CS118
Spring 2007 Midterm Exam
5/7/07

1 hour 50 minutes
OPEN book and OPEN notes

- This exam has 7 pages, including this cover page. Do all your work on these exam sheets, use the back side if needed.
- If you use the back pages for *scratch space*, cross out your scratch work before you submit the exam.
- Show *all* your work if you wish to be considered for partial credit for unfinished problems.
- Be *specific* and *clear* in your answers, and *explain* all your answers.

Your name: Andrew Ackerman

ID number: 703148627

		Points	extra credit
16	Problem 1	20	
18	Problem 2	20	
17	Problem 3	20	
12	Problem 4	20	5
23	Problem 5	20	5
6	Problem 6		10
92	**Total**	**100**	

5

Problem 1 (20 points) Short and quick questions

(1) How long does it take to transmit *X* KB data over a *Y* Mbps link? Express your answer as a function of *X* and *Y*. (KB = 1000 bytes; Mbps = 10^6 bits-per-second)

$$\frac{X \cdot 10^3 \cdot 8}{Y \cdot 10^6} = \boxed{\frac{8x}{10^3 \cdot y} \text{ sec}}$$

1

(2) Popular web servers, such as cnn.com, may handle web requests from millions of users at the same time. How does a web server identify the connections with individual browsers?

After initial connection comes thru ~~other~~ port 80 ? The cnn.com server will fork a ~~process~~ that would handle the new connection. The process ~~would~~ generate the necessary httpResponse, send it to the client, and then exit.

How the machine/server over the network (src IP, src port, dst IP, dst port)

5

(3) HTTP is called a stateless protocol, that is it handles each request independently from any previous ones. However when one looks up certain websites, such as www.amazon.com, that s/he visited before, the websites can recognize the user instantly. Can you explain how (1)HTTP being a stateless protocol and (2)this recognition can both be true?

cookies

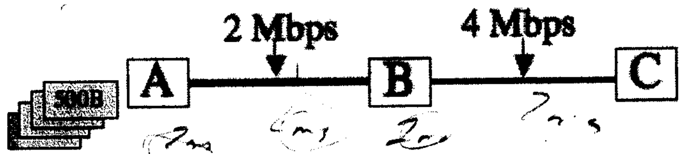
The HTTP is still stateless because the reply the user got back has nothing to do with previous requests. However, unknown to the user is that when s/he looked up a previously looked at site, his/her httpRequest included a cookie (which was set up by server on previous visit). Thus the site was able to recognize him/her because of the cookie (store in the client computer). However, it is stateless because the request was different, thus the different reply (now recognized user). (now had cookie)

2

(4) email delivery uses 2 types of protocols. The first is made of a single protocol, SMTP. The second type includes POP3, IMAP, etc. which are used for accessing email. Why do we need both types? Can one build an email system using only one of the two types?

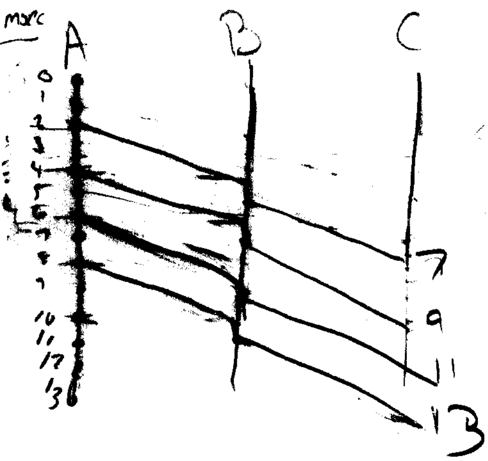
The reason for the two types is that anyone can send mail to anyone, but only the authorized person can recieve mail from a mailbox. That is, sending needs no authorization, but recieving does, thus 2 protocols. A system can be built that only use one of the types, however, in this system a sender must authorize who he is in order to send mail.

Problem 2 (20 points)
Consider sending 4 packets from Node A to Node C via Node B (see the figure below). The packet length is 500 bytes each. The propagation delay of both Link A-B and link B-C is 2 msec (0.002 second). Link A-B's bandwidth is 2 Mbps (2x10^6 bits per second), and link B-C's bandwidth is 4 Mbps.



Assume A starts transmitting the first packet at time t = 0,
a) What is the time gap between the first and second packets when they arrive at C?

Trans delay of Pkt A→B: $\frac{500\times 8}{2\times 10^6} = 2$ msec ✓
trans delay of Pkt B→A: $\frac{500\times 8}{4\times 10^6} = 1$ msec ✓



∴ gap = 9 − 7 = $\boxed{2 \text{ msec}}$

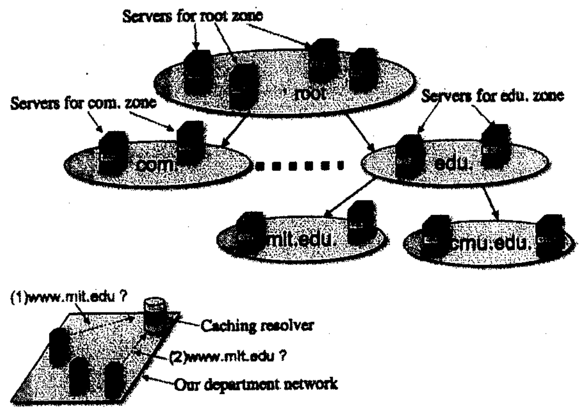
+8

b) How long does it take for C to receive all the 4 packets?

It would take $\boxed{13 \text{ msec}}$ to send all 4 pkts

+10

Problem 3 (20 points)
Consider the following DNS resolution process: assuming that at time T=0, the caching resolver in the figure has an empty cache, and Host-A sends a query to resolve the DNS name of *mit.edu*. Right after Host-A received the answer from the caching resolver, Host-C sends a query for the same name. 10 seconds after Host-C receives its answer, Host-B sends a query for DNS name *cmu.edu* (not shown in the figure). Assuming that it takes 0 (zero) second for all packet exchanges between the local hosts and the caching resolver, and it takes 100 msec for the caching resolver to get a reply for all the DNS queries it sends, and all the DNS data has a TTL of 24 hours.



(1) How long does it take for Host-A to get the answer back for the IP address of mit.edu?

Host to Resolver: 0
Resolver to Root: 100 msec
Resolver to Edu: 100 msec
Resolver to mit.edu: 100 msec
Resolver to Host: 0 sec

∴ $\boxed{\text{Total time is 300 msec}}$

7

(2) How long does it take for Host-C to get the answer back for the IP address of mit.edu?

5

$\boxed{\text{0 seconds}}$, This is because IP is cached from Hosts A's DNS look up

(3) How long does it take for Host-B to get the answer back for the IP address of cmu.edu?

4

Host to Resolver: 0 sec
Resolver to Root: 100 msec
~~Resolver to Edu: 100 msec~~ cached
Resolver to cmu.edu: 100 msec
Resolver to Host: 0 sec

∴ Total time is ~~300~~ msec
200 msec

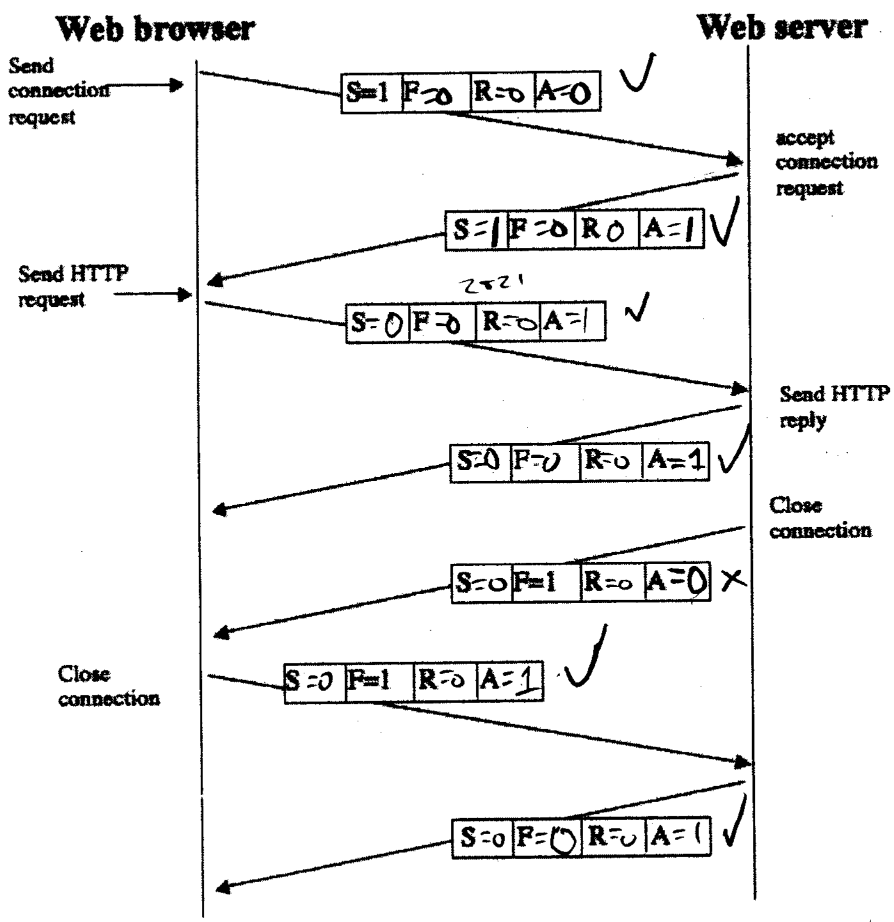
(4) As the results of handing the 3 queries from hosts A, B, and C, what DNS resource records (RRs) are being cached at the local caching resolver? You can describe the RRs in terms of what they are (as the problem did not specify the specific DNS names or IP addresses of the DNS servers).

3

The Caching resolver should now have mit.edu and cmu.edu along with their corresponding ip addresses in its cache. + edu's NS

+12

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).
(1) 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).



6×2 = +12

(2) If the web browser starts its TCP connection with the initial sequence number 2821 and the HTTP request size is 100 bytes, and the HTTP reply is one packet with 500 byte data. What is the sequence number on the *last* packet (with F=1) sent by the browser?

2822 (because only 1 packet of data gets sent)

+0

2922 = 2821 + 100 + 1

(3)(extra credit) What is the value in the sequence number, and acknowledgement number in the TCP header of the last packet sent by the Web server?

seq number: 101 +0 (server only sent 1 packet of data)

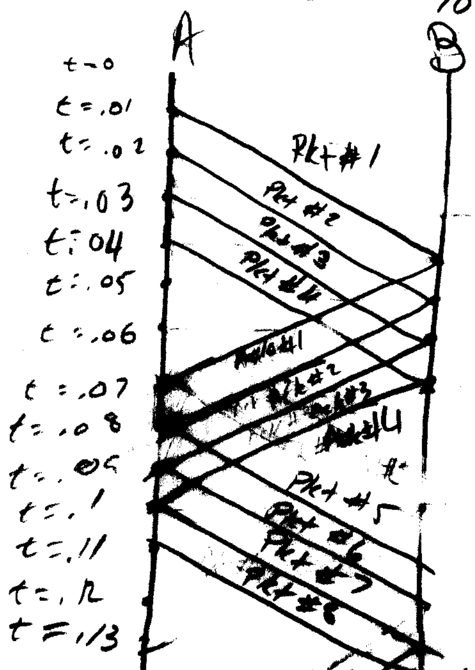
ack number: 2822 (acks the last packet from above question)

(+23)

Problem 5 (20 points) Two hosts A and B are connected by a link with bandwidth of 1 Mbps (10^6 bits-per-second) and propagation delay of 0.03 seconds. Host A has a 200,000-bit file to send to host B. A uses GoBackN reliable transport protocol and divides the file into 10,000-bit packets. The GoBackN protocol uses a fixed window size of 4 packets. You may assume the *transmission time* of ACK packets is negligible and no data or ACK packet ever gets lost.

(1) How long will it take before the 8[th] packet has completely arrived at Host B? (drawing a diagram may help answer this question).

Trans delay = $\frac{10000}{10^6}$ = .01 sec



∴ Packet 8 will completely arrive to Host B after $\boxed{.14 \text{ seconds}}$

(+10)

at t=.07

$\underbrace{1\ 2\ 3\ 4\ 5}_{\text{Win}}$ 6 7

(2) How long will it take before the entire file is received by Host B?

In all there are 20 packets, which is sending 5 windows worth

(.07 × 5) = $\boxed{.35 \text{ seconds}}$ (+10)

(3)(extra credit) Can the file be delivered to host B faster by adjusting the window size? If so what is the minimal window size that would allow the file be received at B with shortest possible time (assume no other setting is changed)?

yes, if all else remains same then the window size needs to be just long enough that the last packet in the window will be transmitting as the first ACK comes back.

(+3)

The minimal window size to accomplish this is $\boxed{7 \text{ packets}}$

$\left(W = \frac{.01 + 2 \cdot .03}{.01} = \frac{7(.01)}{.01} = 7\right)$

Problem 6 (This problem does not carry lots points and is only for those who are interested. You should first finish all the above problems before working on this one.) As we have discussed in the class, timer is a useful component in various protocol designs: because one communication end cannot see what is going on either inside the network or at the other end, when *needed* it sets up an "alarm", and takes certain actions when the alarm goes off.

a) Does IP use any timer(s)? If so, please briefly describe how each timer is used. If not, please explain why it does not need one.

No, This level in the layer would not use the timer

+0

b) Does UDP use any timer(s)? If so, please briefly describe how each timer is used. If not, please explain why it does not need one.

+2

NO, because in UDP after a host sends out a packet it doesn't care about what happens to it. Thus no need to time anything.

c) Does TCP use any timer(s)? If so, please briefly describe how each timer is used. If not, please explain why it does not need one.

Yes, Timer should start on sending of packet to other end. If timeout occurs before ACK for packet gets back then the packet should resend

+2

d) Does DNS use any timers(s)? If so, please briefly describe how each is used. If not, please explain why it does not need one.

Yes, One timer for each DNS request from Resolver to a zone. Timer should start after request is sent out to a DNS server. On timeout the Resolver should resend request

+2

e) Does HTTP use any timer(s)? If so, please briefly describe how each is used. If not, please explain why it does not need one.

NO, the timers would be taken care in the transport layer (i.e TCP)

+0