Solutions to Assignment Unit 6

1. Assuming standard 1500 byte Ethernet max payloads: how many IPv4 fragments will be needed to transfer 2000 bytes of user data with a single UDP send? And, how do the 2000 bytes get split over the frags?

Two IPv4 fragments will be needed. The first one will be in a full size Ethernet frame. So, 1500 – 20 (IP header) – 8 (UDP header) = 1472 bytes of user data go in 1st frag. The remaining (2000 – 1472 =) 528 bytes go in the 2nd and final frag.

2. In spite of its conceptual elegance, RPC (Remote Procedure Call) has a few problems. Discuss any 3 of those in brief.

- Protocol Requirements – As each procedure being called could have its own unique specification in regards to parameters, and perhaps even the format of the input and output that is different between the client and server. For instance even a numeric value such as an integer could be a 32 bit number on one system and a 64 bit number on the other.
- Communication handling – We can use any of the transport protocols or even construct our own to communicate, further to this, if we use a connectionless protocol how does the client-server handle the situation where a request or response goes missing?
- Security – RPC is executing code on a remote system, we need to have a means to ensure that the person executing this code has rights to do so, RPC does not specifically define any means of securing the communication and thus our RPC procedures need to authenticate and authorize.

3. Why is time stamping needed in Real Time Applications? (This is in context to RTP)

Time stamping is used for synchronization as well as discarding old invalid packets. As the receiver processes the frames sent, any packets which had an older time stamp would obviously mean those packets were delayed in transmission. Imagine if this was a voice application, if a few bits of audio go missing, you would not want those to suddenly reappear in the middle of the stream, it would become in audible and create problems. Thus the timestamps can be used to drop packets that no longer have relevance.

4. Why does UDP exist? Would it not have been enough to just let user processes send raw IP packets?

UDP exists to support real time applications, applications where data packets are time-sensitive and would prefer to drop the data than wait, and for applications that handle error checking and correction thus removing the need for this overhead at the transport level. While it could be argued that essentially we are allowing application to nearly talk direct, and thus could allow them to just send raw IP packets, the truth is that UDP still performs functions that the transport layer provides such as adding the port number to the service which is crucial for talking to the right process.

5. Explain how QUIC eliminates a couple of RTTs usually needed at the start of a secure web connection.

QUIC allows data to be included in the first packet of a connection (which saves one RTT compared to TCP), and usually eliminates a second RTT during TLS encryption negotiation.