REPORT – Networks Lab Assignment 2

The parse script named parse.py is attached along with this document.

The parsing strategy-

Every packet is assumed to be a type of Packet Object. This packet object has various attributes whose values include the details obtained from Udp-echo.tr trace file about both way transmission. We have maintained two set of information for each packet object. The first set of information deals with the client to server interaction, on the other hand, the second set of information contains the values related to the same packet when it is returned from server back to the client. It should be noted that when obtaining the statistical parameters using our script, we took special care that both of these set of information are kept separate when calculating transferred/received packets/bytes and throughput obtained.

The following observations were taken when we increased the packet size keeping the time interval of sending the packets at 0.01 s with total time of transmission 10s

1) Irrespective of the size of packet, the first packet is queued at the same time after adding appropriate headers. On the other hand last packet queuing time increases with packet size very minutely.

2) Time of retrieval of first packet as well as last packet increases with packet size which is due to the fixed data rate of the NIC.

3) The delay of transmission increases noticeably with packet size. As more time is taken while queuing and dequeuing the large size packet and transmission time of large packet is proportionally higher than that smaller one because of fixed data rate, the sum of all end-to-end delay time increases with packet size.

4) Surprisingly there was no packet loss or data loss in the whole network, which might be attributed to the way the ns-3 simulation, takes place. All the data packets were captured properly. The script has taken complete care of telling dropped or lost data packets in case of such happening but none of these drops of losses happened in our network.

5) One interesting thing to note was that with the increase in size of packets, the throughput also increased proportional to the amount of increase in packet size. This might be attributed to the fact that none of the data is lost and hence in the same time more data is being received by server/client.

The table given below tells us the statistically obtained values of packets for both server side and client side.

Packet Size → / Statistics	64 bytes	128 bytes	256 bytes	512 bytes	1024 bytes
TimeFirstTPacket (seconds)	2.012210	2.012210	2.012210	2.012210	2.012210
TimeLastTPacket (seconds)	9.992180	9.992280	9.992480	9.992890	9.993710
timeFirstRPacket (seconds)	2.014380	2.014490	2.014690	2.015100	2.015920
timeLastRPacket (seconds)	9.994350	9.994560	9.994970	9.995790	9.997420
delaySum (seconds)	3.493250	3.661440	3.995380	4.662620	5.985010
tBytes (bytes)	Client: 51200 Server: 51200	Client: 102400 Server: 102400	Client: 204800 Server: 204800	Client: 409600 Server:409600	Client:819200 Server:819200
tPackets	Client: 800 Server: 800	Client: 800 Server: 800	Client: 800 Server: 800	Client: 800 Server: 800	Client: 800 Server: 800
rBytes (bytes)	Client: 51200 Server: 51200	Client: 102400 Server: 102400	Client: 204800 Server: 204800	Client: 409600 Server:409600	Client:819200 Server:819200
rPackets	Client: 800 Server: 800	Client: 800 Server: 800	Client: 800 Server: 800	Client: 800 Server: 800	Client: 800 Server: 800
lostPackets	0	0	0	0	0
timesForwarded	No Packets Forwarded	No Packets Forwarded	No Packets Forwarded	No Packets Forwarded	No Packets Forwarded
packetsDropped	0	0	0	0	0
bytesDropped	0	0	0	0	0
transmitterThroughput (bytes/second)	Client: 6417.817466 Server: 6423.598696	Client: 12835.634931 Server: 12847.197391	Client: 25671.269863 Server: 25707.844407	Client: 51342.539726 Server: 51418.980584	Client: 102685.079452 Server: 102846.353173
receiverThroughput (bytes/second)	Client: 6423.606755 Server: 6417.809421	Client: 12847.197391 Server: 12835.634931	Client: 25707.844407 Server: 25671.269863	Client: 51418.980584 Server: 51342.539726	Client: 102846.353173 Server: 102685.079452

interPacketInterval = 0.01 seconds

The following observations were taken when the packet size was kept fixed at 128 bytes and the time intervals were increased for transmission of packets keeping the total time of transmission fixed at 10s

1) As expected, there was no change in first packet transmission.

2) Due to the fixed time interval for transmission of packets and also the total time of transmission to be fixed, the network automatically adjusted total number of packets to transmit and hence we transmitted the last packet at approx (10-0.02)s for 0.02s interval and then went up at approx (10-1)s for last packet transmission.

3) The same logic as above can be applied for reception of packet at the server side. The statistical support of the table indeed proves this assertion.

4) As small number of packets are transferred with increasing time interval, the delay sum decreases as the number of packets involved in addition are becoming less and less.

5) Note that there is still no data or packet loss in the case of our NS-3 simulated network.

6) One thing to note is that transmitted bytes are decreasing because less number of 128 bytes packets are being transferred due to increase in time interval of transmission.

7) The proportional decrease in throughput can be mathematically obtained using the number of bytes transferred as defining factor.

The following table tells the statistics obtained when increasing the time interval and keeping the packet size same at 128 bytes and keeping the total time of transmission at 10s.

InterPacketInterval → Statistics	0.02 seconds	0.05 seconds	0.1 seconds	1 second
timeFirstTPacket (seconds)	2.012210	2.012210	2.012210	2.012210
timeLastTPacket (seconds)	9.982280	9.952280	9.902280	9.002280
timeFirstRPacket (seconds)	2.014490	2.014490	2.014490	2.014490
timeLastRPacket (seconds)	9.984560	9.954560	9.904560	9.004560
delaySum (seconds)	1.826250	0.729600	0.364800	0.036480
tBytes (bytes)	Client: 51200 Server: 51200	Client: 20480 Server: 20480	Client: 10240 Server: 10240	Client: 1024 Server: 1024
tPackets	Client: 400 Server: 400	Client: 160 Server: 160	Client: 80 Server: 80	Client: 8 Server: 8
rBytes (bytes)	Client: 51200 Server: 51200	Client: 20480 Server: 20480	Client: 10240 Server: 10240	Client: 1024 Server: 1024
rPackets	Client: 400 Server: 400	Client: 160 Server: 160	Client: 80 Server: 80	Client: 8 Server: 8
lostPackets	0	0	0	0
timesForwarded	No Packets Forwarded	No Packets Forwarded	No Packets Forwarded	No Packets Forwarded
packetsDropped	0	0	0	0
bytesDropped	0	0	0	0
transmitterThroughput (bytes/second)	Client: 6425.872168 Server: 6430.068608	Client: 2580.063217 Server: 2581.754509	Client: 1298.209004 Server: 1299.065409	Client: 146.541324 Server: 146.650455
receiverThroughput (bytes/second)	Client: 6430.068608 Server: 6425.872168	Client: 2581.754509 Server: 2580.063217	Client: 1299.065409 Server: 1298.209004	Client: 146.650455 Server: 146.541324

packetSize = 128 bytes