COMP3331

Assignment 1- Report

Implementation

Packets

The implementation of the packet is by far the most complicated part of the assignment. The packets for both my Sender and Receiver utilises a String array to store all my information.

public static String[] packet (boolean SYN, boolean ACK, boolean FIN, int SeqNo, int AckNo, String partialString)

with the exception of receiver's packet not have an extra array since it doesn't need to store strings. This is then converted to a Byte Array through a converter which utilises *ByteArrayOutputStream* and *ObjectOutputStream* to convert the String Array to the form of a Byte Array for the Datagram Packet.

3-way Handshake

The implementation is the same as its name. Receiver waits for the Sender's packet and check if its a SYN, if so it reply with a SYNACK and wait for an ACK. this will then trigger the connected to 1 and allow the intake of data packets.

Data Input and Output

On the Receiver end, Once the 3-way handshake sets the connected flag to 1, it can take in data and reply with and ACK. Theres and if statement to look out for receiving the FIN flag to initiate the 4-way connection teardown.

On the Sender end, it takes the text file and and store it as a string and using that string it splits it to the MMS by converting it to a byte first then converting it back to a string then storing it into the String Array. Then it goes through the PLD and see whether to drop or send the packet. Then it waits for a reply, if it triggers the timeout it will set the byte position and sequence number back and retransmit the packet. Once its done it will initialise the 4-way connection teardown.

Only feature not implemented is the MWS.

Diagram

Sender & Receiver

SYN	ACK	FIN
SEQUENCE NUMBER	ACKNOWLEDGE NUMBER	

SYN, ACK, FIN are just a boolean. The program will check for the appropriate flags easily since its in the form of a String array. which basically requires you to only type to check for SYN:

if (String[0].equals("true"){

...

}

The Sequence number is for the program to keep track of its own packet. for sender it will increment by the size of each string sent.

The Acknowledge number is for the program to keep track of other's sequence number for sender it will keep track of the receiver's sequence and vice versa.

Question

a) 50ms since it will be able to quickly retransmit the data due to its high packet drop rate.

b) i) 385 - 18321 ii) 385 - 69412ms iii) (result not available since it times me out on my connection establishment due to how small the timeout)

The result may are similar, which may be due to MWS not implemented.