

Testing Documentation – Final Project

Testing ID	Description	Expected Outcome	Result
Transmitter			
1a Send and Wait	Upon running the server binary, with the network emulator's IP address, split a resource file into packets and send a window of packets with sequence numbers then wait for all ACKs (updating the timeout interval with each ACK received) from client before sliding the window forward to send new packets	<p>-Transmitter logs contain a trace of packets with sequence numbers being sent in a window size, information on updated timeout intervals/when all ACKs are received, trace of packets being sent in new window size.</p> <p>-Network Emulator shows corresponding datagrams being sent from both transmitter (DATA)/receiver(ACKs).</p> <p>-Packet Capture shows corresponding datagrams being sent between transmitter/network emulator/receiver.</p>	As Expected
1b ARQ	Upon sending a window of packets, the server shall start a timer for the batch of packets sent.	<p>- Transmitter logs contain a trace of packets being sent, information on updated timeout intervals, output of when RTT exceeds timeout intervals</p> <p>- When timeout events occur, the network emulator will display retransmitted DATA packets with seqNum corresponding to the packets that have yet to be ACKed. In addition, after all ACKs have been sent, depending on the number of timeout events that occurred in the last window, the window size will be adjusted.</p> <p>- Packet Capture shows corresponding datagrams being sent between transmitter/network emulator/receiver.</p>	As Expected
1c EOT	Upon sending the last window of packets, and receiving ACKs for them all the transmitter will send EOT packets to indicate the end of transmission.	<p>- Transmitter logs contain a trace of EOT packets being sent after receiving the last ACKs for the last window of DATA packets. Terminates after sending.</p> <p>- Network Emulator shows EOT packets being sent after last ACK is received by transmitter</p>	As Expected

Receiver			
2a ACK	Upon running the receiver binary, the receiver listens on the specified port for incoming packets. Upon receiving a data(DATA) packet, the receiver immediately responds with an acknowledgment(ACK) packet.	<ul style="list-style-type: none"> - A message confirming the DATA packet was received is logged to the log file and STDOUT, along with the packet details. - A message confirming the ACK packet was sent is logged to the log file and STDOUT, along with the packet details. - ACK packet's acknowledgment number matches the sequence number of the DATA packet being acknowledged. - Network Emulator shows corresponding packets being sent from both transmitter (DATA)/receiver(ACK). - Packet Capture shows corresponding datagrams being sent between transmitter/network emulator/receiver. 	As Expected
2b In/out of order data handling	Upon packet arrival, if the packet is received in order(next expected sequence number) the packet data is immediately written to the output file; otherwise, the packet is buffered and the data is written before the first packet in the next transmitter's window is processed.	<ul style="list-style-type: none"> - New packets' data is visible in the output file as the new packets arrive in order. - All buffered packets' data is written when the new transmission window begins. - The checksum of the received file matches the checksum of the sent file at the end of the transmission. 	As Expected
2c EOT	Upon receiving the end of transmission (EOT) packet, the receiver application is terminated.	<ul style="list-style-type: none"> - A message confirming the EOT packet was received is logged to the log file and STDOUT, along with the packet details. - Network Emulator shows corresponding EOT packet being sent from transmitter to receiver. - Packet Capture shows corresponding datagrams being sent between transmitter/network emulator/receiver. 	As Expected

Network Emulator			
3a App Start	Upon user launching the application, the network emulator will load common configuration parameters into the application and set the appropriate UI elements with the specified values.	<ul style="list-style-type: none"> - Settings table will populate with the endpoint details (IP/Port) for the transmitter, receiver, and network emulator, payload length, max window size. - Bit Error Rate and packet delay sliders are set to default value. 	
3b Start button	Upon user pressing the start button, the network emulator is ready to relay packets from transmitter to receiver and from receiver to transmitter.	<ul style="list-style-type: none"> - Status label changes from red "Stopped" to green "Active". <p><u>Upon packet arrival, the following are updated:</u></p> <ul style="list-style-type: none"> - Packet table (relative time, window size, packet type, retransmit, seq/ack#, source/dest IP/port) - Summary table (total capture time, total packets, dropped packets, retransmits) - Sequence Time graph (sequence number vs relative time) 	As Expected
3c Stop button	Upon user pressing the stop button, the network emulator halts the relay of packets from transmitter to the receiver and from the receiver to the transmitter.	<ul style="list-style-type: none"> - Status label changes from green "Active" to red "Stopped" - Time Sequence Graph, Packet Table, summary table stops updating 	As Expected
3d Reset button	Upon the user pressing the reset button, the network emulator halts the relay of packets (if it is currently active) and resets the UI to the initial program entry state.	<ul style="list-style-type: none"> - Status label changes from green "Active" to red "Stopped" - Time Sequence Graph, Packet Table, summary table stops updating and is refreshed to a clean state showing no data. 	As Expected
3e Save button	Upon the user pressing the save button, a window prompt to save packet table data to a CSV file is presented to the user.	<ul style="list-style-type: none"> - Packet Table data matches the saved CSV file. 	As Expected
3f Packet Delay	Upon moving the packet delay slider, the changes to the delay are applied to the packets.	<ul style="list-style-type: none"> - The packet delay value matches the value set by the slider. - The label above the slider reflects the packet delay value. 	As Expected
3g Bit Error Rate	Upon moving the bit error rate slider, the changes to the bit error rate are applied to the packet loss events.	<ul style="list-style-type: none"> - The BER value matches the value set by the slider. - The label above the slider reflects the BER value. 	As Expected

Test 1a: Send and Wait

Refer to nominal-transmitter-network_emulator.mkv

```
utopia@DESKTOP-GYRIAPU:/mnt/c/Users/Derek_Wong/Documents/PacketLens7005/transmitter$ ./build/transmitter.o 192.168.1.78
[2020-12-3 12:52:55] Host found: 192.168.1.78
[2020-12-3 12:52:55] The network emulator's port is: 7002
[2020-12-3 12:52:55] Sending data in file path: ./resource/message.txt
[2020-12-3 12:52:55] Number of lines in the file are: 150
[2020-12-3 12:52:55] Current window size: 1
[2020-12-3 12:52:55] Sent DATA (seqNum: 1)
[2020-12-3 12:52:55] Window of packets sent, waiting for ACKs
[2020-12-3 12:52:55] Received ACK (ackNum: 1)
[2020-12-3 12:52:55] Updating timeout interval: 2478
[2020-12-3 12:52:55] All ACKs received

[2020-12-3 12:52:55] Current window size: 2
[2020-12-3 12:52:55] Sent DATA (seqNum: 2)
[2020-12-3 12:52:55] Sent DATA (seqNum: 3)
[2020-12-3 12:52:55] Window of packets sent, waiting for ACKs
[2020-12-3 12:52:55] Received ACK (ackNum: 2)
[2020-12-3 12:52:55] Updating timeout interval: 2712
[2020-12-3 12:52:55] Received ACK (ackNum: 3)
[2020-12-3 12:52:55] Updating timeout interval: 2769
[2020-12-3 12:52:55] All ACKs received

[2020-12-3 12:52:55] Current window size: 4
[2020-12-3 12:52:55] Sent DATA (seqNum: 4)
[2020-12-3 12:52:55] Sent DATA (seqNum: 5)
[2020-12-3 12:52:55] Sent DATA (seqNum: 6)
[2020-12-3 12:52:55] Sent DATA (seqNum: 7)
[2020-12-3 12:52:55] Window of packets sent, waiting for ACKs
[2020-12-3 12:52:55] Received ACK (ackNum: 4)
[2020-12-3 12:52:55] Updating timeout interval: 2690
[2020-12-3 12:52:55] Received ACK (ackNum: 5)
[2020-12-3 12:52:55] Updating timeout interval: 2546
[2020-12-3 12:52:55] Received ACK (ackNum: 6)
[2020-12-3 12:52:55] Updating timeout interval: 2364
[2020-12-3 12:52:55] Received ACK (ackNum: 7)
[2020-12-3 12:52:55] Updating timeout interval: 2163
[2020-12-3 12:52:55] All ACKs received
```

Figure 1: stdout of transmitter showing a nominal test run of a file transfer

- Transmitter sends a number of DATA packets specified by window size.
- Receiver sends ACKs for each DATA packet.
- Update timeout interval with each ACK received.
- For each ACK received, increase window size by one (reflected on next window).
- Once all packets are received, send new window of packets

```

nominal-transmitter.log X
1  [INFO] [2020-12-3 19:12:52] Host found: 192.168.1.78
2  [INFO] [2020-12-3 19:12:52] The network emulator's port is: 50001
3  [INFO] [2020-12-3 19:12:52] Sending data in file path: ./resource/message.txt
4  [INFO] [2020-12-3 19:12:52] Number of lines in the file are: 150
5  [INFO] [2020-12-3 19:12:52] Current window size: 1
6  [INFO] [2020-12-3 19:12:52] Sent DATA (seqNum: 1)
7  {
8      packetType: DATA,
9      seqNum: 1,
10     data: BEGIN
11 ,
12     windowSize: 1,
13     ackNum: 0,
14     retransmit: false,
15 }
16 [INFO] [2020-12-3 19:12:52] Window of packets sent, waiting for ACKs
17 [INFO] [2020-12-3 19:12:52] Received ACK (ackNum: 1)
18 {
19     packetType: ACK,
20     seqNum: 0,
21     data: (null),
22     windowSize: 1,
23     ackNum: 1,
24     retransmit: false,
25 }
26 [INFO] [2020-12-3 19:12:52] Updating timeout interval: 2450
27 [INFO] [2020-12-3 19:12:52] All ACKs received
28
29 [INFO] [2020-12-3 19:12:52] Current window size: 2
30 [INFO] [2020-12-3 19:12:52] Sent DATA (seqNum: 2)
31 {
32     packetType: DATA,
33     seqNum: 2,
34     data: Final Project:
35 ,
36     windowSize: 2,
37     ackNum: 0,
38     retransmit: false,
39 }
40 [INFO] [2020-12-3 19:12:52] Sent DATA (seqNum: 3)
41 {
42     packetType: DATA,
43     seqNum: 3,
44     data: Due: December 2, 2020 at 1000 hrs. Late submissions will not be accepted.
45 ,
46     windowSize: 2,
47     ackNum: 0,
48     retransmit: false,
49 }
50 [INFO] [2020-12-3 19:12:52] Window of packets sent, waiting for ACKs
51 [INFO] [2020-12-3 19:12:52] Received ACK (ackNum: 2)
52 {
53     packetType: ACK,
54     seqNum: 0,
55     data: (null),
56     windowSize: 2,
57     ackNum: 2,
58     retransmit: false,
59 }
60 [INFO] [2020-12-3 19:12:52] Updating timeout interval: 2695
61 [INFO] [2020-12-3 19:12:52] Received ACK (ackNum: 3)
62 {
63     packetType: ACK,
64     seqNum: 0,
65     data: (null),
66     windowSize: 2,
67     ackNum: 3,
68     retransmit: false,
69 }
70 [INFO] [2020-12-3 19:12:52] Updating timeout interval: 2763
71 [INFO] [2020-12-3 19:12:52] All ACKs received
72
73 [INFO] [2020-12-3 19:12:52] Current window size: 4

```

Figure 2: Log file of transmitter showing nominal test run of a file transfer, with individual packet information

1	Relative Time	Window Size	Packet Type	Retransmit	Seq #	Ack #	Source IP	Destination IP	Source Port	Destination Port
2	0:01:298	1	DATA	No	1	0	192.168.1.78	192.168.1.72	7001	7003
3	0:01:321	1	ACK	No	0	1	192.168.1.72	192.168.1.78	7003	7001
4	0:01:333	2	DATA	No	2	0	192.168.1.78	192.168.1.72	7001	7003
5	0:01:347	2	DATA	No	3	0	192.168.1.78	192.168.1.72	7001	7003
6	0:01:360	2	ACK	No	0	2	192.168.1.72	192.168.1.78	7003	7001
7	0:01:374	2	ACK	No	0	3	192.168.1.72	192.168.1.78	7003	7001
8	0:01:387	4	DATA	No	4	0	192.168.1.78	192.168.1.72	7001	7003
9	0:01:401	4	DATA	No	5	0	192.168.1.78	192.168.1.72	7001	7003
10	0:01:414	4	DATA	No	6	0	192.168.1.78	192.168.1.72	7001	7003
11	0:01:442	4	DATA	No	7	0	192.168.1.78	192.168.1.72	7001	7003
12	0:01:468	4	ACK	No	0	4	192.168.1.72	192.168.1.78	7003	7001
13	0:01:482	4	ACK	No	0	5	192.168.1.72	192.168.1.78	7003	7001
14	0:01:497	4	ACK	No	0	6	192.168.1.72	192.168.1.78	7003	7001
15	0:01:511	4	ACK	No	0	7	192.168.1.72	192.168.1.78	7003	7001

Figure 3: User interface of packet table showing UDP packet information captured to/from transmitter/receiver

nomial-transmitter.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
2	0.063924	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
3	0.065231	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
4	0.065544	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
5	0.096099	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
6	0.107079	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
7	0.108152	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
8	0.108410	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
9	0.108604	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
10	0.108794	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
11	0.187546	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
12	0.198230	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
13	0.209444	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
14	0.220979	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273

Figure 4: Wireshark capture of packets sent from transmitter to network emulator in a nominal test run

Test 1b: ARQ

Refer to high-BER-transmitter-network_emulator.mkv or high-delay-transmitter-network_emulator.mkv

```
outlog
3109 [INFO][2020-12-3 22:26:44] Current window size: 4
3110 [INFO][2020-12-3 22:26:44] Sent DATA (seqNum: 4)
3111 {
3112     packetType: DATA,
3113     seqNum: 4,
3114     data: (null),
3115     windowSize: 4,
3116     ackNum: 0,
3117     retransmit: false,
3118 }
3119 [INFO][2020-12-3 22:26:44] Sent DATA (seqNum: 5)
3120 {
3121     packetType: DATA,
3122     seqNum: 5,
3123     data: You may work in groups of two. ,
3124     windowSize: 4,
3125     ackNum: 0,
3126     retransmit: false,
3127 }
3128 [INFO][2020-12-3 22:26:44] Sent DATA (seqNum: 6)
3129 {
3130     packetType: DATA,
3131     seqNum: 6,
3132     data: (null),
3133     windowSize: 4,
3134     ackNum: 0,
3135     retransmit: false,
3136 }
3137 [INFO][2020-12-3 22:26:44] Sent DATA (seqNum: 7)
3138 {
3139     packetType: DATA,
3140     seqNum: 7,
3141     data: Objective: ,
3142     windowSize: 4,
3143     ackNum: 0,
3144     retransmit: false,
3145 }
3146 [INFO][2020-12-3 22:26:44] Window of packets sent, waiting for ACKs
3147 [INFO][2020-12-3 22:26:45] Received ACK (ackNum: 4)
3148 {
3149     packetType: ACK,
3150     seqNum: 0,
3151     data: (null),
3152     windowSize: 4,
3153     ackNum: 4,
3154     retransmit: false,
3155 }
3156 [INFO][2020-12-3 22:26:45] Updating timeout interval: 2701
3157 [INFO][2020-12-3 22:26:45] Received ACK (ackNum: 5)
3158 {
3159     packetType: ACK,
3160     seqNum: 0,
3161     data: (null),
3162     windowSize: 4,
3163     ackNum: 5,
3164     retransmit: false,
3165 }
3166 [INFO][2020-12-3 22:26:45] Updating timeout interval: 2571
3167 [INFO][2020-12-3 22:26:45] Received ACK (ackNum: 6)
3168 {
3169     packetType: ACK,
3170     seqNum: 0,
3171     data: (null),
3172     windowSize: 4,
3173     ackNum: 6,
3174     retransmit: false,
3175 }
3176 [INFO][2020-12-3 22:26:45] Updating timeout interval: 2398
3177 [INFO][2020-12-3 22:26:47] RTT (2398) >= Timeout Interval (=2398), packet loss event detected
3178 [INFO][2020-12-3 22:26:47] Retransmitting 1 unACKs...
3179 [INFO][2020-12-3 22:26:47] Updating timeout interval: 3833
3180 [INFO][2020-12-3 22:26:47] Received ACK (ackNum: 7)
3181 {
3182     packetType: ACK,
3183     seqNum: 0,
3184     data: (null),
3185     windowSize: 4,
3186     ackNum: 7,
3187     retransmit: false,
3188 }
3189 [INFO][2020-12-3 22:26:47] Updating timeout interval: 3574
3190 [INFO][2020-12-3 22:26:47] All ACKs received
3191
3192 [INFO][2020-12-3 22:26:47] Current window size: 4
```

Figure 5: Log file of transmitter showing packet loss test run of a file transfer, after packet with seqNum 7 gets dropped, the transmitter resends that packet, reduces the window size by half ($7/2 = 3$) and receives the ACK; Finally updating the next window size to 4

- Transmitter sends a number of DATA packets specified by window size.
- Receiver sends ACKs for each DATA packet.
- Update timeout interval with each ACK received.
- While waiting for ACKs, if the timer recording the round trip time exceeds the timeout interval, retransmit all packets that have not been ACKed.
- Every packet loss event will reduce the window size by half.

8	0:01:007	4	DATA	No	4	0	192.168.1.72	192.168.1.69	50000	50002
9	0:01:018	4	DATA	No	5	0	192.168.1.72	192.168.1.69	50000	50002
10	0:01:029	4	DATA	No	6	0	192.168.1.72	192.168.1.69	50000	50002
11	0:01:047	4	DATA	No	7	0	192.168.1.72	192.168.1.69	50000	50002
12	0:01:064	4	ACK	No	0	4	192.168.1.69	192.168.1.72	50002	50000
13	0:01:075	4	ACK	No	0	5	192.168.1.69	192.168.1.72	50002	50000
14	0:01:086	4	ACK	No	0	6	192.168.1.69	192.168.1.72	50002	50000
15	0:01:098	4	ACK (DROPPED)	No	0	7	192.168.1.69	192.168.1.72	50002	50000
16	0:03:411	4	DATA	Yes	7	0	192.168.1.72	192.168.1.69	50000	50002
17	0:03:424	4	ACK	No	0	7	192.168.1.69	192.168.1.72	50002	50000
18	0:03:462	4	DATA	No	8	0	192.168.1.72	192.168.1.69	50000	50002
19	0:03:480	4	DATA (DROPPED)	No	9	0	192.168.1.72	192.168.1.69	50000	50002
20	0:03:509	4	DATA	No	10	0	192.168.1.72	192.168.1.69	50000	50002
21	0:03:528	4	DATA	No	11	0	192.168.1.72	192.168.1.69	50000	50002

Figure 6: User interface of packet table showing UDP packet information captured to/from transmitter/receiver

No.	Time	Source	Destination	Protocol	Length	Info
37	1.290234	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
38	1.295403	192.168.1.78	192.168.1.69	UDP	315	50001 → 50002 Len=273
41	1.307580	192.168.1.69	192.168.1.78	UDP	315	50002 → 50001 Len=273
42	1.313370	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
43	1.319862	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
44	1.320006	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
45	1.332456	192.168.1.78	192.168.1.69	UDP	315	50001 → 50002 Len=273
46	1.335131	192.168.1.69	192.168.1.78	UDP	315	50002 → 50001 Len=273
47	1.344210	192.168.1.78	192.168.1.69	UDP	315	50001 → 50002 Len=273
48	1.349927	192.168.1.69	192.168.1.78	UDP	315	50002 → 50001 Len=273
49	1.355582	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
50	1.367871	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
51	1.371965	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
52	1.371965	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
53	1.372814	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
54	1.372814	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
55	1.378679	192.168.1.78	192.168.1.69	UDP	315	50001 → 50002 Len=273
56	1.380875	192.168.1.69	192.168.1.78	UDP	315	50002 → 50001 Len=273
57	1.389698	192.168.1.78	192.168.1.69	UDP	315	50001 → 50002 Len=273
58	1.391698	192.168.1.69	192.168.1.78	UDP	315	50002 → 50001 Len=273
59	1.406863	192.168.1.78	192.168.1.69	UDP	315	50001 → 50002 Len=273
60	1.408875	192.168.1.69	192.168.1.78	UDP	315	50002 → 50001 Len=273
61	1.424371	192.168.1.78	192.168.1.69	UDP	315	50001 → 50002 Len=273
62	1.428049	192.168.1.69	192.168.1.78	UDP	315	50002 → 50001 Len=273
63	1.435798	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
64	1.446839	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
65	1.458102	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
157	3.777414	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
158	3.783585	192.168.1.78	192.168.1.69	UDP	315	50001 → 50002 Len=273
159	3.785318	192.168.1.69	192.168.1.78	UDP	315	50002 → 50001 Len=273
160	3.796721	192.168.1.78	192.168.1.72	UDP	315	50001 → 50000 Len=273
161	3.804479	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
162	3.804479	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
163	3.804872	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
164	3.804872	192.168.1.72	192.168.1.78	UDP	315	50000 → 50001 Len=273
165	3.839355	192.168.1.78	192.168.1.69	UDP	315	50001 → 50002 Len=273
166	3.840967	192.168.1.69	192.168.1.78	UDP	315	50002 → 50001 Len=273

Figure 7: Wireshark capture of packets sent from transmitter to network emulator in with packet loss in test run

Test 1c: EOT

Refer to nominal-transmitter-network_emulator.mkv

```
[2020-12-3 22:35:7] Current window size: 20
[2020-12-3 22:35:7] Sent DATA (seqNum: 145)
[2020-12-3 22:35:7] Sent DATA (seqNum: 146)
[2020-12-3 22:35:7] Sent DATA (seqNum: 147)
[2020-12-3 22:35:7] Sent DATA (seqNum: 148)
[2020-12-3 22:35:7] Sent DATA (seqNum: 149)
[2020-12-3 22:35:7] Sent DATA (seqNum: 150)
[2020-12-3 22:35:7] Window of packets sent, waiting for ACKs
[2020-12-3 22:35:7] Received ACK (ackNum: 145)
[2020-12-3 22:35:7] Updating timeout interval: 1604
[2020-12-3 22:35:8] Received ACK (ackNum: 146)
[2020-12-3 22:35:8] Updating timeout interval: 1708
[2020-12-3 22:35:8] Received ACK (ackNum: 147)
[2020-12-3 22:35:8] Updating timeout interval: 1704
[2020-12-3 22:35:8] Received ACK (ackNum: 148)
[2020-12-3 22:35:8] Updating timeout interval: 1627
[2020-12-3 22:35:8] Received ACK (ackNum: 149)
[2020-12-3 22:35:8] Updating timeout interval: 1493
[2020-12-3 22:35:8] Received ACK (ackNum: 150)
[2020-12-3 22:35:8] Updating timeout interval: 1353
[2020-12-3 22:35:8] All ACKs received

[2020-12-3 22:35:8] Completed Data Transfer
[2020-12-3 22:35:8] Sending EOT Packet
[2020-12-3 22:35:8] Terminating Transmitter...
```

Figure 8: stdout of transmitter showing the end of a nominal test run of a file transfer, with transmitter sending the EOT packets and terminating

```

[INFO][2020-12-3 22:35:7] Sent DATA (seqNum: 150)
{
  packetType: DATA,
  seqNum: 150,
  data: END ,
  windowSize: 20,
  ackNum: 0,
  retransmit: false,
}
[INFO][2020-12-3 22:35:7] Window of packets sent, waiting for ACKs
[INFO][2020-12-3 22:35:7] Received ACK (ackNum: 145)
{
  packetType: ACK,
  seqNum: 0,
  data: (null),
  windowSize: 20,
  ackNum: 145,
  retransmit: false,
}
[INFO][2020-12-3 22:35:7] Updating timeout interval: 1604
[INFO][2020-12-3 22:35:8] Received ACK (ackNum: 146)
{
  packetType: ACK,
  seqNum: 0,
  data: (null),
  windowSize: 20,
  ackNum: 146,
  retransmit: false,
}
[INFO][2020-12-3 22:35:8] Updating timeout interval: 1708
[INFO][2020-12-3 22:35:8] Received ACK (ackNum: 147)
{
  packetType: ACK,
  seqNum: 0,
  data: (null),
  windowSize: 20,
  ackNum: 147,
  retransmit: false,
}
[INFO][2020-12-3 22:35:8] Updating timeout interval: 1704
[INFO][2020-12-3 22:35:8] Received ACK (ackNum: 148)
{
  packetType: ACK,
  seqNum: 0,
  data: (null),
  windowSize: 20,
  ackNum: 148,
  retransmit: false,
}
[INFO][2020-12-3 22:35:8] Updating timeout interval: 1627
[INFO][2020-12-3 22:35:8] Received ACK (ackNum: 149)
{
  packetType: ACK,
  seqNum: 0,
  data: (null),
  windowSize: 20,
  ackNum: 149,
  retransmit: false,
}
[INFO][2020-12-3 22:35:8] Updating timeout interval: 1493
[INFO][2020-12-3 22:35:8] Received ACK (ackNum: 150)
{
  packetType: ACK,
  seqNum: 0,
  data: (null),
  windowSize: 20,
  ackNum: 150,
  retransmit: false,
}
[INFO][2020-12-3 22:35:8] Updating timeout interval: 1353
[INFO][2020-12-3 22:35:8] All ACKs received

[INFO][2020-12-3 22:35:8] Completed Data Transfer
[INFO][2020-12-3 22:35:8] Sending EOT Packet
[INFO][2020-12-3 22:35:8] Terminating Transmitter...

```

Figure 9: Log file of transmitting showing end of a nominal test with EOT packets being sent after all ACKs received

320	0:25:197	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
321	0:25:242	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
322	0:25:285	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
323	0:25:330	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
324	0:25:375	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
325	0:25:421	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
326	0:25:465	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
327	0:25:509	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
328	0:25:552	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
329	0:25:595	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002

Figure 10: Network Emulator capturing the EOT packets being sent from transmitter to receiver

308	9.454273	192.168.1.78	192.168.1.72	UDP	315 50001 → 50000 Len=273
309	9.472912	192.168.1.78	192.168.1.72	UDP	315 50001 → 50000 Len=273
310	9.474307	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273
311	9.474356	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273
312	9.474371	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273
313	9.474396	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273
314	9.474421	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273
315	9.474445	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273
316	9.474469	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273
317	9.474493	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273
318	9.474517	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273
319	9.474540	192.168.1.72	192.168.1.78	UDP	315 50000 → 50001 Len=273

Figure 11: Packet capture showing 10 EOT packets being sent from transmitter to receiver

Test 2a: ACK

Refer to nominal-receiver.mp4

Receiver responds with ACK for each received DATA packet. ACK packets acknowledgment number matches corresponding DATA packet received from the transmitter.

```
[maksymc@maksym-hpspectrex360convertible13aw0xxx receiver]$ ./build/receiver.out
[2020-12-3 21:0:4] received DATA (seqNum: 1)
[2020-12-3 21:0:4] sent ACK packet (ackNum: 1)
[2020-12-3 21:0:4] received DATA (seqNum: 2)
[2020-12-3 21:0:4] sent ACK packet (ackNum: 2)
```

Figure 12: receiver outputting application status to STDOUT.

```
receiver > logs > $ out.log
1 [INFO][2020-12-3 19:29:39] received DATA (seqNum: 1)
2 {
3   packetType: DATA,
4   seqNum: 1,
5   data: BEGIN
6   ,
7   windowSize: 1,
8   ackNum: 0,
9   retransmit: false,
10 }
11 [INFO][2020-12-3 19:29:39] sent ACK packet (ackNum: 1)
12 {
13   packetType: ACK,
14   seqNum: 0,
15   data: (null),
16   windowSize: 1,
17   ackNum: 1,
18   retransmit: false,
19 }
20 [INFO][2020-12-3 19:29:39] received DATA (seqNum: 2)
21 {
22   packetType: DATA,
23   seqNum: 2,
24   data: Final Project:
25   ,
26   windowSize: 2,
27   ackNum: 0,
28   retransmit: false,
29 }
30 [INFO][2020-12-3 19:29:39] sent ACK packet (ackNum: 2)
31 {
32   packetType: ACK,
33   seqNum: 0,
34   data: (null),
35   windowSize: 2,
36   ackNum: 2,
37   retransmit: false,
38 }
```

Figure 13: receiver logging application status and packet details to a file.

1	Relative Time	Window Size	Packet Type	Retransmit	Seq #	Ack #	Source IP	Destination IP	Source Port	Destination Port
2	0:01:315	1	DATA	No	1	0	192.168.1.72	192.168.1.69	50000	50002
3	0:01:353	1	ACK (DROPPED)	No	0	1	192.168.1.69	192.168.1.72	50002	50000
4	0:03:315	1	DATA	Yes	1	0	192.168.1.72	192.168.1.69	50000	50002
5	0:03:352	1	ACK	No	0	1	192.168.1.69	192.168.1.72	50002	50000

Figure 14: DATA and ACK packets are displayed in the Network Emulator.

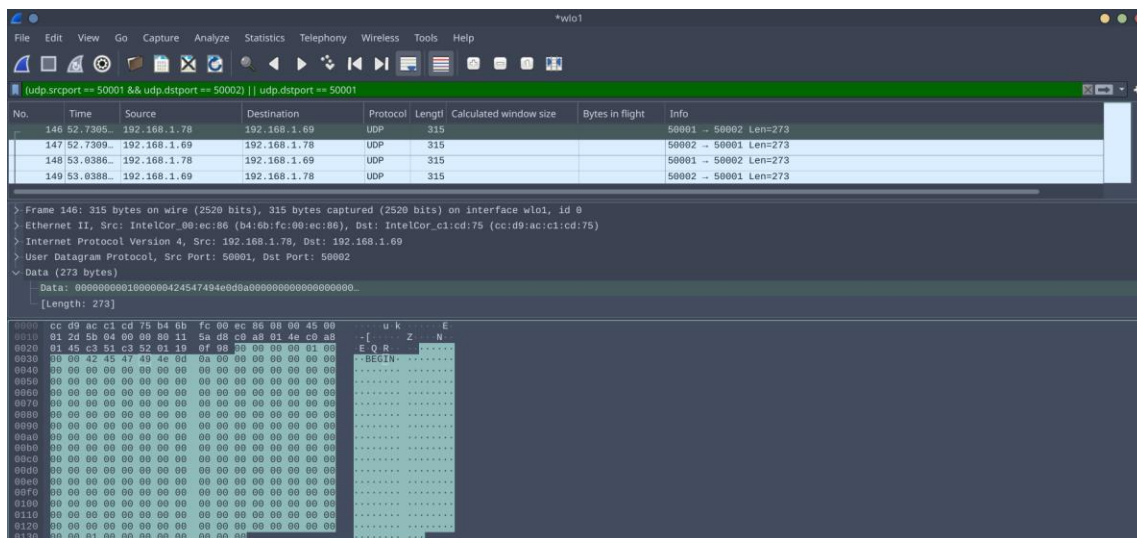


Figure 15: DATA and ACK packet exchange between the receiver(192.168.1.69) and the Network Emulator(192.168.1.77).

Test 2b: In/out of order data handling

Refer to 2b.mp4 showcasing data being written to the file over the transmission period (in order and out of order packet handling).

```
root@localhost:/home/radiant/7005/finalProject/PacketLens7005/transmitt...
[root@localhost resource]# pwd
/home/radiant/7005/finalProject/PacketLens7005/transmitter/resource
[root@localhost resource]# ls
message.txt
[root@localhost resource]# cat message.txt | sha256sum
625c4957dc29572452cd1f1dc058d2f62bc403ef87f72237904dcc873012ef31 -
[root@localhost resource]#
```

Figure 16: checksum of the message sent by the transmitter.

```
receiver : bash — Konsole <3>
File Edit View Bookmarks Settings Help
[maksymc@maksym-hpspectrex360convertible13aw0xxx receiver]$ pwd
/home/maksymc/Desktop/COMP-7005-Final-Project/receiver
[maksymc@maksym-hpspectrex360convertible13aw0xxx receiver]$ cat ./data/message.txt | sha256sum
625c4957dc29572452cd1f1dc058d2f62bc403ef87f72237904dcc873012ef31 -
[maksymc@maksym-hpspectrex360convertible13aw0xxx receiver]$
```

Figure 17: checksum of the message received by the receiver confirming protocol's reliability.

Test 2c: EOT

Refer to nominal-receiver.mp4

```
[2020-12-3 22:26:21] received EOT packet  
[2020-12-3 22:26:21] terminating receiver...
```

Figure 18: receiver outputting application status after receiving EOT packet.

```
[INFO][2020-12-3 22:26:21] received EOT packet  
{  
  packetType: EOT,  
  seqNum: 0,  
  data: (null),  
  windowSize: 0,  
  ackNum: 0,  
  retransmit: false,  
}  
[INFO][2020-12-3 22:26:21] terminating receiver...  
{  
  packetType: EOT,  
  seqNum: 0,  
  data: (null),  
  windowSize: 0,  
  ackNum: 0,  
  retransmit: false,  
}
```

Figure 19: receiver logging EOT packets details and application status.

No.	Time	Source	Destination	Protocol	Length	Calculated window size	Bytes in flight	Info
634	92.6691	192.168.1.69	192.168.1.78	UDP	315			50002 → 50001 Len=273
635	92.7264	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
636	92.7268	192.168.1.69	192.168.1.78	UDP	315			50002 → 50001 Len=273
637	92.7645	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
638	92.7833	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
639	92.7834	192.168.1.69	192.168.1.78	ICMP	343			Destination unreachable (Port unreachable)
640	92.8019	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
641	92.8019	192.168.1.69	192.168.1.78	ICMP	343			Destination unreachable (Port unreachable)
642	92.8220	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
643	92.8220	192.168.1.69	192.168.1.78	ICMP	343			Destination unreachable (Port unreachable)
644	92.8403	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
645	92.8404	192.168.1.69	192.168.1.78	ICMP	343			Destination unreachable (Port unreachable)
646	92.8580	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
647	92.8580	192.168.1.69	192.168.1.78	ICMP	343			Destination unreachable (Port unreachable)
648	92.8794	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
649	92.8794	192.168.1.69	192.168.1.78	ICMP	343			Destination unreachable (Port unreachable)
650	92.8972	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
651	92.9165	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273
652	92.9346	192.168.1.78	192.168.1.69	UDP	315			50001 → 50002 Len=273

Figure 20: last packets in the transmission.

320	0:25:197	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
321	0:25:242	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
322	0:25:285	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
323	0:25:330	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
324	0:25:375	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
325	0:25:421	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
326	0:25:465	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
327	0:25:509	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
328	0:25:552	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002
329	0:25:595	0	EOT	No	0	0	192.168.1.72	192.168.1.69	50000	50002

Figure 21: 10 EOT packets sent from the transmitter to the receiver to indicate the end of transmission.

Test 3a: App Start

Refer to 3a-g.mkv

```
/*----- Symbolic Constants -----*/
#define NETWORK_EMULATOR_PORT    50001
#define TRANSMITTER_PORT          50000
#define RECEIVER_PORT              50002
#define PAYLOAD_LEN                256
#define INITIAL_WINDOW_SIZE        1
#define MAX_WINDOW_SIZE            20
#define INITIAL_SEQ_NUM            1

/*----- Default Strings -----*/
#define TRANSMITTER_IP              "192.168.1.72"
#define NETWORK_EMULATOR_IP        "192.168.1.78"
#define RECEIVER_IP                  "192.168.1.69"
```

Figure 22: common.h specifying default values for endpoint details, payload length and window size

Settings	
Transmitter IP	192.168.1.72
Transmitter Port	50000
Receiver IP	192.168.1.69
Receiver Port	50002
Network Emulator IP	192.168.1.78
Network Emulator Port	50001
Payload Length	256
Max Window Size	20

Figure 23: Settings table is populated with default values specified in the configuration file

```
/*----- Symbolic Constants -----*/
#define NETWORK_DELAY_MS           30
#define MIN_NETWORK_DELAY_MS        5
#define MAX_NETWORK_DELAY_MS       100
#define ERROR_RATE_PERCENT          5
#define MIN_ERROR_RATE_PERCENT      4
#define MAX_ERROR_RATE_PERCENT     100
#define INITIAL_MAX_X                3
#define INITIAL_MAX_Y                5
```

Figure 24: networkemulator.h specifying default values for packet drop parameters

Active

Start

Stop

Save

Reset

Packet Delay (ms): 30

Bit Error Rate: 5%

Figure 25: On app start, the default values for BER and packet delay are set in the sliders

Test 3b: Start Button

Refer to 3a-g.mkv

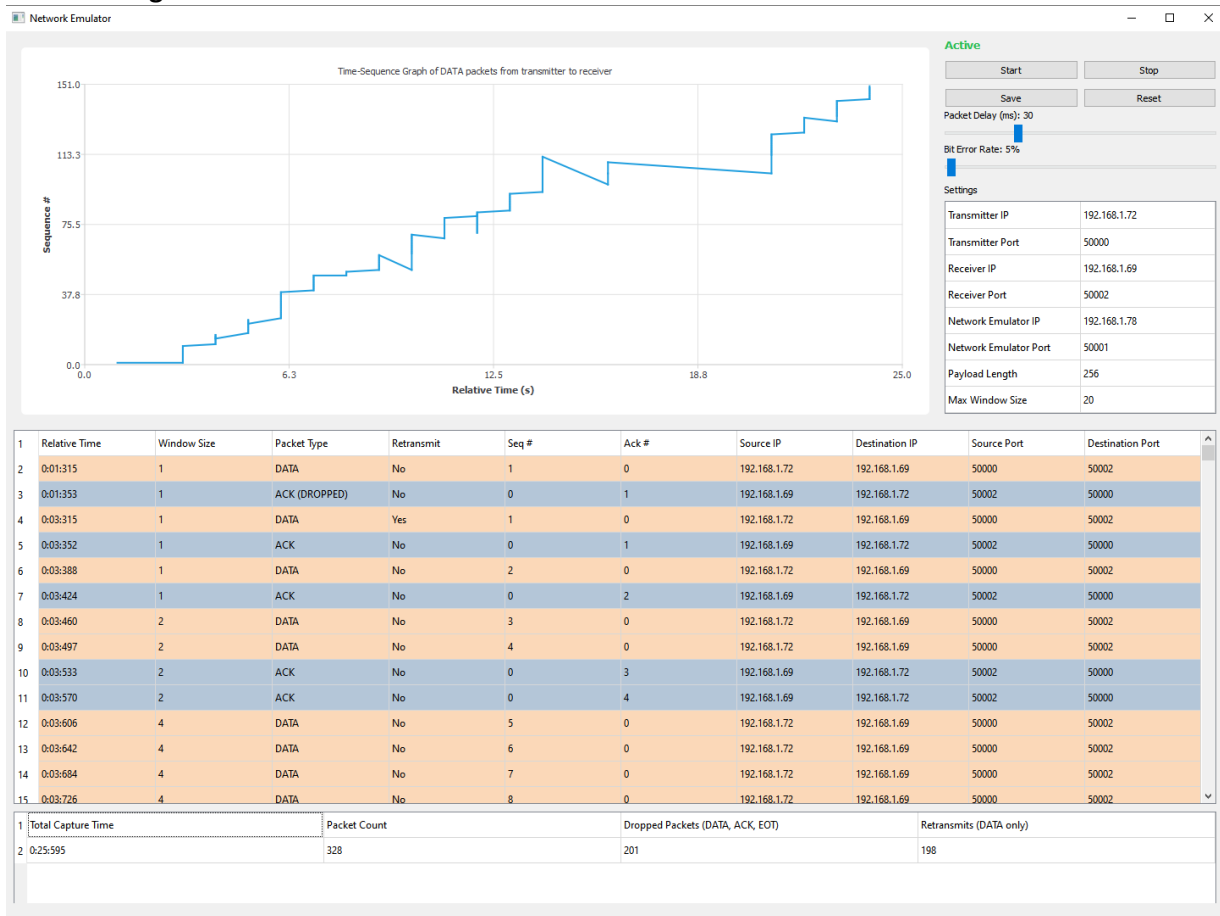


Figure 25: Pressing start allows for packets to be relayed to/from transmitter/receiver. For each packet that is received on the transmitter, the packet capture table will log its details, update the network summary table. Only packets sent from the transmitter to the network emulator are graphed on the Time Sequence diagram.

Test 3c: Stop Button

Refer to 3a-g.mkv

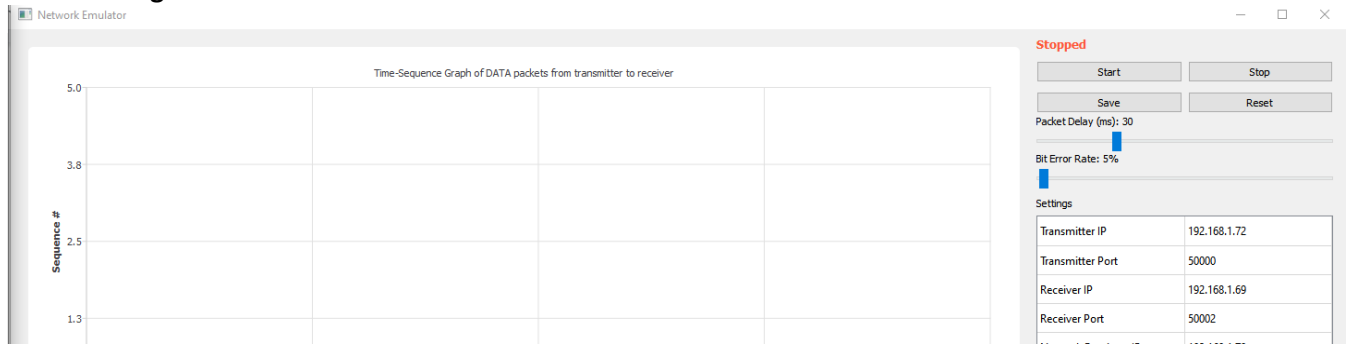


Figure 25: Pressing stop prevents packets from being relayed. No updates to the graphs are reflected.

```
radiant@localhost:~/7005/finalProject/PacketLens7005/transmitter
[radiant@localhost transmitter]$ ./build/transmitter.o
[2020-12-3 23:17:53] Host found: 192.168.1.78
[2020-12-3 23:17:53] The network emulator's port is: 50001
[2020-12-3 23:17:53] Sending data in file path: ./resource/message.txt
[2020-12-3 23:17:53] Number of lines in the file are: 150
[2020-12-3 23:17:53] Current window size: 1
[2020-12-3 23:17:53] Sent DATA (seqNum: 1)
[2020-12-3 23:17:53] Window of packets sent, waiting for ACKs
[2020-12-3 23:17:55] RTT (2000) >= Timeout Interval (=2000), packet loss event detected
[2020-12-3 23:17:55] Retransmitting 1 unACKs...
[2020-12-3 23:17:55] Updating timeout interval: 2749
[2020-12-3 23:17:58] RTT (2750) >= Timeout Interval (=2749), packet loss event detected
[2020-12-3 23:17:58] Retransmitting 1 unACKs...
[2020-12-3 23:17:58] Updating timeout interval: 3968
[2020-12-3 23:18:2] RTT (3969) >= Timeout Interval (=3968), packet loss event detected
[2020-12-3 23:18:2] Retransmitting 1 unACKs...
[2020-12-3 23:18:2] Updating timeout interval: 5000
[2020-12-3 23:18:7] RTT (5000) >= Timeout Interval (=5000), packet loss event detected
[2020-12-3 23:18:7] Retransmitting 1 unACKs...
[2020-12-3 23:18:7] Updating timeout interval: 5000
```

Figure 26: Transmitter stdout showing that packets aren't able to transmit to the network emulator

Test 3d: Reset Button

Refer to 3a-g.mkv

Test 3e: Save Button

Refer to 3a-g.mkv

/	192.168.1.192.168.1.	50002	50000	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
8	192.168.1.192.168.1.	50002	50000	
9	192.168.1.192.168.1.	50002	50000	
10	192.168.1.192.168.1.	50002	50000	
12	192.168.1.192.168.1.	50002	50000	
13	192.168.1.192.168.1.	50002	50000	
14	192.168.1.192.168.1.	50002	50000	
15	192.168.1.192.168.1.	50002	50000	
0	192.168.1.192.168.1.	50000	50002	
11	192.168.1.192.168.1.	50002	50000	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
16	192.168.1.192.168.1.	50002	50000	
17	192.168.1.192.168.1.	50002	50000	
18	192.168.1.192.168.1.	50002	50000	
19	192.168.1.192.168.1.	50002	50000	
20	192.168.1.192.168.1.	50002	50000	
21	192.168.1.192.168.1.	50002	50000	
22	192.168.1.192.168.1.	50002	50000	
23	192.168.1.192.168.1.	50002	50000	
0	192.168.1.192.168.1.	50000	50002	
17	192.168.1.192.168.1.	50002	50000	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	
0	192.168.1.192.168.1.	50000	50002	

Figure 27: Saved csv file containing the packet capture table contents

Test 3f: Packet Delay

Refer to 3a-g.mkv

Test 3g: Bit Error Rate

Refer to 3a-g.mkv

Test 3a-g: Packet Capture (refer to 3a-g.pcap)

Wireshark - 3a-g.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter: <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
2	0.030421	192.168.1.78	192.168.1.78	UDP	305	50001 → 50002 Len=273
3	0.036140	192.168.1.78	192.168.1.78	UDP	305	50002 → 50001 Len=273
4	0.070327	192.168.1.78	192.168.1.78	UDP	305	50001 → 50000 Len=273
5	0.074855	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
6	0.075838	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
7	0.105756	192.168.1.78	192.168.1.78	UDP	305	50001 → 50002 Len=273
8	0.111791	192.168.1.78	192.168.1.78	UDP	305	50002 → 50001 Len=273
9	0.140930	192.168.1.78	192.168.1.78	UDP	305	50001 → 50002 Len=273
10	0.147581	192.168.1.78	192.168.1.78	UDP	305	50002 → 50001 Len=273
11	0.176828	192.168.1.78	192.168.1.78	UDP	305	50001 → 50000 Len=273
12	0.212549	192.168.1.78	192.168.1.78	UDP	305	50001 → 50000 Len=273
13	0.216825	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
14	0.217564	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
15	0.218225	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
16	0.218879	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
17	0.247749	192.168.1.78	192.168.1.78	UDP	305	50001 → 50002 Len=273
18	0.252652	192.168.1.78	192.168.1.78	UDP	305	50002 → 50001 Len=273
19	0.283549	192.168.1.78	192.168.1.78	UDP	305	50001 → 50002 Len=273
20	0.288659	192.168.1.78	192.168.1.78	UDP	305	50002 → 50001 Len=273
21	0.325284	192.168.1.78	192.168.1.78	UDP	305	50001 → 50002 Len=273
22	0.330362	192.168.1.78	192.168.1.78	UDP	305	50002 → 50001 Len=273
23	0.365756	192.168.1.78	192.168.1.78	UDP	305	50001 → 50002 Len=273
24	0.370749	192.168.1.78	192.168.1.78	UDP	305	50002 → 50001 Len=273
25	0.437253	192.168.1.78	192.168.1.78	UDP	305	50001 → 50000 Len=273
26	0.474409	192.168.1.78	192.168.1.78	UDP	305	50001 → 50000 Len=273
27	0.511127	192.168.1.78	192.168.1.78	UDP	305	50001 → 50000 Len=273
28	2.356991	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
29	2.388434	192.168.1.78	192.168.1.78	UDP	305	50001 → 50002 Len=273
30	2.389660	192.168.1.78	192.168.1.78	UDP	305	50002 → 50001 Len=273
31	2.425028	192.168.1.78	192.168.1.78	UDP	305	50001 → 50000 Len=273
32	2.429367	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
33	2.430176	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273
34	2.430851	192.168.1.78	192.168.1.78	UDP	305	50000 → 50001 Len=273

Wireshark - Conversations - 3a-g.pcap

Ethernet		IPv4 · 1		IPv6		TCP		UDP · 2							
Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A		
192.168.1.78	50001	192.168.1.78	50000	988	301k	450	137k	538	164k	0.000000	237.0871	4631		5536	
192.168.1.78	50002	192.168.1.78	50001	979	298k	475	144k	504	153k	0.030421	237.2428	4885		5183	

☐ Name resolution ☐ Limit to display filter ☐ Absolute start time

Copy Follow Stream... Graph... Close Help

Packets: 1967 · Displayed: 1967 (100.0%) Profile: Default