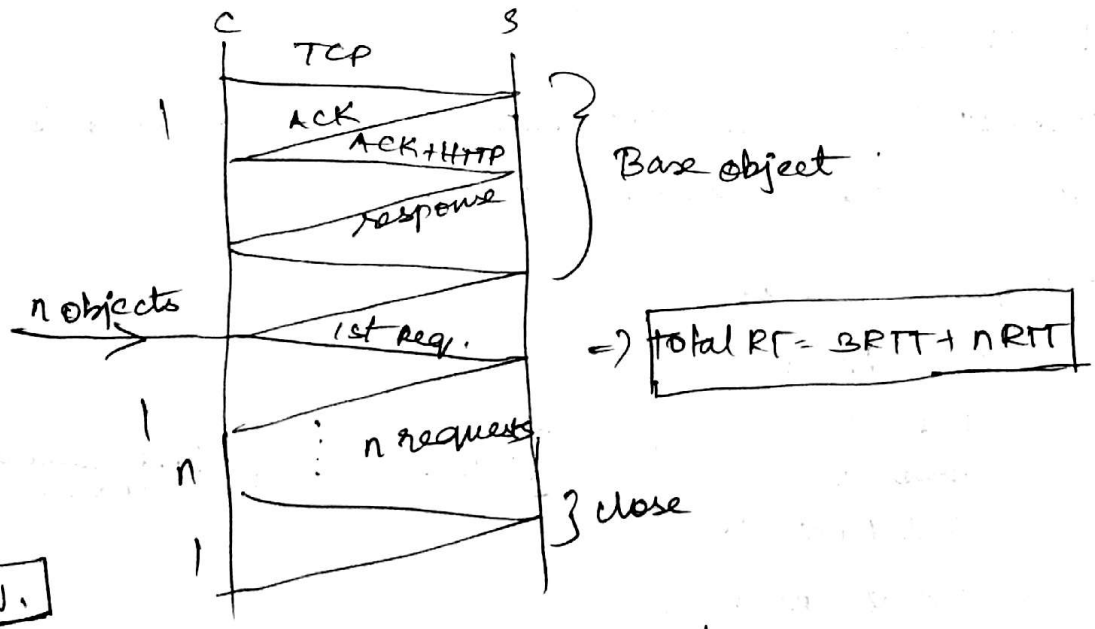


⇒ total RT for one object = 6RTT + file T.T.

Persistent Pipeline :-



Persistent Non-Pipelined



H.W.

a) Assume a webpage with 3 objects.
a) NP no ||some. b) NP with parallel c) P with ||
d) P with No||.

Request Form | HTTP messages < Request / Response.

Request Message

method	—	URL	—	version	CR	LF

CR - Carriage Return.
LF - Line feed.
CR moves it to start
LF moves to next line.

Header lines

Header field name :		value	CR	LF

⋮

Entity body is an empty portion filled with request message.

The requests are GET, POST, HEAD, PUT, DELETE. (HTTP methods)

→ HEAD is purely used for debugging purposes.
→ PUT is used for putting an object onto the server.
→ DELETE deletes the object on the server.
→ POST posts information on the server. (Eg:- DB activities)

13/07/2017.

Eg get method is used for retrieval the form input is added to the URL field.

The version field can be HTTP V1.0 or HTTP V1.1. The entity body is used for carrying the requested object.

HTTP Response Message

Connection :-
Date :-
Server :-
Last Modified :-
Content-length :-
Content-type :-

2nd Assignment :- HTTP response codes.

Response Status Codes

200 — OK.
301 — Removed Permanently.
400 — Bad Request
404 — Not found.
505 — Not Supported.

Cookies

1) Set-Cookie header line to Response message.
2) Another header line for Cookie
3) Cookie file kept in user's host
4) back end database in the company.

Lab on 7th Packet using structure

message format for HTTP calls

User-Server Interaction.

Web-Site needs to identify users
→ restrict user access
→ wants to serve content as a function of the user identity.
→ Solution - HTTP uses Cookies [RFC-6265].

Adv.

→ Shopping Cart
→ One-Click Shopping
→ Shopping recommended products

Disadv.

→ Invasion of Privacy.

Web Caches (proxy Servers)

goal :- satisfy client request without involving origin server.

→ user set browser :- Web accesses via cache.
→ browser sends all HTTP requests to the cache.
→ Cache acts as both the client and the server.
→ Typically cache is installed by the ISP.

Web caching
→ Reduces response time
→ Reduces traffic
→ Bandwidth can be optimized.

Cache has hits and misses

Hit Rate = No. of Successful Requests / Total No. of Requests

Till 0.8 the value of traffic intensity value is negligable. Adding the hit and miss of the cache delays we can arrive at a reduced delay time.

Conditional GET :-

→ The aim is to not send the object if it has an updated cache version.

→ Specify date of cached copy in HTTP request.
(If modified since <date>)

→ Response contains no object if cached copy is up-to-date. (304 not modified).

2nd Session

→ Client sends the HTTP request to proxy. If the object is there then the conditional get message is answered with the cache object itself.

FILE TRANSFER PROTOCOL (FTP)

→ Follows client-server.

→ Stateful protocol.

→ Uses TCP as transport layer with port no. being 21.

→ Client is going to put request with server being a remote host

→ FTP uses 2 TCP connections, one is control and the other is data.

→ Control carries FTP commands and replies. The data conn. is used to carry files.

→ Control is established from client to server. Data is from Server to Client.

→ FTP client establishes TCP with server in control conn. server tries to authenticate by asking Username. Client gives the U.N. to the server upon which a password is requested. The server authenticates the client and an FTP session is established between them. The client requests for connection via control. Upon reception the data conn. is initiated and data transfer happens.

Control Conn. → Persistant

Data Conn. → Non-Persistent.

FTP follows Out of Band communication ie. two different connections for control & data transfer whereas HTTP is Inband.

commands

Eg:- USER name ; PASS password. etc.

a) Consider two hosts A and B connected by a single link of rate R bps. Suppose the 2 hosts are separated by a distance of m meters and suppose the propagation speed is S m/s. Host A should send a packet of size L bits to B. Express the propagation delay and the transmission delay. Ignoring processing & queueing delays, obtain an end-to-end delay. Suppose S = 2.5 × 10^8 and L = 120 bits and R = 56 kbps. find 'm' if P. delay is equal to transmission delay.

Ans:- P. Delay = $\frac{m}{s}$ seconds.

Transmission Delay = $\frac{L}{B.W.} = \frac{L}{R}$.

End-to-end delay = $t_{processing} + t_{queueing} + t_{transmission} + t_{propagation}$.

End-to-end delay $= \frac{m}{s} + \frac{L}{R}$.

$\Rightarrow$

$$\frac{M}{8} = \frac{L}{R}.$$

$\Rightarrow$ $\frac{M}{2.5\times10^8} = \frac{120}{56\times10^3}$

$$\frac{M}{} = \frac{120 \times 2.5\times10^8}{56\times10^3} = \frac{300}{56}\times10^5 = 536\text{kms}.$$

5) Calculate the total time needed to transfer 1.5 MB file in the following cases.

i) An RTT 80 ms.

ii) Packet size of 1 KB, Initial to 2RTT of handshaking.

iii) the Bandwidth is 10 Mbps. and data packets can be sent continuously.

b) 10Mbps and after finishing the departure of a data packet we have to wait for the next transmission.

Ans:- Size $= 1.5\text{MB} = 8\times1.5\times10^6 = 12\times10^6$ bits.

BW = 10 Mbps.

delay = $t_{prop} + t_{trans}$.

delay $= \frac{80}{2} + \left[\frac{12\times10^6}{10^7} + 2\text{RTT}\right]$.

delay $= 40 + \left[\frac{12\times10^7}{1.2} + .16\right]$.

delay = 1.2 + .04 + .16.

delay = 1.48.

ii) No. of packets $= \frac{1.5\times10^6}{10^3}$

$= 1.5\times10^3$

$= 1500$

$= 1499$ packets

$= 1499\times80\times10^{-3}$

≈ 120 seconds.

Total delay = RTT delay + total delay

$= 120 + 1.4 = 121.48$.

i) Consider a P-to-P link of length 50 kms. At what B.W. would propagation delay (at a speed of 2×10^8 m/s) equal transmission delay. Transmission is for 100 Byte packets?

Ans:- P. Delay $= \frac{50 \times 10^3}{2 \times 10^8} = 25 \times 10^{-5}$

$t_{propagation} = 250\,\mu s$.

$\frac{h}{BW} = 250\,\mu s$.

$B.W. = \frac{Size}{time}$.

$\Rightarrow 250\,\mu \times BW = 100 \times 8$.

$BW = \frac{100 \times 8}{250 \times 10^{-6}} = \frac{800}{250} \times 10^6$

$= 3.2$ Mbps.

Peterson Ex 3.85. (a,b)

ii) for a packet of 512 Bytes.

$BW = \frac{512 \times 8}{250 \times 10^{-6}}$

$=$

17/07/2017

Electronic Mail :-

→ It is an asynchronous communication mode and reads whenever convenient.

Three Components

1. Useragents
2. Mail Server.
3. SMTP.

User agents:- ~~Devices that~~ Applications that are used to get the mail ready.

Mail Server:- Stores outgoing and incoming messages.

It has two components:

1) message Queue.
2) Mailbox.

SMTP

→ Simple Mail Transfer Protocol.
→ Uses TCP for mail transfer and connects to port 25.
→ There are 3 phases to its working.

- → handshaking
- → Transfer of Messages.
- → Closure.

→ The interaction is generally via commands and responses. Message are in 7-bit ASCII requires binary multimedia data to be encoded to ASCII. reconversion happens at the other end.

SMTP uses a persistent connections.

HTTP – pull protocol

→ Initiated by Client (One who wants).

→ Not heavily imposed.

→ Each object has its own response

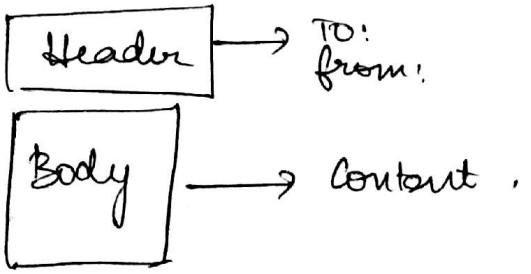
SMTP – Push Control.

→ One who has information initiates the link.

→ SMTP has a compulsion on encryption

→ place all of the message's in one response message.

Format :-



Mail Access Protocols :-

POP → Post Office Protocol, port 110, three phase :- authorisation, transaction and update.

IMAP → Internet Mail Access Protocol more features, including manipulation of stored messages on Server.

HTTP/Web based email → use Web browsers as user agent, gmail, hotmail, Yahoo mail etc.

POP3 ← Download & delete / Download and keep.

Download & delete → Once downloaded Deleted from Server.

Download and keep. → Server Copy Remains Intact.

Update phase

→ happens after Quit command.

→ POP3 is stateless across sessions.

1st Chapter 11, 12, 13

IMAP.

→ Messages are always at the Server.

→ Changes made are reflected on the servers.

19/1/2017

Domain Name System (DNS):-

A host can be identified either by hostname or i/P address.

[Hostname eg:-] google, annauniv.

IP address provides a better search mode that enables the user to identify a website precisely.

Multipurpose Internal Mail Extension (MIME)

↓

Sending User Request

↓

data → encode → 7-bit ASCII

decode ←

The DNS converts the hostname into its corresponding IP address. DNS has two components:-

1) Distributed Database: → Comprises of a hierachial arrangement of DNS Servers.

2) Application Layer Protocol.

↳ Interface where the comm. happens

DNS Services

→ Translation.

→ Host Aliasing :- When a hostname is highly mnemonic we use host aliasing. Canonical Names are (lengthy). difficult to remember.

→ Mail Server Aliasing :- Related to Mail Servers.

→ Load Distribution :- Resolves problems among replicated servers.

HostName to IP address Translation :-

→ User machine runs Client side.
→ Hostname is extracted and sent to DNS App.
→ Client sends a query to the DNS Server.
→ Client receives the IP address for the hostname.
→ Once the IP is received, comm. is initiated via TCP or HTTP.

Why no Centralized DNS ? :-

→ Single point of failure.
→ Traffic Volume.
→ Distant Centralized database.
→ Maintainance.

Classes of DNS Servers :-

→ Root DNS
→ Top level Domain (TLD) Servers
→ Authoritative DNS.

→ Client queries Root initially. Host name is detected and redirected to the appropriate domain server is

pigged.

TLD Servers:-

→ Responsible for .com, .in and other domains.

Authoritative:-

→ Maintained by Companies or by Service Provider.

Local DNS Name Server

→ Does not belong to hierarchy.

→ Query raised by host can be sent to a local DNS server. Acts as proxy, forwards query into hierarchy.

DNS Name Resolution → Iterated Query.
→ Requested Query.

Iterated Query

Query is replied with IP of next server to be contacted.

Recursive Query

The name server is contacted and is incharge of replying the IP.

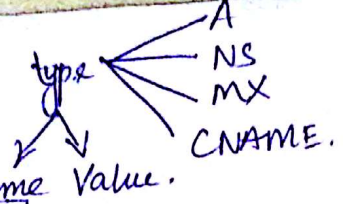
Caching can reduce delay and traffic ~~and~~ on servers.
Caching has to be done with update of information.
TTL → defines the expiry period for an entry in the cache.

20/07/2017

DNS is a distributed Database that can store resource records.

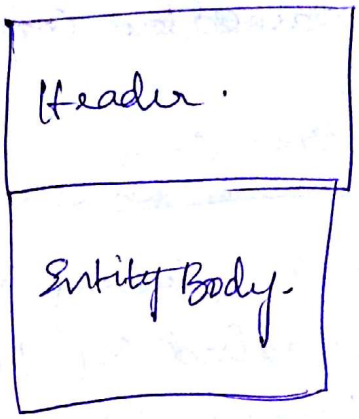
Record Format :- (name, Value, type, ttl).

→ Name is the hostname
→ Value is the IP address.



Name	Value	Type
Hostname	IP address	A
Domain Name	host name of the authoritative Server	NS
Alias Name of mail Server	Canonical Name	MX.
Alias Name for web Server	Canonical Name	CNAME.

TLD must have NS, A.
A is there in authoritative (Transfers IP address).



DNS messages are either query, or reply. flags are query(0) or reply (1). A query can be solved in a recursive fashion (In request it is a request desired, If in reply its called Request Available). Based on the message carried, the header bits will change.

IANA manages DNS Servers under the supervision of ICANN. The root zone file contains info regarding all TLDs. Domains are decided by ICANN.

TRANSPORT LAYER.

A transport layer is an end-to-end protocol. It is shaped by 2 forces :-

→ Application process.

→ Network layer bcoz it has some limitations.

node-to-node delivery is facilitated by the DataLink layer. Network facilitates host-to-host.

Transport layer focusses on end-to-end.

Transport Protocols have 2 ends :-

→ Sending side.

→ Receiving side.

TCP	UDP
→ Reliable, In order delivery	→ unreliable.
→ Congestion control	→ No frills extension.
→ flow control	→ Connectionless.

Flow control

Fast sender may have to wait incase the receiver isn't competant.

Allocating space in buffer + initializing variables is a part of connection oriented protocol.

End-to-End Protocols

Typical limitations of the network on which transport protocol will operate.

→ Drop messages

→ Reorder messages

→ Deliver duplicate copies of a given message.

→ Limit messages.

Multiplexing at send host:-

→ gathering data from multiple sockets,

Demultiplexing at rcv host:-

→ Delivering received segments to correct socket.

→ This is done using IP addresses and port numbers.

UDP

→ Extends host-to-host delivery service of the underlying network into a process-to-process communication service.

→ Add a level of demultiplexing which allows multiple application processes on each host to share the network.

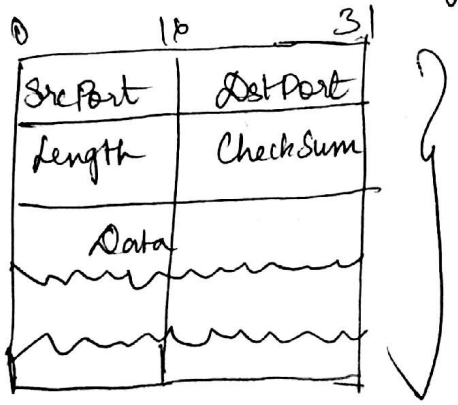
Importance of UDP

→ Reliability.

→ Small Segment header

→ No connection establishment.

→ SNMP Simple network Management Protocol.



UDP CheckSum → It is used to check the correctness of the message.

There is a pseudoheader with 3 fields.

→ Protocol No.

→ Source and destination IP address.

→ Pseudo Header. UDP Length field.

The Checksum is used to detect errors. The Checksum is checked at both sender and receiver stages.

1) How wide is a bit on 10 Gbps?

2) How long is bit in copper wire where speed is 2.3×10^8 m/s.

3) How long does it take to transmit x KBs over a y Mbps link?

4) Suppose a 128PTOP link is set up between earth & Mars. The distance between earth & mars when closest is app. 55 Gm. Data travels over the link at the speed of the light. Calculate DB product.

5) A camera is set up on a Rover takes pictures of the surroundings and sends them back. How quickly can it reach the Earth. Assume each image is 5Mb.

1) $\frac{1}{10^{10}} = 10^{-10}$.

2) $D = S \times t = 2.3 \times 10^8 \times \frac{1}{10^{10}} = 0.023$ m.

3) $\frac{x \times 10^3 \times 8}{y \times 10^6} = 8 \times \frac{x}{y} 10^{-3}$.

4) Propagation delay = $\frac{\text{distance}}{\text{Speed}} = \frac{55 \times 10^9}{3 \times 10^8} = 184$ s.

RTT = 2*prop. delay = $2 \times 184 = 368$ s.

5) $184 \times 128 \times 10^3 = 2.81$ MB.

$\Rightarrow$ Transmission Delay = $\frac{5 \times 10^6}{128 \times 10^3} = 3.28$ s.

Total delay = 338 + 184 = 512 secs.

24/07/2017

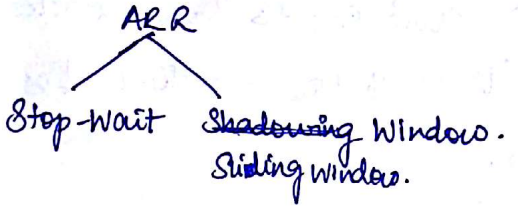
Reliable Transmission.

→ Function of data link and transport layer.
→ Data can be corrupted in transmission.
→ A protocol wants to deliver frames reliably must recover from these discarded frames.
→ Achieved using two ways (Acknowledgments and Timeouts).

Automatic Repeat Request is the process of ensuring Reliable transmission.

We use an ACK packet to ensure that the frame has been successfully delivered.

Piggybacking → Carrying ACK for previously sent data



Stop & Wait

→ Sender's sends a packet and waits for an ACK for successful Reception.
→ If ACK isn't received based on a timeout mechanism the sender resends the previous frame. Else the next frame is sent.

By comparing sequence number, the receiver detect. the duplicate packets

→ The sender has only one outstanding frame on the link at a time. Outstanding frames are frames that can be in transit before receiving the acknowledgement.

Send Window Size (SWS)
Receive Window Size (RWS) } Shading Window Protocol.

The SWS transmits when a frame is sent. The RWS transits when an ACK is received for successful reception.

26/07/2014.

Sliding Window Protocol:-

→ Sender assigns a sequence number for each frame.

→ Sender maintains 3 variables.

→ Sending Window Size (SWS).

→ Last Acknowledgement Received (LAR).

→ Last Frame Sent (LFS).

SWS - upperbound on the no. outstanding frames.

LAR - Sequence number of the ~~last~~ ACK received.

LFS - Sequence number of the last frame sent.

$$LFS - LAR \leq SWS$$

When ACK arrives, sender moves LAR to the right.

On the receiver's side, there are 3 variables that are maintained.

Receiving Window Size (RWS) - Upper bound on the no. of out of order frames the receiver is willing to accept.

Largest Acceptable Frame (LAF) → Sequence number of the largest LAF (Accepted frame).

Last Frame Received (LFR) → Sequence no. of the last frame received.

$$(LAF - LFR) \leq RWS.$$

If the above condition is satisfied for a frame it can be stored else it is discarded.

SeqNumToAck

It denotes the sequence number such that the frames whose sequence numbers are less than the frame in question have been accepted.

Cumulative ACK:-

ACK's are sent in clusters when a sequence of frames are transmitted.

Eg:- If frame 0-3 are received and we can send an ACK only for 3 implicitly meaning that 0,1,2 have been received.

Issues with Sliding Window

→ Timeout occurs, and the amount of data transits decreases.

→ Packet loss results in the pipe being half full.

→ Others ACKs are Negative Acknowledgement, Additional Acknowledgement, Selective Acknowledgement.

NAK:- sent when preceding frames don't arrive and succeeding frames arrive.

Duplicate ACK:- Sends an ACK for a frame that's already been ACKed if the frames don't arrive in the expected order.

Selective ACK:- ACKs only the frames that the receiver receives.

→ Improves performance by early detection of packet loss.

→ Sender can keep the pipe full.

Fixing Window Size

→ Utilize bandwidth fully. (=SWS)

→ RWS can be either 1 or SWS.

When RWS = SWS, there is no meaning.

→ Finite Sequence Number

→ Specified in Header field. Eg 3 bits (0-7).

→ It is necessary to wrap around.

→ Distinguishing between duplicate Sequence Numbers :-

→ No. of possible sequence numbers must be larger than the number of outstanding frames allowed.

a) Stop & Wait :- One frame can be in transit.

b) MaxSeqNum = available Seq. Nos.

⇒ MaxSeqNum (ie.) SWS <= MaxSeqNum − 1.

MaxSeqNum ≥ SWS + 1?

→ Depends on RWS.

→ RWS = 1 will suffice not the same for RWS = SWS.

The best scenario is $SWS \leq \frac{(MaxSeqNum+1)}{2}$

This will help the protocol to alternate between the two halves of the sequence no. space making it a special case of stop and wait.

→ Provides Reliability, Preserves the Order, Frame Control.

↓ Sequence Nos. ↓ Size

27/07/2017

TCP :-

→ TCP offers the following services :-

a) Reliable

b) Connection oriented.

c) Byte-Stream Service.

d) Full duplex protocol.

e)

→ Flow control involves preventing senders from a fast send.

→ At the heart of TCP is the SWP because it runs over the internet.

→ TCP supports logical connections between two different suggestions.

a) explicit connection establishment phase and connection teardown phase.

→ TCP connections are likely to have widely different RTT times.

→ Variations in the RTT are even possible during a single TCP connection and lasts only a few minutes.

→ Packets can get reordered in the internet. Worst case a packet can be delayed in the internet until the IP time to live field expires and the packet gets discarded.

→ TCP assumes that each packet has a maximum lifetime - maximum segment lifetime (MSL), is an engineering choice. The current recommended setting is 120 seconds.

→ TCP mechanism using each side of a connection will learn what resources the other side is able to apply to the connection.

→ TCP needs a mechanism using which the sending side will learn the capacity of the network.

TCP is a byte-oriented protocol

Packets exchanged between TCP peers is called segment. The Src Port and the Dst Port are each of 16 bits long. The sequence number is 32 bits long. Each byte of data has a sequence number. The ACK number is 32 bits long. If ACK is set then it tells the sequence number that the sender is expecting to receive. Header length is a 4 bit long element. The TCP header is an integral number of 32 bits long.

There are 6 flags used with TCP peers. They are SYN, RESET, PUSH, URG and ACK. SYN and FIN are used in Connection Establishment.

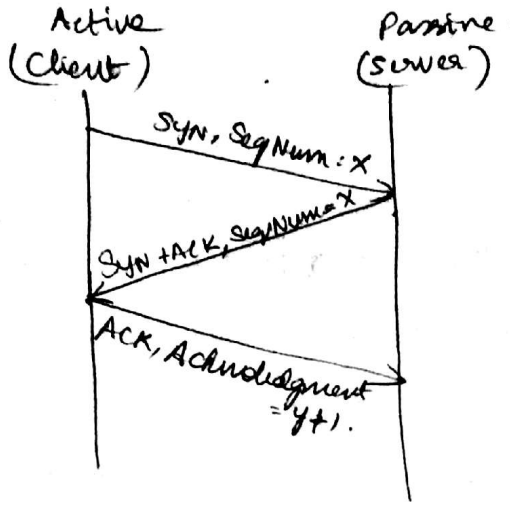
The URG flag points to the end location of the urgent data. The PUSH op signifies the invocation by the sender. RESET flag signifies the point where a receiver receives an unexpected segment.

Advertised Window:-

→ 16 bits long.
→ The number of data octets beginning with one indicated in ACK field which the sender is willing to accept.

Checksum :-

→ 16 bits long
→ One's comp. of one's comp. sum of all 16 bit words in a header, text and pseudo header.
→ Computed over the TCP header, the TCP data, and the pseudoheader.
→ Client does active open.
→ Server does passive open.
→ TCP performs a Connection-Establishment & Connection termination phase.



Timeline for a 3-way handshake.

31/07/2017

SYN - Synchronisation, used to establish a TCP conn.

Sequence Number identifies the frame uniquely. (TCP header field).

Simultaneous Open

→ It is possible, although improbable for two apps. to both perform an active open.

Closed - is a state in a dead Client component.

SYN-SENT - When a SYN is transferred.

SYN-RECEIVED - When a SYN is received.

ESTABLISHED - enabled when communication is ready.

FIN - To connection Termination.

FIN-WAIT - State where a FIN call is sent. It is closed only when this side sends an ACK.

TCP-HALF CLOSE - When FIN-WAIT is set.

TIME WAIT - Wait for Double max. Segment Life (MSL) Time.

CLOSE-WAIT - Another intermediate state.

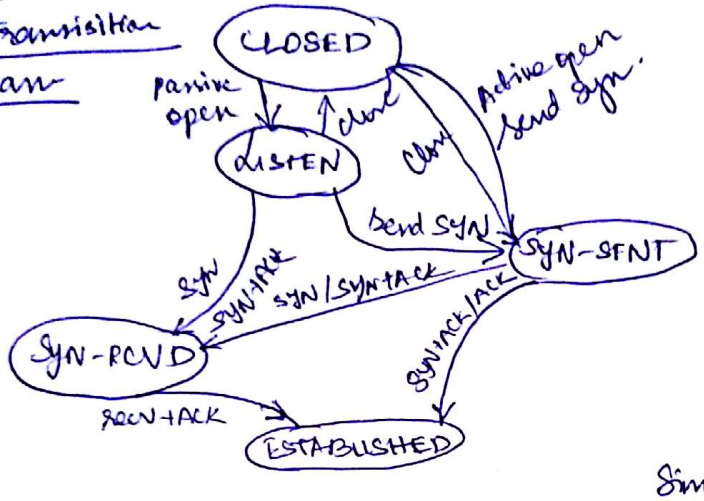
LAST-ACK - Waits for an ACK.

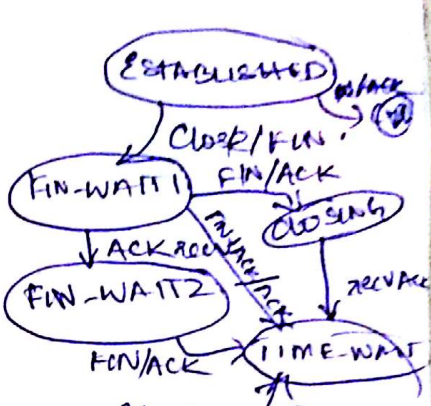
Simultaneous Close

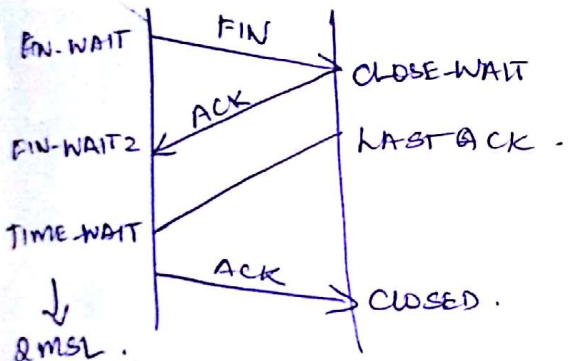
→ Both sides can close at the same time. Both sides will need to send the fin call to activate it. It goes to CLOSING state.

3/05/2017.

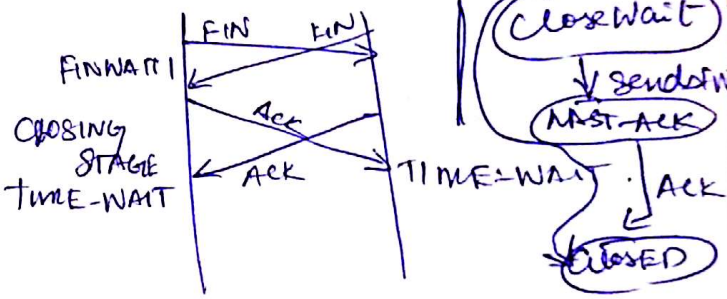
TCP Transition Diagram







Simultaneous Close



The side that initiates FIN first is called the active close.

MSL for TCP is 120 seconds. (Dictates the amount of time a packet can live in the socket).

TCP Sliding window algo Advantages

→ In order delivery of data.
→ Reliable data transmission.
→ Flow Control.
→ Send data based on capacity of the receiver.
→ Receiver can declare its capacity in Advertised Window field (16 bit). Sequence Number field (32).

The application on the ~~receiver~~ sender side can buffer the incoming data. The buffer can be overwritten. On the receiver side, the unread received data has to be preserved.