CS 436 Intro to Networking: Homework 2

P1. (5pts) Please answer true or false for each of the following scenarios and then explain.

a. A user requests a Web page that consists of some text and three images. For this page, the client will send one request message and receive four response messages.
   False. If you receive 4, you sent 4.
b. Two distinct Web pages (for example, www.mit.edu/research.html and www.mit.edu/students.html) can be sent over the same persistent connection.
   True. Persistent HTTP leaves the connection open.
c. With nonpersistent connections between browser and origin server, it is possible for a single TCP segment to carry two distinct HTTP request messages.
   False. Each TCP segment can only carry one request.
d. The Date: header in HTTP response message indicates when the object was last modified.
   False. Date is request generation time
e. HTTP response messages never have an empty message body.
   Generally false, there will be a HTTP 204 No Content message

P2. (5pts) Imagine that you are trying to visit www.enterprise.com, but you don't remember the IP address the web-server is running on.

1. Can you send multiple DNS questions and get multiple answers in one message?
   Yes
2. To which DNS server does a host send their requests to?
   Local DNS Server
3. Which type of DNS server holds a company's DNS records?
   authoritative DNS server
4. In the example given, what is the IP address of the name server for enterprise.com?
   dns.enterprise.com
5. When you make the request for www.enterprise.com, your local DNS requests the IP on your behalf. When it contacts the TLD server, how many answers (RR) are returned?
   2

P3. (5pts) What is the difference between recursive and iterative DNS queries?

A recursive DNS lookup involves a client communicating with only one DNS server which that DNS server itself then reaches out to several other DNS servers to locate an IP address and return it to the client. In an iterative DNS query, the client communicates with each DNS server involved in the query directly.

Which one is used by DNS resolver and why?

DNS recursive queries have a faster resolution time than DNS iterative queries, due to caching.

What do TLD DNS servers do?

A TLD nameserver maintains information for all the domain names that share a common domain extension. For example, a .com TLD nameserver contains information for every website that ends in '.com'.

What do authoritative DNS servers do?

Authoritative DNS nameservers provide answers to recursive DNS nameservers with information on each / a collection of domain(s), such as the matching IP addresses.

P4. (5pts) Suppose within your web browser you click on a link to obtain a web page. The IP address for the associated URL is already cached in your local host, so there is no DNS look-up. Let RTT denotes the round-trip time between the local host and the server containing the web page. Suppose the HTML file references four very small objects on the same server. Neglecting transmission time, how much time elapses with

a. Non-persistent HTTP with no parallel TCP connections?
   = RTT3 + RTT2 + … + RTTn + 2RTT0 + 3*2RTT0
   = 8RTT0 + RTT1 + ... + RTTn
b. Persistent HTTP with no parallel TCP connections?
   = RTT1 + RTT2 + … + RTTn + 2RTT0 + RTT0
   = 3RTT0 + RTT1 + ... + RTTn
c. Persistent HTTP with pipelining (parallel connections)?
   = RTT1 + RTT2 + … + RTTn + 2RTT0 + RTT0
   = 3RTT0 + RTT1 + ... + RTTn

P5. (5pts) Consider an e-commerce website that wants to keep a purchase record for each of its customers.

1. Describe how this can be done with cookies (i.e. no login needed for customers).
   When the user first visits the site, the site returns a cookie ID. This cookie ID is stored on the user's host and is managed by the browser. During each subsequent visit (and purchase), the browser sends the cookie ID back to the site. Thus, the site knows when that user (more precisely, that specific browser) is visiting the site.
2. What are the privacy concerns on using cookies?

   The main problem with cookies is that websites can't distinguish if requests come from the actual user or someone else. This "cookie neutrality" is something cybercriminals can take advantage of to initiate a malicious action.

P6. (5pts) Please answer the questions in the two scenarios below.

1. Suppose a process in Host C has a UDP socket with port number 6789. Suppose both Host A and Host B, each send a UDP segment to Host C with destination port number 6789. Will both of these segments be directed to the same socket at Host C? If so, how will the process at Host C know that these two segments originated from two different hosts?

   Yes, both will be directed to the same socket. For each of the segments received at Host C the operating system will provide the process with the IP details to differentiate between the individual segments arriving at host C.

2. Suppose that a Web server runs in Host C on port 80. Suppose this Web server uses persistent connections, and is currently receiving requests from two different Hosts, A and B. Are all of the requests being sent through the same socket at Host C? If they are being passed through different sockets, do both of the sockets have port 80? Discuss and explain.

   For each persistent connection, the Web server creates a connection socket. Each connection socket on the web server is identified with a four-tuple. It contains source IP address, source port number, destination IP address, and destination port number. Host C will use these 4 fields in the datagram to determine to which socket it should pass the data. Thus, the request coming from Host A and Host B will pass through different sockets. The identifier for both of these sockets has 80 for the destination; however, the identifiers for these sockets have different values for source addresses. The source IP address will be specified by the socket identifier.

P7. (5pts) TCP provides reliable transport, flow and congestion control, and connection-oriented service, while UDP doesn't. So why bother to still have UDP? Please give at least three reasons.

UDP has Faster delivery of packets since UDP is lightweight and it has small size of packets than TCP, Quick processing time since there is no handshaking process, and Lack of overhead / Efficient use of bandwidth since UDP is not connection-oriented protocol and it does not provide the features of retransmission of packets

Also please list two application protocols that use UDP and discuss why?

SNMP (simple network management protocol) - because there is no need of sequencing as every response from the other side transmits as a single datagram

DNS (domain name server) – because UDP is preferred as it is fast, lightweight and has less overhead.

P8. (5pts) UDP and TCP use 1s complement for their checksums. Suppose you have the following three 8-bit bytes: 01011011, 01100110, 01110100.

What is the 1s complement of the sum of these 8-bit bytes? (Note that although UDP and TCP use 16-bit words in computing the checksum, for this problem you are being asked to consider 8-bit sums.) Show all work.

Find the sum of the first two bytes.

```
  0 1 0  1 1 0 1 1
+ 0 1 1  0 0 1 1 0
  1 1 0  0   0 0 0 1
```

Now, add the result of the sum and the third 8-bit byte.

```
    1 1 0 0 0 0 0 1
+   0 1 1 1 0 1 0 0
1   0 0 1 1   0 1 0 1
```

In this new result we got a carry bit 1, wrap around the carry bit by adding it to the new result.

```
  0 0 1 1 0 1 0 1
+             1
  0 0 1 1   0 1 1 0
```

Now find the 1's complement of the sum of three 8-bit bytes.

Invert the bits of the sum.

Thus, the final result is: 1 1 0 0 1 0 0 1.

With the 1s complement scheme, how does the receiver detect errors?

To detect errors, the receiver adds all the 16-bit words of the segment, including the checksum. The result should be all bits 1. If any bit of the result contains a zero, the receiver knows there is an error in the segment.

Is it possible that a 1-bit error will go undetected? How about a 2-bit error?

All 1-bit errors can be detected. But yes, in some cases two-bit errors can be undetected. For example, if the last digit of the first word is converted to a 0 and the last digit of the second word is converted to a 1.