

SL-1 of trigger wave

no bias voltage but  $i_{ds}$  (transistor circuit with bias)  
zigzag waveform

on-off bias waveform right edge

if we go threshold voltage to off state

if at below threshold voltage of min bias  
threshold bias no waveform

to get new threshold voltage  
req sweep rate

est. ID-24141264

(a) estimate

self oscillating bias

Section-23 (CSE421)

oscillations

if sensitivity of input

transistor to input

threshold voltage of

the bias waveform if the dc voltage

(from waveform

switching) take up two new positions without

Ans. to the Que no - 1

. 2Y (P)

(a) Connect order from Application Layer (7) to physical Layer (1)

to connect the layers between host to host

1. Application Layer (L7) - Step-6

2. Presentation layer (L6) - Step-5

3. Session (L5) - Step-3 in handshake, making connection

4. Transport (L4) - Step-1

5. Network (L3) - Step-2

6. Data Link (L2) - Step-7 frame reaches receiving host

7. Physical (L1) - Step-9 frame reaches destination host

Host has a host receiving host, receiving host

host has a host receiving host, receiving host



I - on and off of 2NA

(b) Yes.

In BitTorrent, peers exchange different pieces of the file. The protocol is designed so that there is at least one seeder → (a peer with complete file).

collectively the swarm has all pieces distributed among peers, downloading is possible.

(c) No.

The proxy server won't send a conditional Get request since it is dependent on the freshness of cache. proxy server checks if the cached copy is valid and fresh. In question, the proxy has a copy but doesn't mention client validation headers.

(d) This possible because of;

1. Video Compression & Efficient Encoding

2. Content delivery networks

CDN's

3. Buffering

�िराइग्रफ रेसर्व.

\* access link bandwidth is lower than 100 mbps but higher than actual average bitrate of the compressed stream. That's why it can stream 8k without buffering.

एडोर्ड ब्रिफर्म.

जो जो डिफरेंट लेवल्स का स्ट्रीमिंग होता है।

सिफारिश सिद्ध.

नोट्स मिट्टो स्ट्रीमिंग.

Ans. to the Que no -2

(a) Websites track you even without sign up by ;

1. cookies

2. IP address

3. Browser fingerprinting

4. Local storage

Tracking helps the HTTP protocol

1. managing sessions

2. provide personalized contents or ads

3. Enable analytics

4. Caching optimization

(b) Recursive DNS lookup is better overall for the typical internet ecosystem.

### Justification:

1. End-user experience - Client gets faster cached response
2. Widely deployed; Google, cloudflare use recursive resolution for client resolution
3. Reduces complexity - don't need to handle multi-step iterative process
4. Caching efficiency - Recursive resolves cache results for many users.

2] (c)

, 25/11/ (iii)

① PC to Local DNS = 23 ms

with RTT total = 46 ms

$$\text{Total recursive resolution} = (3 \times 46) = 138 \text{ ms}$$

$$\text{DNS RTT} = 500 \text{ s} - 138 \text{ ms} = 462 \text{ ms}$$

$\therefore$  Total RTT = local DNS + recursive resolution

$$= 46 + 138$$

$$= 184 \text{ ms.}$$

Ans.

$$\text{DNS RTT} = 2 \times 0.9 = 1.8 \text{ ms}$$

② Here,

$$\text{DNS time} = \frac{0.88 \text{ s}}{184 \text{ ms}} = \text{DNS RTT}$$

1 RTT between client and server = 78 ms

TCP handshake to open persistent connection;

$$\frac{1}{x} = \frac{0.88 \text{ s}}{0.001 \text{ s}}$$

$$\text{for 30 objects} = 30 \times 78 = 2340 \text{ ms.}$$

$$0.001 \text{ s} \approx 0.001 \text{ s}$$

$$\therefore \text{Total} = \text{DNS} + \text{TCP} + 30 \text{ objects}$$

$$= 184 + 78 + 2340$$

$$= 2602 \text{ ms.}$$

(iii) Here,

(ii) 12  
①

2008 - 2009 Total of 59

Total time = RTT time + Data transfer time

$$\Rightarrow 5344 = 2602 + \text{Data transfer time}$$

(DP × E) = (Data transfer time) + (RTT)

$$\Rightarrow \text{Data transfer time} = 5344 - 2602 = 2742 \text{ ms.}$$

Now,  $RTT = 2602 + 2742 = 5344 \text{ ms.}$ 

$$\text{each object} = 12 \text{ MB} = 96 \text{ Mb}$$

$$\text{Total data} = 30 \times 96 = 2880 \text{ Mb}$$

Now,

$$\text{Data transfer time} = \frac{2880}{x} \text{ ms.}$$

Now  $x = \text{round trip time} = \text{RTT}$ Now  $x = \frac{2880}{1050} \text{ ms.}$ 

$$\Rightarrow \frac{2742}{1000} = \frac{2880}{x}$$

$$\Rightarrow x = 1050 \cdot 69 \approx 1.05 \text{ Gbps}$$

$$(\text{Round trip time} + \text{Object size}) / \text{Throughput} = \text{RTT} + \frac{\text{Object size}}{\text{Throughput}} = 1.05 \text{ ms.}$$

$$OP + SF + PS =$$

$$RTT + Object size / Throughput =$$

Ans. to the Ques no - 3  
Ques. If two different clients having same IP (d)

(a) Yes, It is possible for a web server to receive multiple HTTP request that contain the same port number. However, each request does not arrive over the same TCP connection. TCP connection is uniquely identified by;

Source IP, Source Port, Destination IP and destination port.

The server differentiates them based on their unique combination of source IP and source port number.

(b) UDP doesn't provide reliability because it lacks fields such as Sequence Number, Acknowledgment number and Window Size in its header. However, reliability is achieved at the application layer, where applications implement their own mechanism like ACKs, sequence numbering or error correction to ensure data is received correctly when UDP is used.

(c) ① Server ISN = 203

(ii)

1st byte =  $203 + 1 = 204$ , ~~2508~~ = ~~first~~

Sequence number of 1st byte of nth server data;

$$\text{Seq}_n = 204 + (n-1) \times 889$$

For 4th;

$$\text{Seq}_4 = 204 + (3 \times 889) = 204 + 2667$$

~~Therefore, answer~~  $= 2871$  ~~Ans.~~

Now,

Client ISN = 8924

~~using n (e-1)~~

1st " byte =  $8924 + 1 = 8925$

~~(88 \times 0) + NOC~~

4 HTTP request for each 234 bytes =  $4 \times 234 = 936$   
~~1008 + NOC~~

$$\therefore \text{Ack} = 8925 + 936$$

~~2058~~

$$= 9861$$

~~Ans.~~

(ii)

$$EOS = N \times 2^k - 1$$

$$\text{Client Seq}_k = 8925 + (k-1) \times 2^3 Y$$

After receiving ACK for segment 10, the next segment to be transmitted is segment 11.

$$\text{Client Seq}_{11} = 8925 + (11-1) \times 2^3 Y$$

$$(88 \times (1-1)) + 102 = 102$$

$$= 8925 + 2808$$

$$= 11733$$

Ans.

After ACK

$$EOS + POS = (88 \times 9) + 102 = 1002$$

for ACK,

$$ACK = EOS + 1$$

10th server data was lost, only received segments

(1-9) in order;

$$PSCH = N \times 2^k - 1$$

$$PSCH = 1 + PSCH + (9 \times 88)$$

$$ACK = 204 + (9 \times 88)$$

$$ACK = 204 + 800$$

$$= 8205$$

Ans.

$$ACK + PSCH = N \times 2^k - 1$$

$$1080 =$$

Ans.

(iii) Total bytes received =  $12 \times 889$  [1-9 and 11-13]  
= 10668 bytes Total = 2

processed 5 segments bytes =  $5 \times 889$   
= 4445 bytes

Unprocessed bytes =  $(10668 - 4445)$  = 6223 bytes

Advertised RWN.D = receiver buffer size - unprocessed bytes  
=  $(10000 - 6223)$   
= 3777 bytes

An.