CS118 Midterm

Vincent Jin

TOTAL POINTS

74 / 100

QUESTION 1

20 pts

1.1 (4 / 4)

- **- 0 Correct**
- - 0.5 Base object also requested by proxy
- - 1 One object is missing

1.2 (4 / 4)

- **- 0 Correct**
- - 0.5 Base object is also requested by proxy
- - 4 Incorrect
- - 1 One object missing

1.3 (12 / 12)

- - 3 1st wrong
- - 3 2nd wrong
- - 3 3rd wrong
- **- 0 (not graded) foo.com/logo.png and bar.com/logo.png not related, even if IP match**
- - 0 Correct

QUESTION 2

20 pts

2.1 (5.5 / 6)

- + 6 Correct
- **+ 1 know connection setup RTT + http RTT**
- **+ 2 calculate connection setup RTT correctly**
- **+ 2 calculate http RTT correctly**
- **+ 1 know 4 connections one-by-one**
- **- 0.5 calculation error**
- + 0 wrong

2.2 (1 / 6)

- + 6 Correct
- **+ 1 know two rtt**
- + 2 calculate parallel transmission correctly (queueing)
- + 2 calculate queuing correctly
- + 1 calculate last one correctly
- - 0.5 calculation error
- + 0 wrong

2.3 (2 / 4)

- + 4 Correct
- **+ 1 setup tcp**
- **+ 1 4 requests**
- + 1 one request calculation
- + 1 total
- + 0 wrong

2.4 (0 / 4)

- + 4 Correct
- + 1 connection
- + 1 4 back-to-back response
- + 2 queueing delay
- **+ 0 wrong**
- - 1 end-to-end delay error

QUESTION 3

20 pts

3.1 (4 / 4)

- **+ 4 Correct**
- + 1 query to root
- + 1 query to com
- + 1 query to google.com
- + 1 query to caching resolve
- + 0 wrong
- + 1 partial

3.2 (4 / 4)

- **+ 4 Correct**
- + 1 query to com
- + 1 query to amazon.com
- + 1 query to resolver
- + 1 caching
- + 1 partial

3.3 (2 / 3)

+ 3 Correct
+ 1 root
+ 1 .com
+ 1 google.com
+ 1 no cache
+ 1 partial
- 1 additional queries

3.4 (3 / 3)

+ 3 Correct
+ 1 amazon.com
+ 2 caching
- 1 additional query
+ 0 wrong

3.5 (2 / 6)

+ 1 a) is right
+ 1 b) is right
+ 1 c) is right
+ 1 d) is wrong
+ 1 e) is right
+ 1 f) is wrong

QUESTION 4

20 pts

4.1 (4.5 / 6)

- 0 Correct
- 0.5 SYN in 2nd
- 0.5 no SYN in 3rd packet
- 0.5 no SYN in 3-9th packets
- 0.5 no SYN in 3-9th packets
- 0.5 no RST flags anywhere
- 0.5 No FIN in last packet
- 0.5 no FIN in 1st-6th packets
- 0.5 no FIN in 1st-6th packets
- 1 FIN 8th, no FIN 9th
- 0.5 FIN in 7th and 8th
- 0.5 no ACK in 7th packet
- 0.5 ACK in 3rd packet
- 0.5 ACK in 4 or in 4&5th packet
- 0.5 ACK in 6th packet
- 0.5 ACK in 8th packet
- 0.5 ACK in 9th packet
- 0.5 One or two unspecified flags
- 2 4 unspecified/incorrect flags
- 6 Most flags not filled
- 2 4 unspecified/incorrect flags

4.2 (6.5 / 8)

- 0 Correct
- 1 No. 3 incorrect
- 1 No. 4 incorrect
- 1 No. 5 incorrect
- 1 No. 6 incorrect
- 0.5 No. 7 incorrect (no ACK)
- 1 No. 8 incorrect
- 1 No. 9 incorrect
- 8 Incorrect
- 0.1 Error in calculations
- 0.5 Some mixing of ack and seq positions
- 0.5 SeqNo offset error

4.3 (0 / 3)

- 0 Correct
- 3 Incorrect
- 1.5 Incomplete/unclear reasoning
+ 1 2 correct reasons

4.4 (1.5 / 3)

- 0 Correct
- 1.5 Issue with reasoning
- 0.5 Minor issue
- 3 Incorrect/Missing answer

QUESTION 5

20 pts

5.1 (2 / 4)

- 0 Correct
- 2 Didn't mention no need for a connection (extra RTT)
- 4 Incorrect

5.2 (4 / 4)

- 0 Correct
- 4 Incorrect
- 2 Incorrect explanation
- 1 Incomplete explanation (no delay mentioned)

+ 1 (extra credit: firewall issue)

5.3 (4 / 4)

- 0 **Correct**
- 1 (b) is not secured at all (HTTP), so can be monitored or you can be talking not to amazon
- 1 (c) is encrypted (no monitoring), but you will send data not to amazon, but a potential attacker
- 2 Missing/wrong/incomplete discussion of unsafe cases
- 1 Some issues with discussion
- 2 (a) is safe = DV validated cert
- 1 (a) is pretty safe

5.4 (4 / 4)

- 0 **Correct**
- 2 Incorrect/incomplete reasoning
- 1 Issue with reasoning
- 4 Incorrect/missing answer

5.5 (4 / 4)

+ 2 **Head-of-line blocking**
+ 2 Reduce size of headers / overhead
+ 2 **Request prioritization/request-response interleaving**
+ 2 Proactive push of objects
+ 2 Non-optional pipelining
+ 1 Almost correct item
+ 0 **Correct**
+ 0 Incorrect/Missing

CS118
Spring 2016 Midterm Exam

1 hour 50 minutes

Close book and closed notes; NO use of any device except calculators.

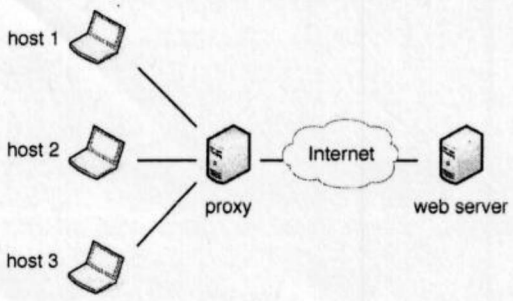
- This exam has 6 pages, including this cover page. Do all your work on these exam sheets.
- Cross out all the scratch work that you do not want to be counted as part of your answer before you submit the exam.
- Show ***all*** your work, including unfinished problems that you wish to be considered for partial credit.
- Be *specific, clear,* ***concise*** in your answers, and *explain* your answers.
- When the answer to a problem is not immediately clear, do not simply dump everything, relevant or irrelevant, on the paper. Irrelevant answers may lead to point-deduction as they show the lack of understanding of the problem.

Your name: Vincent Jin

Student ID: 604464719

Problem 1 (20 points) Three hosts share the same web caching proxy whose cache is empty at the beginning. The browser on host 1 sends the proxy a request for `http://foo.com/info?uid=tom`. This initial object contains three referenced objects, which are then retrieved by the browser on host 1:

```
http://foo.com/logo.png
http://foo.com/profile?uid=tom
http://foo.com/footnote
```



10 seconds later, the browser on host 2 sends a request for `http://foo.com/info?uid=jerry`. This initial object also contains three referenced objects:

```
http://foo.com/logo.png
http://foo.com/profile?uid=jerry
http://foo.com/footnote
```

1.1 (4 points) Please circle one or more HTTP requests that were sent from the caching proxy in the first 10 seconds.

(a) `http://foo.com/info?uid=tom`
(b) `http://foo.com/logo.png`
(c) `http://foo.com/profile?uid=tom`
(d) `http://foo.com/footnote`
(e) `http://foo.com/info?uid=jerry`
(f) `http://foo.com/profile?uid=jerry`

1.2 (4 points) Please circle one or more the HTTP requests were sent from the caching proxy after the first 10 seconds.

(a) `http://foo.com/info?uid=tom`
(b) `http://foo.com/logo.png`
(c) `http://foo.com/profile?uid=tom`
(d) `http://foo.com/footnote`
(e) `http://foo.com/info?uid=jerry`
(f) `http://foo.com/profile?uid=jerry`

1.3 (12 points) Please circle True or False.

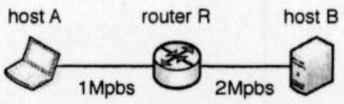
[False] (True or False) If another host, host 3, sends a request for `http://foo.com:8080/logo.png`, the proxy will not send any HTTP request.

[True] (True or False) If host 3 sends a request for `http://foo.com:80/logo.png`, the proxy will not send any HTTP request.

[False] (True or False) Host3 sends another request for `http://bar.com/logo.png`. When the caching proxy sends the DNS query for name `bar.com`, DNS returns the same IP address as the IP address for `foo.com`. This must be an error.

[True] (True or False) If `http://foo.com/logo.png` object is already cached in the proxy, the caching proxy will not send a separate HTTP request for `http://bar.com/logo.png`.

Problem 2 (20 points) A web browser is running on the host A. A web server on the host B. Host A and B are connected to a router R. The bandwidth of Link A-R is 1 Mbps (10^6 bits/sec), while the bandwidth of Link R-B is 2 Mbps (2×10^6 bits/sec). The propagation delay of both links is 10 msec.



A sends 4 HTTP requests to B, each HTTP responses message is sent back in a 1250-byte packet.
Assume that the size of HTTP requests and TCP SYN/SYN-ACK messages are small enough so that their transmission delay can be ignored. Also assume that TCP flow and congestion control window sizes are big enough so that they do not slow down data transmission. There is also no packet loss.

2.1 (6 points) Assuming the browser uses HTTP/1.0 to retrieve the data. The browser only uses a single TCP connection at any given time. Starting from sending the first TCP connection setup (SYN) packet, how long will it take for the browser to receive all the 4 pieces of data?

RTT = 4(10ms) = 40ms

1 Connection setup: 1RTT

1 Http request: 1250 bytes · $\frac{8\text{bits}}{\text{byte}}$ = 10000 bits → $\frac{1}{20}$s + $\frac{1}{10}$s + 1RTT

= 150ms + 1RTT

total: 4(2RTT + 150ms) = 4(80ms + 150ms) = 920ms

2.2 (6 points) To speed up the retrieval, the browser opens 3 TCP connections in parallel. Again starting from sending the first TCP connection setup (SYN) packet, how long will it take for the browser to receive all 4 pieces of data?

connection setup: 1RTT

A sends 3 requests, which start from B: 50ms to reach the router, then 3(100)ms to reach A. + 1RTT.

After 150ms A can send another request - 1RTT + 100ms (from R to A) as the 1st A will be on the router

→ total = 3RTT + 50ms + 3(100ms) + 100ms = 570ms

2.3 (4 points) Assuming the browser uses HTTP/1.1 ***without pipelining*** to retrieve the data over a single TCP connection. How long will it take for the browser to receive all 4 pieces of data in this case?

connection setup: 1RTT (persistent)

150ms | packet still

→ total 1RTT + 4(150ms) = 640ms

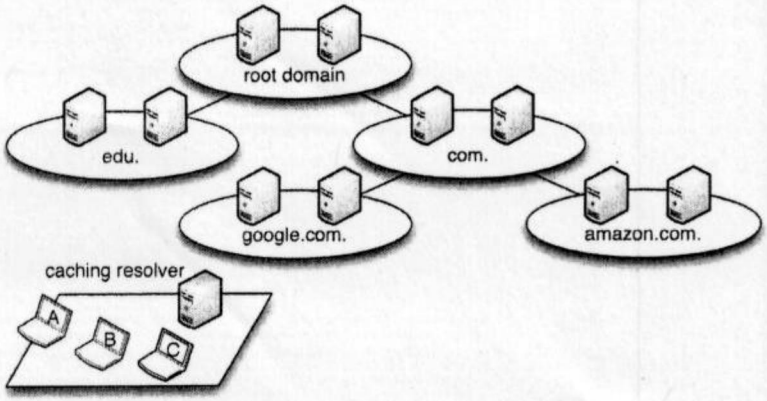
2.4 (4 points) Assuming the browser uses HTTP/1.1 ***with pipelining*** to retrieve the data over a single TCP connection. How long will it take for the browser to receive all 4 pieces of data in this case? Is the delay the same as the one of parallel connections? If so, why we still prefer HTTP/1.1 with pipelining?

connection setup: 1RTT

After sending 4 requests, they are limited by the 1 mpbs connection. That is, it will take $\frac{1}{20}$s to have data on the router, and then subsequent data move on the router, data is being served to the host A.

→ total = 1RTT + 50ms + 4(100ms) = 490ms (faster than parallel connections).

Problem 3 (20 points) Consider the following DNS resolution process:
at time T=0: the caching resolver in the figure has an empty cache. Host-A sends a query to resolve the DNS name *www.google.com* and get the IP address.
T=30 minutes: Host-B sends a query for the IP address of *www.amazon.com* and gets the answer.
T=70 minutes: Host-C sends a query for DNS name *hangout.google.com* and another query for DNS name *video.amazon.com*.



Assuming that it takes 10 msec for packet resolver (10 msec is the round trip delay), and it takes 100 msec for the caching resolver to get a reply from any of the authoritative DNS servers. All authoritative servers support iterative queries only. All the DNS data has a TTL value of 1 hour. There is no packet loss.

3.1 (4 points) How long does it take for Host-A to get the answer back for the IP address of *www.google.com*?

1. root (100 ms)
2. com (100 ms)
3. google.com (100 ms)

→ 300 ms + 10 ms = 310 ms
(10 ms: round trip delay)

3.2 (4 points) How long does it take for Host-B to get the answer back for the IP address of *www.amazon.com*?

1. com (100 ms) - root cached.
2. amazon.com (100 ms)

→ 200 ms + 10 ms = 210 ms

3.3 (3 points) How long does it take for Host-C to get the answer back for the IP address of *hangout.google.com*?

1. com (100 ms) - asks for google.com which is not cached anymore
2. google.com (100 ms) - asks for hangout.google.com

→ 200 ms + 10 ms = 210 ms

3.4 (3 points) How long does it take for Host-C to get the answer back for the IP address of *video.amazon.com*?

1. amazon.com (100 ms) - it knows the amazon.com authoritative server in its cache and asks for video.amazon.com

→ 100 ms + 10 ms = 110 ms

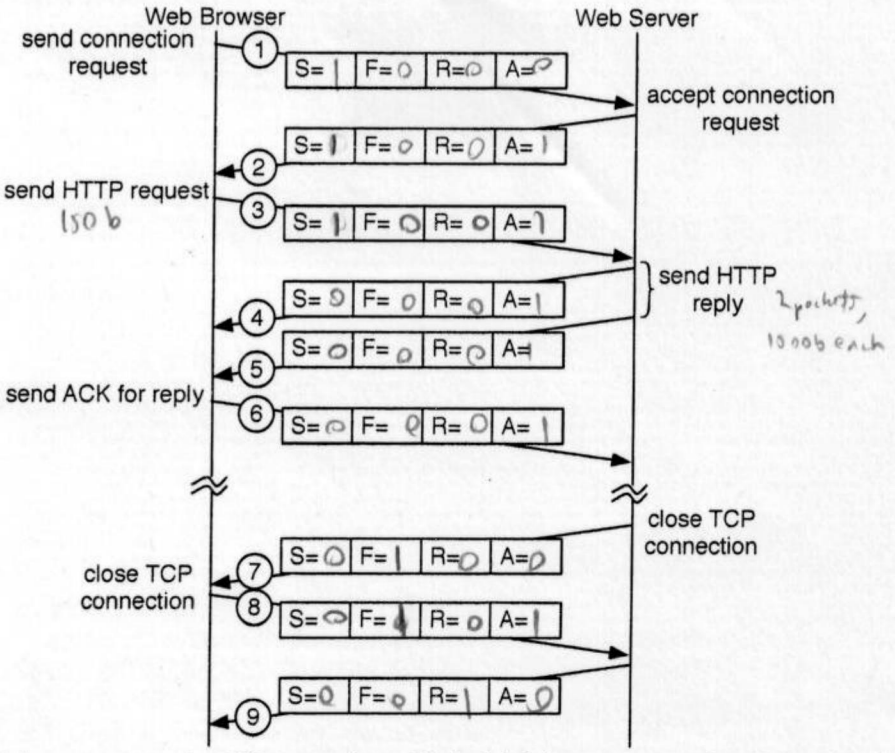
3.5 (6 points) At T=100 minutes, all the authoritative servers of .com go offline. Which domain names below can be resolved by Host-A? Circle those domain names:

in cache: amazon.com server
hangout.google.com

(a) www.google.com (b) hangout.google.com (c) doc.google.com
(d) www.amazon.com (e) video.amazon.com (f) aws.amazon.com

Problem 4 (20 points) The following diagram shows a sequence of TCP packets for a session between a web browser and a web server. The HTTP in use is version 1.0 (non-persistent HTTP).

4.1 (6 points) Fill in all the missing flag values for the SYN, FIN, RST, and ACK flags in the TCP headers (when the flag is set, the value is 1, otherwise is 0).



4.2 (8 points) If the web browser starts its TCP connection with the initial sequence number 308, and web server picks 1110 as its initial sequence number, the HTTP request size is 150 bytes, and the HTTP reply is made of 2 packets with 1500 byte data each. What is the sequence number and acknowledge number on the ***numbered*** packets?

No	Sequence No.	Ack No.
1	308	--
2	1110	309
3	309	1111
4	1111	459

No	Sequence No.	Ack No.
5	2611	459
6	459	4111
7	4111	~~460~~
8	~~1160~~	4112
9	4112	461

4.3 (3 points) Why the sequence number at each end of a TCP connection starts from a random number, instead of zero?

This reduces the possibility that the client/server send seq and ack. numbers for different applications, reducing confusion between the two.

4.4 (3 points) How does the web server know that the browser has received the last packet (packet 9)?

it doesn't, but since the TCP connection has already been closed on the server side, any further TCP packet from the client will be rejected (unless it is to open a new TCP connection). The client has already acknowledged the closing of the TCP connection.

Problem 5 (20 points)

5.1 (4 points) You have learned four application layer protocols: HTTP, FTP, SMTP, and DNS. Only one of them can run over UDP. Which protocol is it? Why it is preferred to run over UDP than TCP? **(in one sentence, otherwise you will not get any credit)**

DNS: we prefer quick responses for DNS and do not care if our packets are dropped. We do not need any reliability that comes with TCP or we can simply retransmit our UDP packets.

5.2 (4 points) If you are asked to develop a real-time online conferencing application, will you choose TCP as the transport layer protocol? Justify your answer.

No - though TCP gives reliable delivery, it is slow and may hinder the real-time aspect of the application. Instead, we use UDP as we can simply transmit new video data in real time even if previous data was lost.

5.3 (4 points) You went to amazon.com website and Chrome shows you the above state in the address bar. In which case you can safely send your Amazon login and password information and why? If in some cases it is not safe, list those and explain why is it not safe and/or what could have gone wrong.

(a)

(b)

(c)

a) We can safely enter our information as our connection is secured and amazon.com's certificate has been verified

b) We cannot safely enter information as our connection runs over HTTP and could be sniffed by a third party.

c) We cannot safely enter information as the certificate is either invalid/expired and a malicious user could be impersonating amazon.com

5.4 (4 points) Some major email service providers recently announced that they have adopted HTTPS-like approach to secure the email communication (each connection between client and SMTP server and between SMTP servers is secured using HTTPS-like connection). Do you think their solution can secure email communication and eliminate all spam? Justify your answer.

No, they cannot eliminate all spam - spammers at a different SMTP server could send secured SMTP messages to a user's server and the spam would be delivered to the user. Additionally, getting all SMTP servers to use this connection will be a hassle and infeasible in practice.

5.5 (4 points) HTTP 1.1 already allows a client to send multiple requests in a single connection. Why we still need multiple streams in HTTP 2.0?

The requests in HTTP 1.1 may not be pipelined and so the response could come sequentially, making it slow. Streams allow these requests to be serviced in parallel, delivering the response faster.