CS 444
Computer Networks
Spring, 2017

Homework #2

Due: Sunday, February 12, at 10 pm, via Blackboard

This assignment will take quite some time, so please start early. Some of these are simple, but others require some thought and creativity. Some will be time consuming and require you to read outside materials such as RFCs.

Show all your work. Partial credit will be given.

Problems (parenthesis give number in 6th edition textbook):

Problem 1 (R1). List five nonproprietary Internet applications and the application-layer protocols that they use.

Problem 2 (R4). For a P2P file-sharing application, do you agree with the statement, "There is no notion of client and server sides of a communication session"? Why or why not?

Problem 3 (R5). What information is used by a process running on one host to identify a process running on another host?

Problem 4 (R7). Referring to Figure 2.4 in the textbook, we see that none of the applications listed in Figure 2.4 requires both no data loss and timing. Can you conceive of an application that requires no data loss and that is also highly time-sensitive?

Problem 5 (R9). Recall that TCP can be enhanced with SSL to provide process-to-process security services, including encryption. Does SSL operate at the transport layer or the application layer? If the application developer wants TCP to be enhanced with SSL, what does the developer have to do?

Problem 6 (R15). Why is it said that FTP sends control information "out-of-band"?

Problem 7 (R20). Look over your received emails, and examine the header of a message sent from a user with an .edu email address. Is it possible to determine from the header the IP address of the host from which the message was sent? Do the same for a message sent from a Gmail account.

Problem 8 (R21). In BitTorrent, suppose Alice provides chunks to Bob throughout a 30-second interval. Will Bob necessarily return the favor and provide chunks to Alice in this same interval? Why or why not?

Problem 9 (R23). What is an overlay network? Does it include routers? What are the edges in the overlay network?

Problem 10 (R27). For the client-server application over TCP described in Section 2.7, why must the server program be executed before the client program? For the client- server application over UDP, why may the client program be executed before the server program?

Problem 11 (P1). True or false?

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.
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.
c. With non-persistent connections between browser and origin server, it is possible for a single TCP segment to carry two distinct HTTP request messages.
d. The **Date:** header in the HTTP response message indicates when the object in the response was last modified.
e. HTTP response messages never have an empty message body.

Problem 12 (P2). Read RFC 959 for FTP. List all of the client commands that are supported by the RFC.

Problem 13 (P6). Obtain the HTTP/1.1 specification (RFC2616). Answer the following questions:

a. Explain the mechanism used for signaling between the client and server to indicate that a persistent connection is being closed. Can the client, the server, or both signal the close of a connection?
b. What encryption services are provided by HTTP?
c. Can a client open three or more simultaneous connections with a given server?
d. Either a server or a client may close a transport connection between them if either one detects the connection has been idle for some time. Is it possible that one side starts closing a connection while the other side is trans- mitting data via this connection? Explain.

Problem 14 (P14). How does SMTP mark the end of a message body? How about HTTP? Can HTTP use the same method as SMTP to mark the end of a message body? Explain.

Problem 15 (P18). Answer the following questions:

a. What is a *whois* database?

b. Use various whois databases on the Internet to obtain the names of two DNS servers. Indicate which whois databases you used.
c. Use nslookup on your localhost to send DNS queries to three DNS servers: your local DNS server and the two DNS servers you found in part (b). Try querying for Type A, NS, and MX reports. Summarize your findings.
d. Use nslookup to find a Webserver that has multiple IP addresses. Does the Web server of your institution (school or company) have multiple IP addresses?
e. Use the ARIN whois database to determine the IP address range used by your university.
f. Describe how an attacker can use whois databases and the nslookup tool to perform reconnaissance on an institution before launching an attack.
g. Discuss why who is databases should be publicly available.

Problem 16 (P21). Suppose that your department has a local DNS server for all computers in the department. You are an ordinary user (i.e., not a network/system administrator). Can you determine if an external Web site was likely accessed from a computer in your department a couple of seconds ago? Explain.

Problem 17 (P25). Consider an overlay network with N active peers, with each pair of peers hav- ing an active TCP connection. Additionally, suppose that the TCP connections pass through a total of M routers. How many nodes and edges are there in the corresponding overlay network?

Problem 18 (P26). Suppose Bob joins a BitTorrent torrent, but he does not want to upload any data to any other peers (so called free-riding).

a. Bob claims that he can receive a complete copy of the file that is shared by the swarm. Is Bob's claim possible? Why or why not?
b. Bob further claims that he can further make his "free-riding" more efficient by using a collection of multiple computers (with distinct IP addresses) in the computer lab in his department. How can he do that?

Problem 19 (P27). In the circular DHT example in Section 2.6.2, suppose that peer 3 learns that peer 5 has left. How does peer 3 update its successor state information? Which peer is now its first successor? Its second successor?

Problem 20 (P34). We have seen that Internet TCP sockets treat the data being sent as a byte stream but UDP sockets recognize message boundaries. What are one advantage and one disadvantage of byte-oriented API versus having the API explicitly recognize and preserve application-defined message boundaries?