Jeff Foreman

Nimitt Tripathy

CSE 160

5 December 2017

Project 4 Discussion Questions

1: The chat client and server application as described above uses a single transport connection in each direction per client. A different design would use a transport connection per command and reply. Describe the pros and cons of these two designs.

i. If the application used a single transport connection per client, far less three-way handshakes would be required. Thus, the application would be moderately quicker. However, this version is far less secure, as it keeps the channel open the whole time the client is connected. Over and above this, it is more strenuous on the server itself, as it must maintain these connections constantly.

ii. If the application used a single transport connection per command and reply, then far more three-way handshakes would be required, thus putting a strain on the network. However, this is a more secure method, and puts less strain the server.

2: Describe which features of your transport protocol are a good fit to the chat client and server application, and which are not. Are the features that are not a good fit simply unnecessary, or are they problematic, and why? If problematic, how can we best deal with them?

Our transport protocol is one that uses the one transport connection per message and reply concept. With how our Transport layer is implemented, it makes implementing the chat client and server much simpler. However, as discussed in the previous question, this causes

unnecessary traffic on the network in terms of three-way handshakes being conducted. One way to combat this would to be use a single connection for a few messages before it times out.

3: Even if you did not implement the extra credit application, read through its protocol specification. Describe which features of your transport protocol are a good fit to the web server application, and which are not. Are the features that are not a good fit simply unnecessary, or are they problematic, and why? If problematic, how can we best deal with them?

Our protocol provides an efficient and secure way to exchange DATA packets via the three-way handshake prior to the exchange, and the data following the synchronization was complete. One problem with it however, is that a three-way handshake occurs per DATA packet, which means this will put extra strain on the network. To deal with this, have one three-way handshake per transfer, that is, send all the necessary DATA packets and then close the connection.

4: Describe one way in which you would like to improve your design.

One way we would like to improve our design would be to have one TCP connection per client. This would be more efficient in sending messages as the three-way handshakes would not be constantly occurring. Also, with the amount of nodes we deal with and their memory requirements, the simultaneous connections would not be that stressful to the server node.