**(1) (20 points)** Remote Method Invocation (RMI) is a Java API feature that allows an application to invoke a method on a remote object that resides in another virtual machine where the other virtual machine may exist on a separate computer connected by a network. RMI is implemented using sockets. However, rather than being concerned with the messy details of socket programming, developers write ordinary object-oriented programs using RMI, and let the RMI library handle the details of managing the socket connection. As a demonstration of RMI, the Hermosa web server and Redondo web browser were entirely written in Java using RMI rather than lower-level sockets. Using this information describing RMI, as well as Hermosa and Redondo, answer the following questions using the protocol stack we are covering this semester:

* (3 points) What protocol is being used at the application layer?

RMI protocol. Although the text shows that RMI is a Java API feature, but it is implemented using sockets. So, I think RMI is a protocol at the application layer just like http, but RMI is based on java environment.

* (2 points) Does this application layer require reliability?

Yes, because developers write ordinary object-oriented programs using RMI.

* (3 points) What protocol is being used at the transport layer?

TCP, but I thought there is a little different with the traditional socket web application, I believe there might be another protocol beyond TCP at the transport layer that let RMI library handle the details of managing the packet from the socket connection.

* (2 points) Is this transport layer protocol reliable?

Yes.

* (5 points) Briefly describe what this transport layer protocol must do with each packet it transmits.

It will use three-way hand-shake to build up connection firstly, and for each packet it transmits, the sender will refresh its Seg number, and receiver will use this number to check if the packet is correctly sent and refresh its ACK number.

However, as what I am thinking above, I thought there might be another protocol beyond TCP at the transport layer that let RMI library handle the details of managing the packet from the socket connection, therefore, when the package arrive to RMI library, the library will using the seg number tell the order of the packet, collect it and send to “real receiver”.

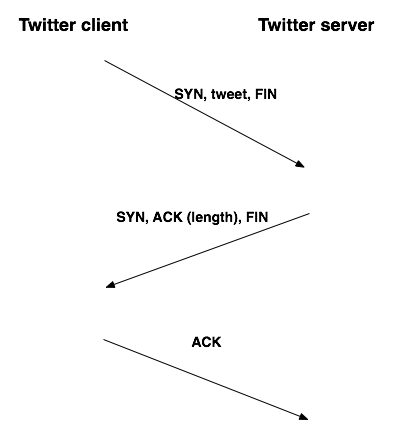
(3 points) The transport layer then hands off the packet to the network layer. What is the responsibility of this layer?

The responsibility of the network layer is takes segments from transport layer, encapsulates each segment into a datagram and then move the datagram from a sending host to a receiving host along a series of routers.

* (2 points) Is the network layer reliable?

The network layer is not reliable.

**(2) (10 points)** Twitter allows tweets to be up to 280 characters in length. Rather than using traditional TCP sockets for sending tweets from a phone to a server, Twitter has explored using *Transactional TCP* that avoids the TCP implementation of (1) establish a connection using the 3-way handshake, (2) exchange data, and then (3) breaking down the connection. Transactional TCP works by having the client send the SYN, tweet (including a sequence number), and FIN in a single packet. The server responds with a SYN, ACK of the tweet, and a FIN. The client then acknowledges receipt of this packet with a final ACK, and the connection is terminated. This is demonstrated below:



Can transactional TCP be used for transmitting tweets that can fit into a single packet, or are there problems with this idea? Be sure to provide support for your answer.

There are several problems with the idea using transactional TCP to transmit tweets into a single packet.

If a request is sent to a server and the server processes the transaction, but before it sends back an acknowledgment, the process crashes. The client-side times out and retransmits the request; if the server process recovers in time, it will repeat the same transaction. Based on Transactional TCP structure, it will complete duplicate times connection and pose these tweets duplicate times.

Another thing it may happen is that even though these tweets can fit into a single packet, it may request more room in the packet for server to distinguish which range between characters belongs to one tweet and which range between characters belongs to another twee. So that may cause space error when transmit data.

The most serious thing that I’m concerned is security issue by using transactional TCP. Because there is no three-way handshake in this process, which means there are no authentication for the ip address which the client sends to server. An attacker can create a packet with a false IP address in it and send to twitter server, the server will accept the packet immediately and send data back to the original client. That may cause security issue.

**(3) (6 points)** Consider a web browser that wants to retrieve a web resource at a given URL, but the IP address of the web server is unknown. Provide three protocols that are in use in this scenario.

The web browser will use DNS protocol transfer URL to IP address.

After it get the IP address, it will use HTTP protocol to ask web recourse from that IP address.

The requestion and web source will be transmitted by TCP protocol.

**(4) (4 points)** TCP transmits data as a byte stream which starts at a given sequence number. This byte stream serves as a logical sequence of bytes that are exchanged between the pair of communicating transport layers. If TCP uses a byte stream (which implies continuity), explain how it is possible for some data to arrive out of order.

When transport data, it will divide data into several packet and each packet will have sequence number when deliver. In the deliver process, different packet may have different routes when deliver, and some of packets may lost which leads that sender has to re-send the lost packet.

All these kinds of traffic issue will cause it is possible for some data to arrive out of order.

**(5) (15 points)** The IP addresses of two hosts are

*Hermosa = 165.26.21.115* and *Redondo = 202.11.95.150*

Assume client-side port numbers will be assigned from *Hermosa* beginning at 2500, and from *Redondo* beginning at 6000, and will increment linearly. Refer to [Wikipedia](https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers) for a listing of port numbers for well-known services when the two hosts are servers.

Answer the following questions **independently** of one another (i.e. treat them as four separate, independent questions.)

* (3 points) If a client on *Hermosa* wants to reach a server on *Redondo*, but it does not know its IP address, it must be looked up using DNS. What is the socket pair that comprises this connection?

If a client on Hermosa looks up Ip using Google public DNS:

165.26.21.115:2500 8.8.8.8:53

* (3 points) What is the socket pair if a client on *Redondo* wants to set up a ssh session on the *Hermosa* server?

202.11.95.150:6000165.26.21.115:22

* (3 points) If someone on *Hermosa* wants to send an email message to someone on *Redondo* using SMTP, what is the socket pair between the *Hermosa* client and *Redondo* server?

165.26.21.115:2500 202.11.95.150:25

* (6 points) *Redondo* is running a web server that supports persistent connections, but only allows three resources (files) to be downloaded with each connection. A web browser on *Hermosa* requests an HTML file images.html that refers to four images. What are the socket pair(s) between *Redondo* and *Hermosa* that are used to request the images.html file, as well as four separate images?

images.html :165.26.21.115:2500 202.11.95.150:80

first image: 165.26.21.115:2500 202.11.95.150:80

second image: 165.26.21.115:2500 202.11.95.150:80

third image: 165.26.21.115:2501 202.11.95.150:80

forth image: 165.26.21.115:2501 202.11.95.150:80

**(6) (6 points)** A network has been assigned the IP address 151.36.217.0/22

* (2 points) How many hosts may be present on this network?

If we exclude broadcast address and gateway, there are 2^10 -3 hosts may be present on this network.

* (2 points) Provide an example of a host IP address on this network.

151.35.217.3

* (2 points) What is the netmask of this network?

255.255.252.0

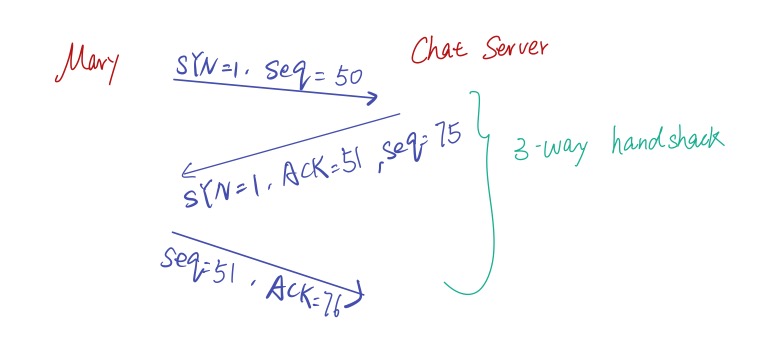
**(7) (4 points)** A host has been assigned the IP address 112.52.42.116/22. What is the IP address of the network?

The netmask of this network is 255.255.252.0, using the bitwise OR of the IP address of the host and the netmask, we can get the IP address of the network is 112.52.40.0

**(8) (15 points)** Later this semester your class will design an application protocol for a chatroom that runs over TCP. People (clients) will log on to the chat server where they will be able to chat with others in the chatroom. Assume Mary is connecting to the chat server, and her initial sequence number is 50. The initial sequence of the chat server is 75. The following table lists the time each segment is sent and its length in parentheses:

| **Time** | **Mary** | **Chat Server** |
| --- | --- | --- |
| T0 | *initiates connection* |  |
| T1 | Login: Mary (11) |  |
| T2 |  | Mary has joined the chatroom. (29) |
| T3 | Hi everyone! (13) |  |
| T4 |  | MARY: Hi everyone! (18) |
| T5 |  | ELIZABETH: Hi Mary - good to see you've joined us. (44) |

* (4 points) What are the values of sequence (**seq**) and acknowledgment (**ack**) fields for the communication setup using the three-way handshake beginning at time T0? Be sure to specify when the **SYN** field is set. (You only have to show the three-way handshake.)



* (6 points) At time T2, the server sends the message to Mary, with a seq=76 and ack=62. What are the values of the seq and ack fields for the exchanges at times T3, T4, and T5?

T3, from Mary to Chat Server, Seq=62, Ack = 105

T4, from Chat Server to Mary, Seq=105, Ack = 75

T5, from Chat Server to Mary, Seq=149, Ack = 75

* (5 points) Assume the server's packet at time T5 is lost during routing. Explain how this packet will ultimately arrive at Mary's connection.

Because the server’s packet at time T5 is lost, which means Mary will not receive this package and do any reply on this packet.

After time out, the server will send this packet to Mary again. If this time the packet isn’t lost, Mary will receive the packet and send acknowledgment to server. However, if the acknowledgment is lost during routing, the server will send the packet repletely until receive Mary’s acknowledgment.