Spring 23 – set B  Assignment 01  2211000 2

(1) a) Correct order: 2, 6, 1, 4, 5, 3, 7

b) The 3 peers should have all the pieces of the file for the 4th to be able to collect them. That is, the total 256 MB (8 chunks) must be there with at least one of the 3 peers.

Optimistic unchoking → it is a Bittorrent system. This happens when a peer randomly chooses to upload to a new neighbour to find potentially better clients.

c) Given, access link = 123 mbps, LAN Link = 1023 mbps.
40 obj/s, → 4mb

for LAN Link → $\frac{40 \times 4M}{1023 \times 10^6} = 0.156 \times 100\%$

$= 15.6\%$

for Access Link → $\frac{40 \times 4M}{123 \times 10^6}$

$= 130.1\%$

Since the values are very large, Adding a proxy would help to reduce access link load.

d) If we delete the Date header, conditional requests might fail and there can not be freshness checks done since servers need to send a date header with each response and caches also need it to check the time of that object

P.T.O

Proxy servers reuse TCP connections. Instead of a handshake for each single request, it keeps the connection open to be reused.

2. a) For slow internet, POP3 is better and faster. because, it downloads the messages from server So it can be read offline so it does not require constant ~~data~~ usage. Whereas Imap need constant synchronisation making it slower.

b) Persistent cookies. When we first visit a website it sets the cookies in our browsers to remember our data by when we revisit that website

c) i) RTT = 2 x 44 = 88 ms

ii) After IP, req time (send) = 89 ms), non-persistent, objects - 28, 24 MB each, upload speed = x.

RTT1 = 89 + 89 = 178 ms

In non persistent, 2RTT

So, total RTT = 88 + 2 (28 x 178)

= 10,056 ms

iii) Given, total FTT = 19988 ms

FTT = 19988 - 10056 = 9932 ms = 9.932 s

Total data = 28 x 24 x 8 = 5376 Mb

$X = \frac{5376 \times 10^6}{9.932} = 541.3$ Mbps = 541 Mbps

(3) a) Half close : This allows one side to stop sending data while still receiving from the other side. This is required when a process has no more data to send but is still left to receive more.

Full close : Communication is terminated from both sides completely and all resources are released. This is needed when there's no more data is to be sent or received.

b) When the TCP receive three duplicate Acknowledgments for same segment, it realizes some packet must have lost, so instead of waiting till Retransmission time is over, it immediately sends the missing segment through Fast Retransmit process.

c) Server ISN = 9429 , each segment = 889 bytes
Client ISN = 8484 , size = 235 bytes

i)

| | | syn | ack |
|---|---|---|---|
| segment 1 | | 9429 | 8484 + 235 = 8719 |
| n | 2 | 9429 + 889 = 10318 | 8719 + 235 = 8954 |
| n | 3 | 10318 + 889 = 11207 | 8954 + 235 = 9189 |
| n | 4 | 11207 + 889 + 1 = 12097 | 9189 + 235 + 1 = 9425 |

Syn = 12097 , Ack = 9425

(3) a)

ii) 11th segment:

Seq no. = 8484 + (235×10) + 1

seq = 10635

9th segment lost, so last acknowledgement would be sent until the 8th segment

ack no = 9429 + (889×8) + 1

ack = 16542

iii) Given, client RWND = 10005 bytes

segment size = 889 bytes

processed = 5 segments

so free = 5×889 = 4445 bytes

when receiving 13th segment, accepted only upto 8th as 9th segment is lost.

so, total bytes received = 8×889 = 7112 bytes

remaining = 7112 - 4445 = 2667 bytes

Now, RWND of client when receiving 13th segment = 10005 - 2667

= 7338