

CS1356 Introduction to Information Engineering

Homework 8

Read textbook 4.4 and answer the following questions (before TCP/IP protocol)

1. What analogy does Textbook use to the message passing activity in the Internet? What are the three-level of hierarchy in the analogy?
2. What are the four layers of the Internet software? When a message is sent out, which order of those four layers does it go through? When a message is received, which order of those four layers does it go through?
3. Name two software units in the **application layer**.
4. What is the relation between the **transport layer** software and the application layer software? Which one should provide the receivers' addresses?
5. What is the most important task of the **transport layer** software? Why does the transport layer software need to divide a transmitted message into small segments? Are the small segments of a message sent out as a group or as individual units?
6. What is a "**packet**"? Why does each packet need be sent with a sequence number? Does the network layer software know the sequence numbers of packets?
7. What is the job of the **network layer** software? By which method does it know how to forward a packet?
8. What is the responsibility of the **link layer** software? Are all link layer software the same for different communication network?
9. Does the link layer software know whether a packet reaches its final destination or not? If it does not, how does it handle a received packet? And how the correct forwarding decision of a packet is made?
10. Which layers of software are involved in the intermediate stops, such as a router?
11. Which layer of software recognizes the completion of packet's journey? After it recognizes the completion, what does it do? Which layer of software assembles the received packets into a message? According to what? Where does the assembled message go?
12. What is the "**port number**" used for? Which layer of software appends the port number to messages? What are the port numbers for HTTP and FTP?
13. Summarize the software involved in the communication over the Internet.

Read textbook 4.1 and answer the following questions (before Combining Networks)

1. What is a **LAN**? What is a **MAN**? And what is a **WAN**?
2. What is an **open** network and what is a **proprietary** network? Give examples.
3. What is a bus network topology? And what is a star network topology? Give an example of network system built on each network topology.
4. What is the "**Access Point**" in a wireless network?
5. How to distinguish a bus network and a star network? What is a "**hub**"? Is a network linked by a hub a bus network or a star network?
6. What is a "**Protocol**"? Why is the development of protocol standards an indispensable process in the development of networking technologies?
7. What is the **CSMA/CD**? What kind of network topology using this protocol?
8. In CSMA/CD, how a message is sent and how a message is received? How do machines sharing a bus avoid the collision of sending messages? Can the collision be avoided always? If not, how do machines detect the collisions? After detecting the collision, what do they do to try avoiding the possible collisions when resending their messages?
9. Why is CSMA/CD not compatible with wireless star network? What are two situations that machines cannot detect collisions? What is the **hidden terminal problem**?
10. What is the **CSMA/CA**? What kind of network topology using this protocol? Can the collisions of sending messages be eliminated completely by using CSMA/CA?
11. In CSMA/CA, who has the advantage of sending messages? If a machine finds a channel is silent (no one is using it), can it send messages immediately? If not, under what conditions can it send the messages? Why this protocol can avoid the collisions between newcomers and those that have already been waiting? Can you see that this protocol is also able to reduce the collisions between waiting machines? How?
12. Can the waiting-and-send protocol solve the hidden terminal problem? Why or Why not? Textbook mentions another protocol to solve the hidden terminal problem, how does it do? How does the requesting machine know there is a hidden terminal problem? When can the requesting machine send out the message to the AP? What would happen if the "request" messages collide?