Note: These answers are only reference answers. Other answers may also correct as long as you can provide reasonable explanations!

1. A factor in the delay of a store-and-forward packet-switching system is how long it takes to store and forward a packet through a switch. If switching time is 10 μsec for each switch, is this likely to be a major factor in the response of a client-server system where the client is in Guangzhou and the server is in Beijing (suppose there are 40 switches on the path between them)? And how about that the client is still in Guangzhou, while the server is in Shenzhen (suppose there are 15 switches on the path between them)? Assume the propagation speed in copper and fiber to be 2/3 the speed of light in vacuum.

Answer: The speed of propagation is 200,000 km/sec or 200 meters/μsec. In 10 μsec the signal travels 2 km. Thus, each switch adds the equivalent of 2 km of extra cable.

If the client and server are separated by 2158 km (Beijing), traversing 40 switches adds only 80 km to the total path, which is only 0.04%. Thus, switching delay is not a major factor under these circumstances.

For the server in Shenzhen, client and server are separated by 140km, traversing 15 switches adds only 30 km to the total path, which is 21%. Then, switching delay is a major factor under this circumstance.

2. Use your own words to explain the reasons for using layered protocols? What is one possible disadvantage of using layered protocols?

Answer: Among other reasons for using layered protocols, using them leads to breaking up the design problem into smaller, more manageable pieces, and layering means that protocols can be changed without affecting higher or lower ones.

One possible disadvantage is the performance of a layered system is likely to be worse than the performance of a monolithic system, although it is extremely difficult to implement and manage a monolithic system.

3. Two networks each provide reliable connection-oriented service. One of them offers a reliable byte stream and the other offers a reliable message stream. Are these identical? If so, why is the distinction made? If not, give an example of how they differ.

Answer: Message and byte streams are different. In a message stream, the network keeps track of message boundaries. In a byte stream, it does not.

For example, suppose a process writes 1024 bytes to a connection and then a little later writes another 1024 bytes. The receiver then does a read for 2048 bytes.

With a message stream, the receiver will get two messages, of 1024 bytes each.

With a byte stream, the message boundaries do not count and the receiver will get the full 2048 bytes as a single unit. The fact that there were originally two distinct messages is lost.

4. When a file is transferred between two computers, two acknowledgement strategies are possible. In the first one, the file is chopped up into packets, which are individually acknowledged by the receiver, but the file transfer as a whole is not acknowledged. In the second one, the packets are not acknowledged individually, but the entire file is acknowledged when it arrives. Discuss these two approaches, e.g., their suitable scenarios, advantages and disadvantages.

Answer:

If the network tends to lose packets, it is better to acknowledge each one separately, so the lost packets can be retransmitted.

On the other hand, if the network is highly reliable, sending one acknowledgement at the end of the entire transfer saves bandwidth in the normal case (but requires the entire file to be retransmitted if even a single packet is lost).