CS3640 Midterm Exam

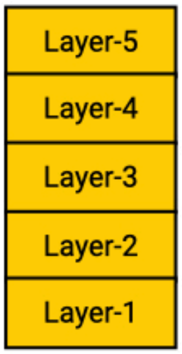
Name:

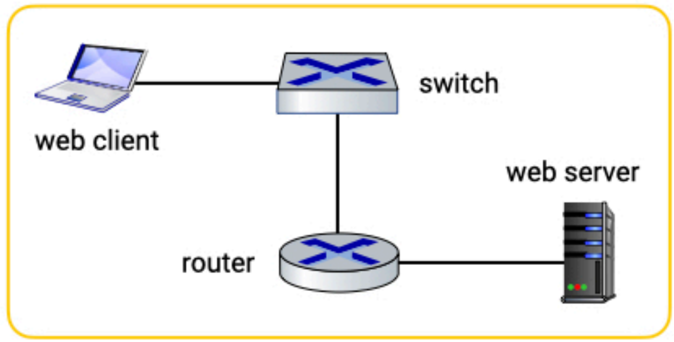
HawkID/university ID:

- This is a closed book, closed notes, closed electronics, but open minds exam
- You have four mandatory questions (Q1 — Q4), all of which carry 25 points
- There is an extra credit question (Q5). Points earned in Q5 can make up for points lost in other questions, but cannot take your score beyond 100
- Keep your answers brief and to the point. While the space provided here should be sufficient to write your answers, we will provide extra sheets if needed.
- You are welcome to clarify any doubts or concerns with the instructor, but cannot engage in discussions with your fellow students
- We expect you to exhibit highest levels of academic integrity and honesty

Q1. Internet Protocols **25 points**

In *Overview* lectures, we discussed how the Internet's protocols are organized in well-defined layers, and how different network elements implement all or parts of those layers.





(a) Name all five layers of the Internet Protocol stack, and explain (in 1-2 sentences) the core functionality of each layer. (15 points)

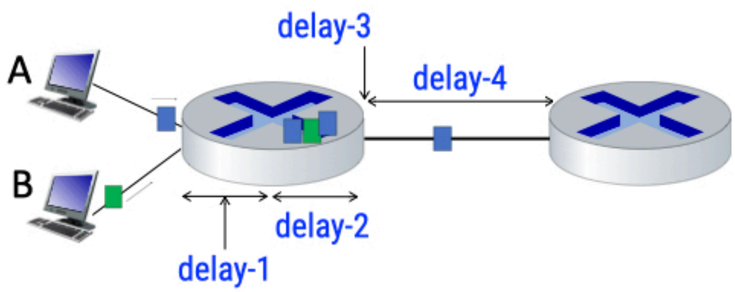
- Layer-5, the application layer, comprises of protocols designed to exchange application level data between the application end-points (i.e., clients, servers). E.g., HTTP, SMTP
- Layer-4, the transport layer, is responsible for end-to-end data transfer between two communicating processes. E.g., TCP, UDP.
- Layer-3, the network layer, is responsible for routing data packets from the source network to the destination network. E.g., IP, IPv6
- Layer-2, the data link layer, handles forwarding of data between adjacent (or directly connected) nodes in the network. E.g., 802.11 WiFi
- Layer-1, the physical layer, is responsible for transmitting bits on the wired/wireless medium. E.g., 10BASE-T

(b) In the given network topology, identify the layers of protocol stack that would exist at each of the network elements. (10 points)

Web client will have layers 1 through 5; switch, being a layer-2 device, will have only layers 1 and 2; routers have to implement layers 1 through 3; and finally, web server needs to have layers 1 through 5.

Q2. Characterizing the Network Delays **25 points**

Here is a network set up with four annotated network delays.



(a) Name all four delays, and explain why they occur. (16 points)

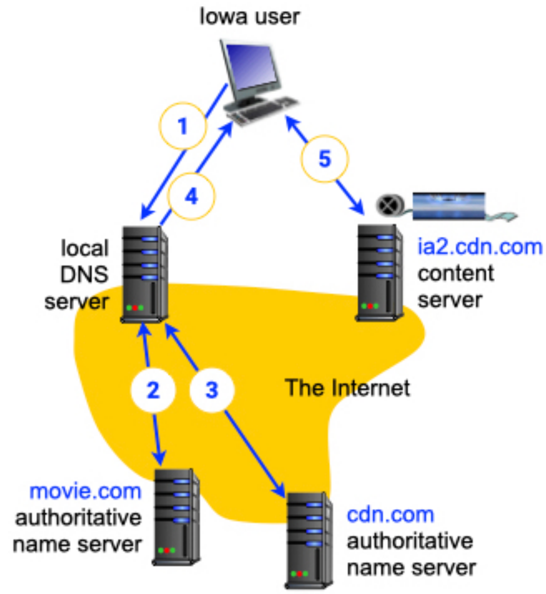
- Delay-1, the processing delay, occurs because the router has to examine the packet's headers and determine how to route/forward it.
- Delay-2, the queuing delay, occurs when there is more than one packet to be sent on a given link at a given time.
- Delay-3, the transmission delay, is the time taken to push all of the packet's bits on to the link. For a packet of L bits sent on a link of capacity R bits/sec, the transmission delay will be L/R sec.
- Delay-4, the propagation delay, is the time taken by a given bit to propagate from one router to the next. For a link of length D meters and a propagation speed of S meters/sec, the propagation delay will be D/S sec.

(b) Network administrators design new techniques, buy new hardware, or implement new policies to decrease network delays. Pick any three delays of your choice, and for each of those, propose an approach that could help minimizing it. (9 points)

- Delay-1 could be reduced by using a faster processor or an ASIC
- Delay-2 could be reduced by adding extra egress capacity, or new links
- Delay-3 could be reduced by increasing the link capacity (for e.g., upgrading a telephone access link to fiber optics line)
- Delay-4 could be reduced by increasing the propagation speed (for e.g., connecting two routers via a fiber optics line as opposed to a satellite link)

Q3. Content Delivery Networks **25 points**

Using the given CDN topology as reference, answer the following questions:



(a) Explain steps 1—5 in full technical detail i.e, describe how CDNs operate to deliver the Iowa user the requested content from IA server. (20 points)

(b) CDNs in the real-world: Mention any one difference between the CDNs operated by Netflix and YouTube. (5 points)

The five steps of CDN operation are

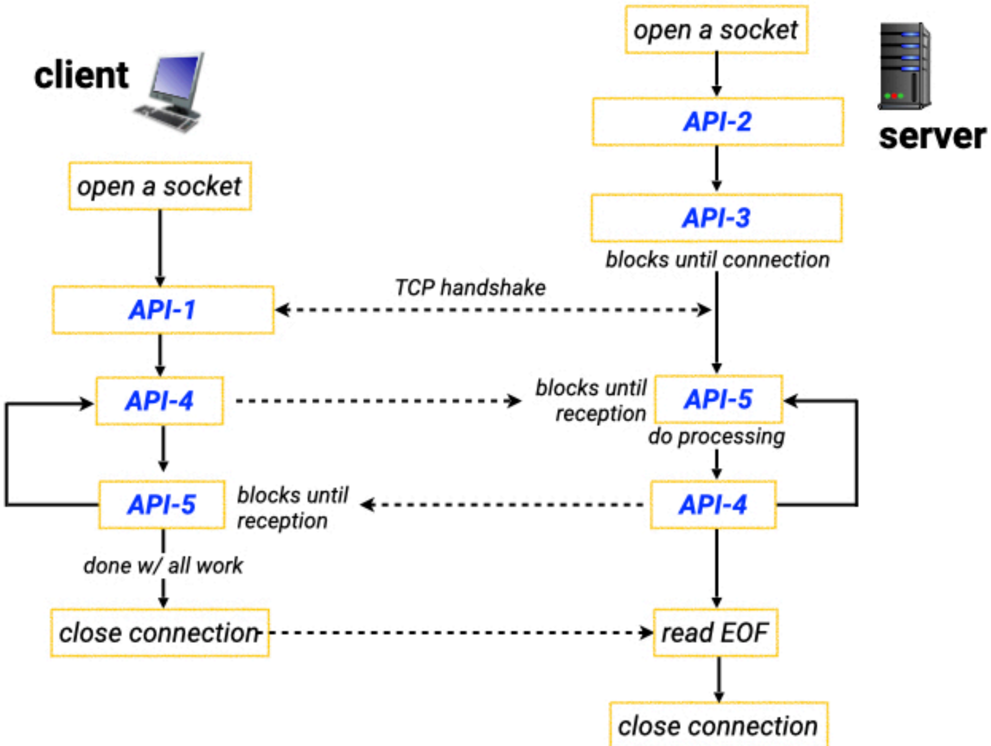
1. **DNS lookup**: Iowa user visits movie.com, and user's host sends a DNS query to the local DNS server
2. **CDN handover**: local DNS server relays the query to an authoritative name server for movie.com. Instead of an IP address, it returns the name of its CDN operator
3. **Content server selection**: local DNS server sends another query to cdn.com, to which the CDN responds with the IP address of the chosen/nearby content delivery server
4. **DNS response**: local DNS forwards the IP address of the content serving CDN node, ia2.cdn.com
5. **Content flow**: client establishes a TCP connection with the local content server, and sends HTTP GET request for the video

Either of the following difference would do:

1. Netflix employs **push caching** (refreshing CDN content daily), whereas YouTube employs **pull caching** (refreshing CDN content based on actual use)
2. Netflix uses **DASH** w/ manifest file, whereas YouTube employs **HTTP streaming**

Q4. Socket programming **25 points**

In the TCP client server interaction diagram below, there are five un-named socket APIs. Name and explain the key functionality of each of the five APIs.

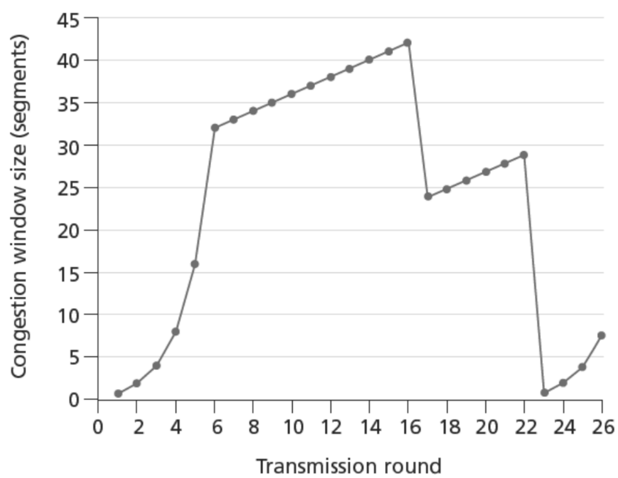


- API-1 is **connect**. This is a client-side API that initiates a TCP connection with the specified server. It returns success if it is able to perform the TCP 3-way handshake, at which point a successful TCP connection will be established.
- API-2 is **bind**. This is a server-side API that associates the given socket with a specified port number. If successful, this API ensures that any packets destined for the specified port number on that server, would be received by the given socket.
- API-3 is **accept**. This is a server-side API that responds to TCP connection requests by a client. Upon success, this API would have completed the 3-way handshake, and created a dedicated socket for communicating with the given client.
- API-4 is **send**. This is both a server-side and client-side API for sending data into an established TCP connection.
- API-5 is **recv**. This is both a server-side and client-side API for receiving data from an established TCP connection. It blocks until any data is received.

Q5. Bonus question **10 points**

Two major components of TCP congestion control algorithm are: *slow start* and *congestion avoidance*. Explain how TCP operates in these two phases in 1-2 sentences. (6 points)

In the figure below, identify the intervals of time when TCP is in the (i) slow start phase and (ii) congestion avoidance phase. Remember that each phase may occur more than once. You can mark these phases on the figure itself. (4 points)



TCP slow start is an operational phase where the congestion window (cwnd) starts slow but grows exponentially. Initially, cwnd is set to 1 MSS, and is doubled every RTT as long as all the transmitted segments are acknowledged. This phase ends when either a timeout occurs or a previously set ssthreshold is met.

TCP congestion avoidance is an operational phase where the congestion window follows additive increase and multiplicative decrease (AIMD). Initially, cwnd is set to 50% of the value when the last congestion occurred. From there on, cwnd increases by 1 MSS every RTT as long as segments are acknowledged, but drops to half its current value when congestion is detected.

In the figure, TCP slow start is operating in the interval 1 through 6, and then again, 23 through 26. TCP congestion avoidance is employed in the interval 6 through 16, and then 17 through 22.