RMI and UDP Coursework

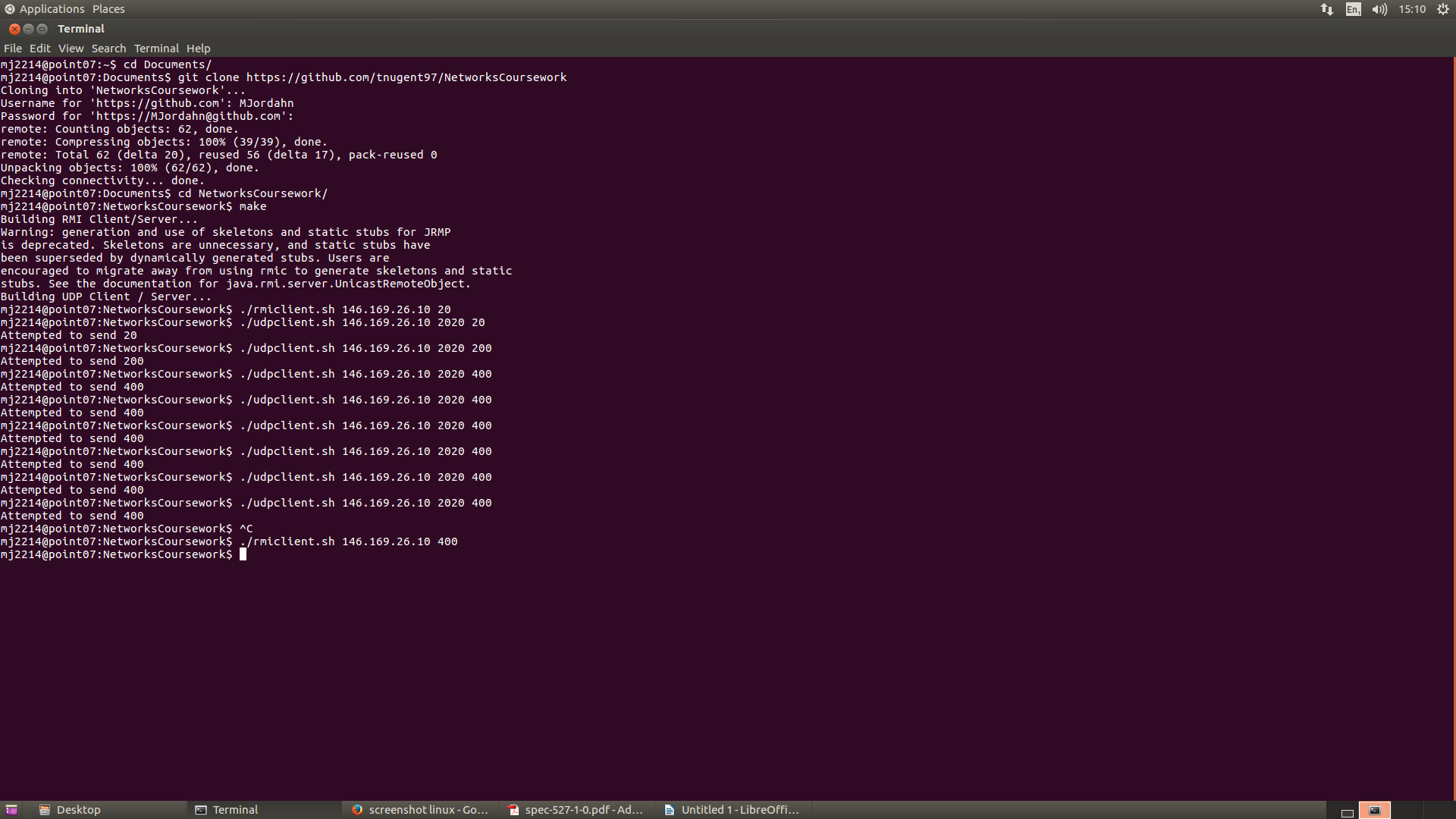
Thomas Nugent (tn1115) & Mikkel Jordahn (mj2214)

For UDP there may be two reasons for message loss. One reason is that noise may distort packets when send, and thus destroy the packet, so that it never reaches its destination. The other reason that UDP may have message loss is because the client does not care whether or not the server is receiving. The server has a receive buffer, which will hold messages if the server is not listening at the time the messages are sent. If the client however, sends more messages than the buffer can hold, before the server can receive them, the messages that exceed the buffer will then be dropped and lost. In RMI on the other hand, messages are not lost, because TCP is used in this case. This means that the client is waiting to receive back a “Received message” reply from the server. If it doesn’t receive this, the client will then resend the same message.

Generally speaking, the only issue we encountered was that the receiving buffer was filled, and therefore could not receive any more messages. As can be seen in the screenshots below, it was typically somewhere after 300 messages that the buffer was filled, and from then on, the rest of the messages were dropped. Sending messages within the lab, it appears that the internet connections are too fast, and therefore no messages were dropped.

RMI is incredibly more reliable than UDP. This is, as mentioned earlier, because RMI utilizes TCP, which means that should messages not be received by the server, then the client will resend the message. In addition, TCP also has a sequence number which is used to identify the order of the bytes when received. UDP on the other hand, does not utilize any reliability mechanisms to ensure that the data is received accurately. The cost of this reliability is the speed of the mechanism, although we did not experience any significant effect of this in the lab because the college network is very sophisticated.

UDP was significantly easier to program than RMI. This is mainly because UDP is a much simpler concept and so was easier for us to understand. It also relies on sockets, meaning you can easily send data using a datagram packet, on a socket that you have set up prior using the address and port. RMI however relies on the client binding to the registry and then waiting for a reply from the server to say that a message has been received, otherwise it will send the message again. UDP is very much a ‘send and forget’ mechanism, whereas RMI needs extra code to check for a reply and deal with each case accordingly.

  
A screen shot from the client machine showing the running of both RMI and UDP.