Nombre: Tomás Maturana
RUT: 19.186.973-7

Tarea 1

1) En caso de que se pierda todo el mensaje, el cliente de eco quedará esperando la respuesta indefinidamente, hasta que se corte el proceso manualmente (o al menos con la implementación que se hizo). Si no se quisiera que se deba reiniciar el cliente manualmente, se podría poner un *timer* en el cliente o en los proxys para limitar la espera y así reconectar automáticamente, avisando al cliente sobre la pérdida de datos. En caso de que se pierda parte del mensaje, la respuesta que le llegará al cliente estará incompleta. Por otro lado, el servidor eco no se vería afectado, ya que si le llega el mensaje, manda lo que recibió y *termina su trabajo*; si no le llega ni una parte del mensaje, entonces hace nada.

2) Ya que al cliente eco se le ha privado la posibilidad de conectarse mediante TCP hacia el exterior, el proxy1 debería estar corriendo en el mismo computador para que se mantenga el funcionamiento. En su defecto, el cliente eco podría conectarse directamente mediante UDP al proxy2 para mantener el mismo funcionamiento, pero en este caso, serían 3 programas los que estarían en ejecución. Por otro lado, ya que el servidor eco y el proxy2 no tienen limitaciones en sus conexiones, podrían estar ejecutándose en computadores diferentes sin problema.

3) La mejor manera que se me ocurre es: cuando se conecte un cliente al proxy1, que éste le envíe un mensaje al cliente, preguntándole a qué servidor se desea conectar y en qué puerto, luego, el cliente deberá mandar un primer mensaje con esta información. Si el mensaje incluye la información necesaria, crea un socket para realizar la conexión deseada; si no, se podría volver a hacer la pregunta inicial.