Problem 1

Protocol Layering:

Consider a scenario where a user on a computer (Host A) wants to send a file to another computer (Host B) over the Internet. The file is first broken down into segments by the transport layer, then into packets by the network layer, and finally into frames by the link layer before being transmitted over the medium.

(a) List the order of headers from innermost to outermost as they are added to the data from the application layer to the link layer on Host A.

(b) At which layer will the routing decision (selecting the next hop) be made for the packet?

(c) When the frames reach Host B and are being processed for the destination application, which layer will first remove its corresponding header?

Write your solution to Problem 1 in this box

Problem 2

Assume that we know the bottleneck link along the path from the server to the client is the first link with rate R_s bits/sec (Fig. 1). Suppose we send a pair of packets back to back from the server to the client, and there is no other traffic on this path. Assume each packet of size L bits, and both links have the same propagation delay d_{prop}.

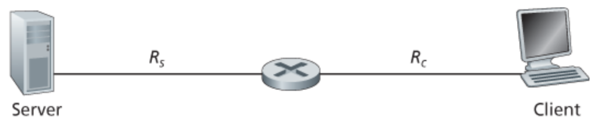

Figure 1: Throughput for a file transfer from server to client

(a) What is the packet inter-arrival time at the destination? That is, how much time elapses from when the last bit of the first packet arrives until the last bit of the second packet arrives?

(b) Now assume that the second link is the bottleneck link (i.e., $R_c < R_s$). Is it possible that the second packet queues at the input queue of the second link? Explain. Suppose that the server sends the second packet T seconds after sending the first packet. How large must T be to ensure no queuing before the second link? Explain.

Write your solution to Problem 2 in this box

Problem 3

Consider a short, 10-meter link, in which the signal propagates at the speed of light. The senders of this link can transmit data at a rate of 200 bits/sec in both directions.

Now suppose that a client at one end of the link browses a webpage using HTTP from a server on the other end. The webpage itself also references 10 other objects from the same server. When this happens, the client and server are the only users of the link.

To make computation easier, we use a simplified model which is *different* from real HTTP and TCP. To download each object (including the webpage), we count the delay of the following four packets:

- One round trip of two packets (one from each side) to establish the connection;
- A request packet from the client side, which represent the HTTP GET;
- A response packet from the server that contains the downloaded object data.

Suppose that each packet in the last step (i.e. the response packet with object) is 100,000 bits long. While packets in other steps are 200 bits long each. We also assume that N parallel connections each get 1/N of the link bandwidth (when parallel connection is used).

Would parallel downloads via parallel instances of non-persistent HTTP have significant gain over non-parallel non-persistent one in this case? What about persistent HTTP vs. the non-persistent case? Justify and explain your answer.

Write your solution to Problem 3 in this box

Problem 4

Consider the following string of ASCII characters that were captured by Wireshark when the browser sent an HTTP GET message (i.e., this is the actual content of an HTTP GET message). The characters `<cr><lf>` are carriage return and line-feed characters (`'\r'` and `'\n'` in C)

```
GET /cs453/index.html HTTP/1.1<cr><lf>Host: gaia.cs.umass
.edu<cr><lf>User-Agent: Mozilla/5.0 (Windows;U; Windows NT 5.1; en
-US; rv:1.7.2) Gecko/20040804 Netscape/7.2 (ax) <cr><lf>Accept:ex
t/xml, application/xml, application/xhtml+xml, text/html;q=0.9
, text/plain;q=0.8,image/png,*/*;q=0.5<cr><lf>Accept-Languag
e: en-us,en;q=0.5<cr><lf>Accept-Encoding: zip,deflate<cr><lf>
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7<cr><lf>Keep-Alive: 3
00<cr><lf>Connection: keep-alive<cr><lf><cr><lf>
```

The server's response begins with:

```
HTTP/1.1 200 OK<cr><lf>Date: Tue, 07 Mar 2008 12:39:45GMT<cr>
<lf>Server: Apache/2.0.52 (Fedora)<cr><lf>Last-Modified: Sat,
10 Dec2005 18:27:46 GMT<cr><lf>ETag: "526c3-f22-a88a4c80"<cr>
<lf>Accept-Ranges: bytes<cr><lf>Content-Length: 3874<cr><lf>
Keep-Alive: timeout=max=100<cr><lf>Connection: Keep-Alive<cr>
<lf>Content-Type: text/html; charset=ISO-8859-1<cr><lf><cr><lf>
<!doctype html public "- //w3c//dtd html 4.0transitional//en">
<lf><html><lf>
```

(a) What is the URL of the document requested by the browser?

(b) What version of HTTP is the browser running?

(c) Does the browser request a non-persistent or a persistent connection? What part in the header tells you the answer?

(d) How many bytes are there in the document returned.

(e) What are the first five bytes of the returned HTML document. How does the browser know where the actual HTML document starts within the response?

Write your solution to Problem 4 in this box

Problem 5

Suppose you have a new computer just set up. `dig` is one of the most useful DNS lookup tool. You can check out the manual of `dig` at `http://linux.die.net/man/1/dig`. A typical invocation of `dig` looks like: `dig @server name type`.

Suppose that on Jan 25, 2025 at 19:00:00, you have issued "`dig www.google.com A`" to get an IPv4 address for `www.google.com` from your caching resolver and got the following result:

```
; <<>> DiG 9.10.6 <<>> google.com A
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32000
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 4

;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         273     IN      A       142.250.217.142

;; AUTHORITY SECTION:
google.com.             55416   IN      NS      ns4.google.com.
google.com.             55416   IN      NS      ns2.google.com.
google.com.             55416   IN      NS      ns1.google.com.
google.com.             55416   IN      NS      ns3.google.com.

;; ADDITIONAL SECTION:
ns1.google.com.         145523  IN      A       216.239.32.10
ns2.google.com.         215985  IN      A       216.239.34.10
ns3.google.com.         215985  IN      A       216.239.36.10
ns4.google.com.         215985  IN      A       216.239.38.10

;; Query time: 5 msec
;; SERVER: 128.97.128.1#53(128.97.128.1)
;; WHEN: Wed Jan 25 19:00:00 2023
;; MSG SIZE  rcvd: 180
```

(a) What is the discovered IPv4 address of `www.google.com` domain?

(b) If you issue the same command 2 minute later, how would "ANSWER SECTION" look like?

(c) When would be the earliest (absolute) time the caching resolver would contact one of the `google.com` name servers again?

(d) If the client keeps issuing `dig www.google.com A` every second, when would be the earliest (absolute) time the caching resolver would contact one of the `.com` name servers?

Problem 5 continued on next page. . .

Write your solution to Problem 5 in this box

Problem 6

Suppose your computer is connected to a WiFi network, which gives you the IP address of the local DNS server; however, the DNS Server was just rebooted and its cache is completely empty.

Suppose that the RTT between your computer and the local DNS server is 5ms, and the RTT between the local DNS server and *any* other DNS server is 60ms. Assume the iterated query is used and all responses have TTL of 5 hours.

(a) If you try to visit www.cs.ucla.edu, what would be the minimum amount of time that you need to wait before the web browser is able to initiate connection to the web server of UCLA CS? (Assume the ucla.edu name server is the authoritative DNS server for www.cs.ucla.edu)

(b) Using the similar assumption as in part(a), if you try to visit bruinlearn.ucla.edu one minute later, what would be the minimum waiting time?

(c) If you try to visit www.gradescope.com one minute later, what would be the minimum waiting time? (Assume the gradescope.com name server is the authoritative DNS server for www.gradescope.com)

(d) Using the similar assumption as in part(c), if you try to visit www.google.com one minute later, what would be the minimum waiting time?

Write your solution to Problem 6 in this box