Homework 3. Internetworking. EP2120 . Pierre FLEITZ

I) Socket Interface :

a) This is a sample code for a server and it uses TCP. We can see that this is a server using TCP because there is a listen and an accept. That means that the server has to wait and accept a connection, therefore it is a connection-oriented server. But there are also the functions recv() and send() used, this is proper to a TCP connection.
b) Since this sample of code is a server using TCP, it's a connection-oriented communication, but there is no fork wich means that the server can only deal with one request at the time and not concurrent requests, therefore it is an iterative communication. We have a **connection-oriented**, **iterative communication**.
c) The main limitation is that this type of communication is not suitable for transaction that last long because in this type of communication the server can only be used by one client at the time, if the communication lasts for a long time queues (connection requests from others clients) can build up quickly. For lengthy transaction it is better to use a concurrent server !

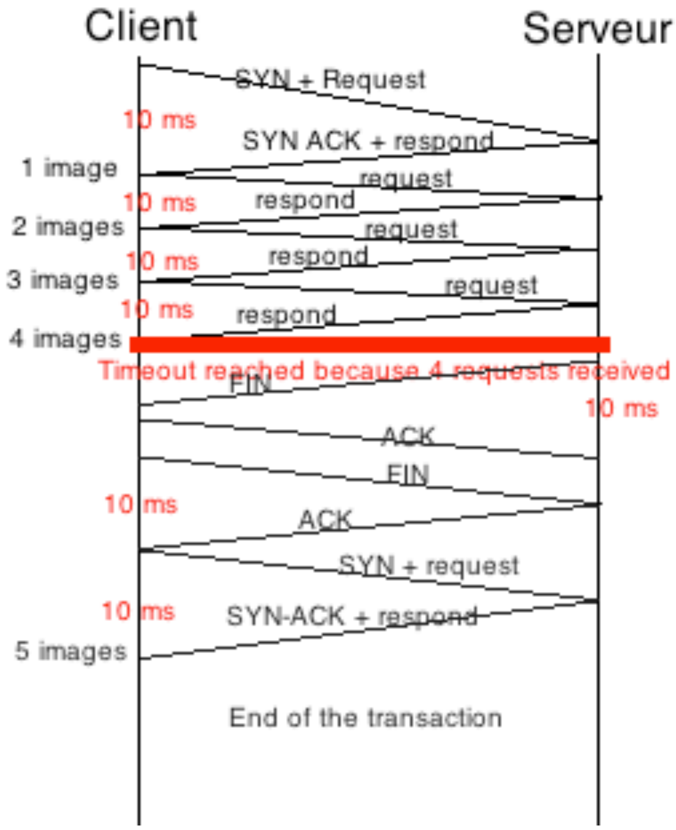
II) Multimedia communication :

a) In order to compensate the jitter the playback buffer should at least be set to 200 ms.
b) UDP has no prevision for timestamping, sequencing, or mixing. Therefore we need another transport layer protocol, RTP, to make up for the deficiencies of UDP.
RTP provides the ability to handle real-time traffic on the internet, it must be used with UDP and the main contribution of RTP are timestramping, sequencing and mixing facilities.
c) After reading the TCP/IP book you have to agree with that sentence. Indeed TCP with all its sophistication is not suitable for interactive multimedia traffic because in TCP when a packet is lost we retransmit it and during a multimedia communication, a live video streaming for example, we cannot allow retransmission of packets. It would upset the whole idea of timestamping and playback.
d) Firstly, TCP is well integrated with http, that allows multimedia on the web. Secondly, in practice, streaming content using TCP is more likely to pass through firwalls.

III) Web :

a) www.opera.org:1935/claypool
b) HTTP version 1.1. The browser is requesting a persistent connection (Connection : keep-alive).
c) The server accepts a persistent connection (connection : Keep-alive).

d) We gonna have to make 5 requests but we can see that we have a keep-alive maximum at 4 requests (keep-alive : timeout=10, max=4), it means that after 4 requests the connection ends and we need to open a new one.
If we consider that opening a TCP connection requires 1 RTT, that closing a TCP connection requieres 2 RTT, we gonna have 7 * RTT so 70 ms. (without counting the connection termination at the end if we do we gonna have 90ms).
Timetable :



e) We can see that there is a cookie value, it means that the client's browser found it in the cookie directory. Therefore we can conclude that the client received a cookie from that same server before thus that the client has visited this web site before.

f) We can see that cache-control is set to "No-cache" therefore there is no response stored in the cache, the proxy server is going to send the request to the corresponding server.

IV) DHCP :

a) Because he didn't received the DHCP ACK. Before receiving the DHCP ACK the client isn't allowed to use the network configuration received from the DHCP server earlier. Client must include server ID (not shown here) to inform about selected offered address.

b) After receiving the DHCP ACK he will be able to use the IP address offered by the server.

c) Knowing that a client can receive DHCP offers from multiple servers and that it will accept only one DHCP offer, we can understand that the client will send the DHCP request as a broadcast message so servers can be informed whose offer the client has accepted and they can withdraw any offers that they might have made to the client and return the offered address to the pool of avaible addresses

d) It is to prevent a problem when the reply, from the server to the client, is broadcast. It prevent the use of the same two destination port numbers. And it is considered "bad manner" to broadcast to a random port number