

COMMUNICATION NETWORKS

ECE 634 PROJECT 2

SAAD SADIQ - C11954772



Table of Contents

1. Introduction	3
1.1 Transport Control Protocols	3
1.1.1 "Stop-and-Wait" protocol	4
1.1.2 Window based Go Back N and Selective Repeat	4
2. Implementation Details	6
2.1 Creating a dummy file	6
2.2 Emulating Unreliable Network	6
2.2.1 Creating a Secret Pass key	6
2.2.2 Time out & Window size	6
2.3 Simulation 1.	7
2.3.1 Stop-N-Wait - Sim1	7
2.3.2 Go-Back-N - Sim1	8
2.3.4 Selective Repeat - Sim1	9
2.4 Simulation 2.	10
2.4.1 Stop N Wait - Sim2	10
2.4.2 Go Back N - Sim2	11
2.4.3 Selective Repeat - Sim2	12
2.5 Simulation 3	13
2.5.1 Stop N Wait - Sim3	13
2.5.2 Go Back N - Sim3	14
2.5.3 Selective Repeat - Sim3	15
2.6 Simulation 4.	16
2.6.1 Stop N Wait - Sim4	16
2.6.2 Go-Back-N - Sim 4	17
2.6.3 Selective Repeat - Sim 4	18
2.7 Ending the simulation	18
3. Comparing Performances of the three Protocols	19
Stop & Wait	19
Go-Back-N vs Sliding Window	19
4. Conclusion	19

1. Introduction

In this project the objective was to design a simple transport protocol that provides reliable datagram service. The protocol should be able to ensure that data is delivered in order and without duplicates. The developed protocol was tested on an emulated unreliable network. Our project consists of two programs: a sending program that sends a file across the network, and a receiving program that receives the file and stores it back to the local disk. The project code was written in C on a Linux operating system. No transport protocol libraries were used in the project. The packets were constructed and acknowledgements by the sender and receiver programs by interpreting the incoming packets.

The following objectives were achieved in the project:

1. The sender reads a local file and uses the developed protocol to send it.
2. The file contents are transferred reliably.
3. The receiver writes the contents it receives into the local directory.
4. The program transfers small as well as large file sizes.
5. A custom designed packet uses UDP as a carrier to transmit packets.
6. The packet includes fields for packet type, ack number, data and retransmission requests.
7. A retransmission protocol was developed to deal with dropped, duplicated, and delayed packets.
8. Three popular protocols were implemented.
9. The designed protocol is fast such that it requires comparatively little time to transfer the file.
10. The requires low overhead data volume to be exchanged over the network to conduct the transfer.

1.1 Transport Control Protocols

The following Transport Control Protocols were implemented in this project

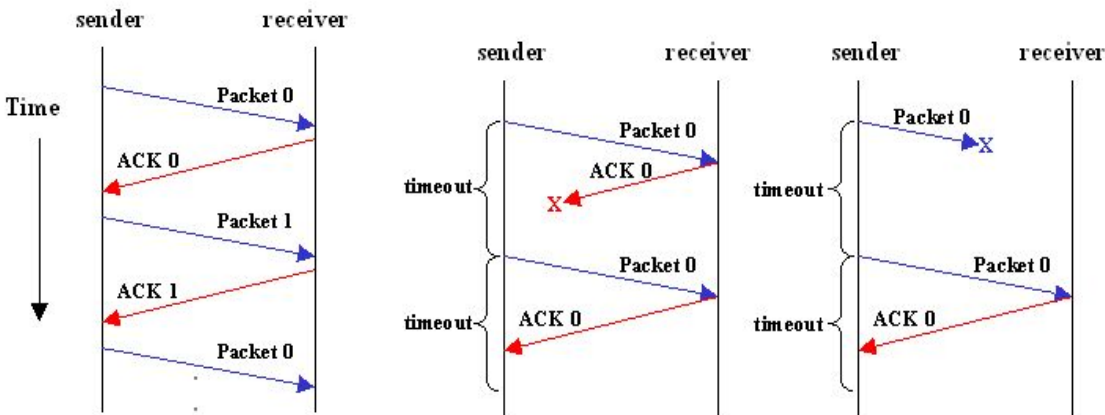
1. Stop N Wait
2. Go Back N
3. Selective Repeat

The short descriptions of each protocols as it was observed and implemented in the project is provided below.

1.1.1 "Stop-and-Wait" protocol

"stop-n-wait" (sometimes known as "positive acknowledgement with retransmission") is the fundamental technique to provide reliable transfer under unreliable packet delivery system.

2. How this protocol works..



After transmitting one packet, the sender waits for an acknowledgment (ACK) from the receiver before transmitting the next one. In this way, the sender can recognize that the previous packet is transmitted successfully and we could say "stop-n-wait" guarantees reliable transfer between nodes.

To support this feature, the sender keeps a record of each packet it sends.

Also, to avoid confusion caused by delayed or duplicated ACKs, "stop-n-wait" sends each packets with unique sequence numbers and receives that numbers in each ACKs

If the sender doesn't receive ACK for previous sent packet after a certain period of time, the sender times out and retransmits that packet again. There are two cases when the sender doesn't receive ACK; One is when the ACK is lost and the other is when the frame itself is not transmitted.

To support this feature, the sender keeps timer per each packet

1.1.2 Window based Go Back N and Selective Repeat

In the Go-Back-N protocol the sending process continues to send the packets until the specified window size is full. Then the program waits for the ACKs or the acknowledgements of the first packet in the window to return from the receiver. This is a special case of the window protocol where the sending size is complete window whereas the ACK receiving side is only 1 packet large. If the ACKs do not reach the sender until a set timer expires, the whole window is sent again. This process is repeated if the timer keeps expiring.

In comparison, the Selective Repeat protocol that first fills the specified window with the file packets and sends the packet of the entire window. This protocol keeps track of the received ACKs for each packet. As soon as the packets start receiving from the start of the window, the Selective Repeat protocol slides the window forward to add more packets. If a packet ACK is not received until the timeout expires, the protocol retransmits only that packet, since each packet sequence number and ACK number are traced.

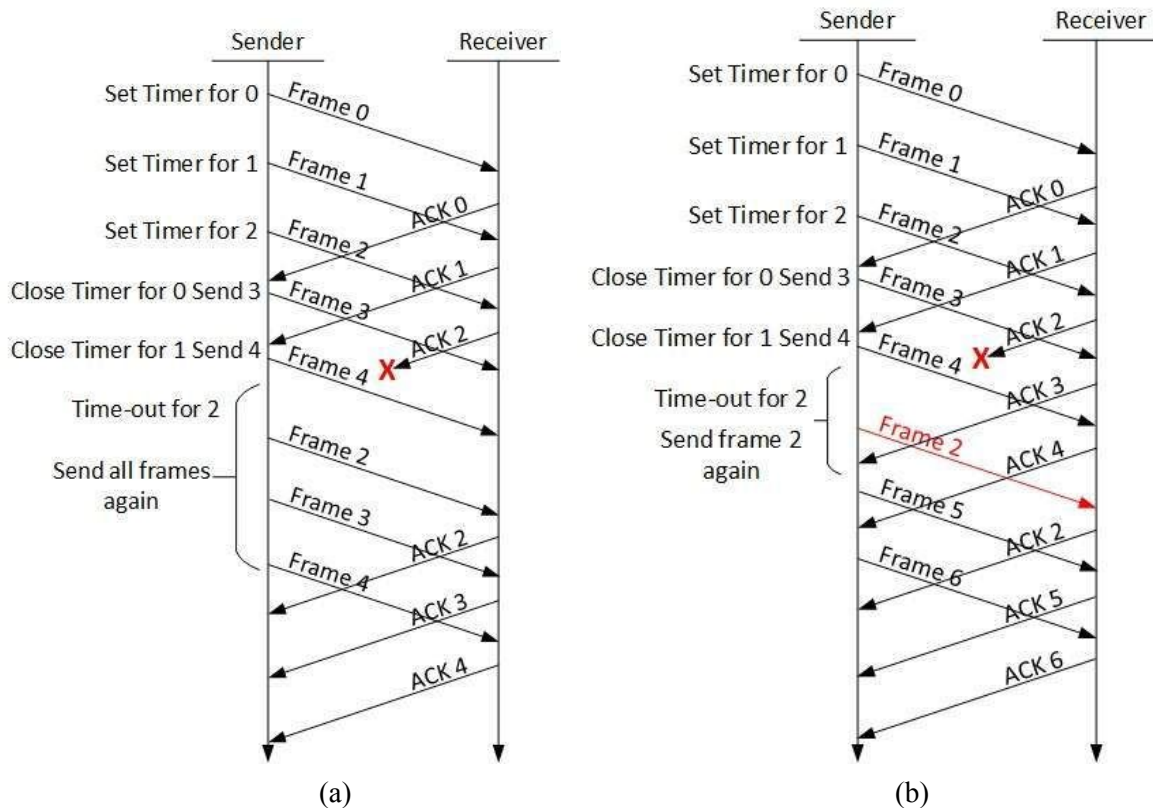


Figure 2. Shows the difference in the two window based protocols i.e. GoBackN (a) and Selective Repeat (b) as it was seen and implemented in the project.

2. Implementation Details

This section discusses the program implementations in detail and provide the different evaluation criteria for the 3 different protocols implemented.

2.1 Creating a dummy file

Our program can handle file of any size. We created a dummy file in Ubuntu, using the following command line:

```
base64 /dev/urandom | head -c 10000000 > file.txt //This created a file with name file.txt size of 10MB.
```

2.2 Emulating Unreliable Network

In order to test the developed project on an unreliable network, the code was emulated using the netem program, which is a simple network packet loss emulation tool in Linux. In order to use netem, first the iproute package was installed through the Ubuntu package manager and the following packet loss configurations were tested. A sample of running the netem emulator is shown below.

```
ddmadmin@ddmnn1:~$ sudo tc qdisc add dev lo root netem delay 30ms 20ms 25% loss 0.5% duplicate 1% reorder 25% 50%
ddmadmin@ddmnn1:~$ sudo tc qdisc del dev lo root netem
ddmadmin@ddmnn1:~$ sudo tc qdisc add dev lo root netem delay 30ms 20ms 25% loss 0.5% duplicate 1% reorder 25% 50%
ddmadmin@ddmnn1:~$ sudo tc qdisc del dev lo root netem
[sudo] password for ddmadmin:
ddmadmin@ddmnn1:~$ sudo tc qdisc add dev lo root netem delay 30ms 20ms 25% loss 0.5% duplicate 1% reorder 25% 50%
```

2.2.1 Creating a Secret Pass key

Since networks are unreliable and unsafe, a secret pass key was also added as part of the unreliable network defenses.

2.2.2 Time out & Window size

Timeout was set to be .5 seconds for all three types of protocols. Window size was kept to 5 for easy tracing of the packets visually. There is no limitation on window size and the program was tested with all different sizes ranging from 1 to the maximum possible window to contain all packets.

2.3 Simulation 1.

This simulation will be run under perfect network conditions without any delay, loss or duplication

This Configuration says

1. 0% of the packets will be delay
2. Packet loss rate is 0%
3. Packet duplication rate is 0%

2.3.1 Stop-N-Wait - Sim1

Sender

```
./sender 49155 file.txt
Waiting for receiver

Secret Pass key Matched!
Sending file.txt to the Receiver

Number of packets for the file file.txt is: 10
Packet Array initialized
Created array of packets with 10 packets

Transmitting packet number 0
ACK for the packet 0 received
Transmitting packet number 1024
ACK for the packet 1024 received
Transmitting packet number 2048
ACK for the packet 2048 received
Transmitting packet number 3072
ACK for the packet 3072 received
Transmitting packet number 4096
ACK for the packet 4096 received
Transmitting packet number 5120
ACK for the packet 5120 received
Transmitting packet number 6144
ACK for the packet 6144 received
Transmitting packet number 7168
ACK for the packet 7168 received
Transmitting packet number 8192
ACK for the packet 8192 received
Transmitting packet number 9216
ACK for the packet 9216 received
Done with file transfer
```

Receiver

```
./receiver localhost 49155 Communications
Sending request to the sender

The secret passkey : Communications
Packet type: data packet.
Allocated space for 10 packets
Got packet number 0.
Sent ACK seqnum 0

Got packet number 1024
Packet type: data packet.
Sent ACK seqnum 1024

Got packet number 2048
Packet type: data packet.
Sent ACK seqnum 2048

Got packet number 3072
Packet type: data packet.
Sent ACK seqnum 3072

Got packet number 4096
Packet type: data packet.
Sent ACK seqnum 4096

Got packet number 5120
Packet type: data packet.
Sent ACK seqnum 5120

Got packet number 6144
Packet type: data packet.
Sent ACK seqnum 6144

Got packet number 7168
Packet type: data packet.
Sent ACK seqnum 7168

Got packet number 8192
Packet type: data packet.
Sent ACK seqnum 8192

Got packet number 9216
Packet type: data packet.
Sent ACK seqnum 9216

Done receiving file packets and sending ACKs back
FILE COPY DONE - Exiting Gracefully
```


2.3.2 Go-Back-N - Sim1

Sender

```
Projects/Project2/GoBackN$ ./sender 49155 file.txt  
Waiting for receiver
```

```
Secret Pass key Matched!  
Sending file.txt to the Receiver
```

```
Number of packets for the file file.txt is: 8  
Packet Array initialized  
Created array of packets with 8 packets
```

```
Window Size if 5 packets  
Transmitting packet number 0  
Transmitting packet number 1024  
Transmitting packet number 2048  
Transmitting packet number 3072  
Transmitting packet number 4096  
ACK for the packet 0 received  
ACK for the packet 1024 received  
Transmitting packet number 5120  
Transmitting packet number 6144  
ACK for the packet 2048 received  
ACK for the packet 3072 received  
Transmitting packet number 7168  
ACK for the packet 4096 received  
ACK for the packet 5120 received  
ACK for the packet 6144 received  
ACK for the packet 7168 received  
ACK for the last packet received  
Done with file transfer
```

Receiver

```
Project2/GoBackN$ ./receiver localhost 49155 Communications  
Sending request to the sender
```

```
The secret passkey : Communications  
Packet type: data packet.  
Allocated space for 8 packets  
Got packet number 0.  
Sent ACK seqnum 0
```

```
Got packet number 1024  
Packet type: data packet.  
Sent ACK seqnum 1024
```

```
Got packet number 2048  
Packet type: data packet.  
Sent ACK seqnum 2048
```

```
Got packet number 3072  
Packet type: data packet.  
Sent ACK seqnum 3072
```

```
Got packet number 4096  
Packet type: data packet.  
Sent ACK seqnum 4096
```

```
Got packet number 5120  
Packet type: data packet.  
Sent ACK seqnum 5120
```

```
Got packet number 6144  
Packet type: data packet.  
Sent ACK seqnum 6144
```

```
Got packet number 7168  
Packet type: data packet.  
Sent ACK seqnum 7168
```

```
Done receiving file packets and sending ACKs back  
FILE COPY DONE - Exiting Gracefully
```


2.3.4 Selective Repeat - Sim1

Sender

```
jects/Project2/Selective Repeat$ ./sender 49155 file.txt  
Waiting for receiver
```

```
Secret Pass key Matched!  
Sending file.txt to the Receiver
```

```
Number of packets for the file file.txt is: 8  
Packet Array initialized  
Created array of packets with 8 packets
```

```
Window Size if 5 packets  
Transmitting packet number 0  
Transmitting packet number 1024  
Transmitting packet number 2048  
Transmitting packet number 3072  
Transmitting packet number 4096  
ACK for the packet 0 received  
ACK for the packet 1024 received  
Transmitting packet number 5120  
Transmitting packet number 6144  
ACK for the packet 2048 received  
ACK for the packet 3072 received  
Transmitting packet number 7168  
ACK for the packet 4096 received  
ACK for the packet 5120 received  
ACK for the packet 6144 received  
ACK for the packet 7168 received  
ACK for the last packet received  
Done with file transfer
```

Receiver

```
ject2/Selective Repeat$ ./receiver localhost 49155 Communications  
Sending request to the sender
```

```
The secret passkey : Communications  
Packet type: data packet.  
Allocated space for 8 packets  
Got packet number 0.  
Sent ACK seqnum 0
```

```
Got packet number 1024  
Packet type: data packet.  
Sent ACK seqnum 1024
```

```
Got packet number 2048  
Packet type: data packet.  
Sent ACK seqnum 2048
```

```
Got packet number 3072  
Packet type: data packet.  
Sent ACK seqnum 3072
```

```
Got packet number 4096  
Packet type: data packet.  
Sent ACK seqnum 4096
```

```
Got packet number 5120  
Packet type: data packet.  
Sent ACK seqnum 5120
```

```
Got packet number 6144  
Packet type: data packet.  
Sent ACK seqnum 6144
```

```
Got packet number 7168  
Packet type: data packet.  
Sent ACK seqnum 7168
```

```
Done receiving file packets and sending ACKs back  
FILE COPY DONE - Exiting Gracefully
```

2.4 Simulation 2.

Sudo tc qdisc add dev lo root netem delay 100ms 20ms 25% loss 0.5% duplicate 1% reorder 25% 50%

This configuration says:

1. Delay is 100ms \pm 20ms with the next random element depending 25% on the last one.
2. Packet loss rate is 0.5%
3. Packet duplicate rate is 1%•
4. 25% of packets (with a correlation of 50%) will get sent immediately, others will be delayed.

Simulation Results of the 3 methods are shown below

2.4.1 Stop N Wait - Sim2

Sender

```
Project2/StopNwait$ ./sender 49155 file.txt
Waiting for receiver
```

```
Secret Pass key Matched!
Sending file.txt to the Receiver
```

```
Number of packets for the file file.txt is: 10
Packet Array initialized
Created array of packets with 10 packets
```

```
Transmitting packet number 0
ACK for the packet 0 received
Transmitting packet number 1024
Packet number 1024 has timed out. Repeated ~ 20 times
Retransmitting packet number 1024
ACK for the packet 1024 received
Transmitting packet number 2048
Packet number 2048 has timed out. Repeated ~ 100 times
Retransmitting packet number 2048
ACK for the packet 2048 received
Transmitting packet number 3072
Packet number 3072 has timed out.
Retransmitting packet number 3072
ACK for the packet 3072 received
Transmitting packet number 4096
Packet number 4096 has timed out.
Retransmitting packet number 4096
ACK for the packet 4096 received
Transmitting packet number 5120
Packet number 5120 has timed out.
Retransmitting packet number 5120
ACK for the packet 5120 received
Transmitting packet number 6144
Packet number 6144 has timed out.
Retransmitting packet number 6144
ACK for the packet 6144 received
Transmitting packet number 7168
Packet number 7168 has timed out. Repeated ~ 100 times
Retransmitting packet number 7168
ACK for the packet 7168 received
Transmitting packet number 8192
Packet number 8192 has timed out.
Retransmitting packet number 8192
ACK for the packet 8192 received
Transmitting packet number 9216
Packet number 9216 has timed out.
Retransmitting packet number 9216
ACK for the packet 9216 received
ACK for the last packet received
Done with file transfer
```

Receive

```
Project2/StopNwait$ ./receiver localhost 49155 Communications
Sending request to the sender
```

```
The secret passkey : Communications
Packet type: data packet.
Allocated space for 10 packets
Got packet number 0.
Sent ACK seqnum 0
```

```
Packet type: data packet.
Got packet number 1024
Sent ACK seqnum 1024
```

```
Packet type: data packet.
Got packet number 2048
Sent ACK seqnum 2048
```

```
Packet type: data packet.
Got packet number 3072
Sent ACK seqnum 3072
```

```
Packet type: data packet.
Got packet number 4096
Sent ACK seqnum 4096
```

```
Packet type: data packet.
Got packet number 5120
Sent ACK seqnum 5120
```

```
Packet type: data packet.
Got packet number 6144
Sent ACK seqnum 6144
```

```
Packet type: data packet.
Got packet number 7168
Sent ACK seqnum 7168
```

```
Packet type: data packet.
Got packet number 8192
Sent ACK seqnum 8192
```

```
Packet type: data packet.
Got packet number 9216
Sent ACK seqnum 9216
```

```
Done receiving file packets and sending ACKs back
FILE COPY DONE - Exiting Gracefully
```

2.4.2 Go Back N - Sim2

Sender

```
Protocol-master/GoBackN$ ./sender 49155 file.txt
Waiting for receiver
```

```
Secret Pass key Matched!
Sending file.txt to the Receiver
```

```
Number of packets for the file file.txt is: 8
Packet Array initialized
Created array of packets with 8 packets
```

```
Window Size if 5 packets
Transmitting packet number 0
Transmitting packet number 1024
Transmitting packet number 2048
Transmitting packet number 3072
Transmitting packet number 4096
ACK for the packet 1024 received
ACK for the packet 3072 received
ACK for the packet 4096 received
ACK for the packet 0 received
Transmitting packet number 5120
Transmitting packet number 6144
ACK for the packet 5120 received
ACK for the packet 6144 received
Packet number 2048 has timed out.
Retransmitting packet number 2048
Retransmitting packet number 3072
Retransmitting packet number 4096
Retransmitting packet number 5120
Retransmitting packet number 6144
ACK for the packet 5120 received
ACK for the packet 6144 received
ACK for the packet 3072 received
ACK for the packet 4096 received
Packet number 2048 has timed out.
Retransmitting packet number 2048
Retransmitting packet number 3072
Retransmitting packet number 4096
Retransmitting packet number 5120
Retransmitting packet number 6144
ACK for the packet 3072 received
ACK for the packet 4096 received
ACK for the packet 5120 received
ACK for the packet 2048 received
Transmitting packet number 7168
ACK for the packet 6144 received
ACK for the packet 7168 received
ACK for the last packet received
Done with file transfer
```

```
Done receiving file packets and sending ACKs back
File space ready - time to copy
FILE COPY DONE
```

Receiver

```
DP-TransferProtocol-master/GoBackN$ ./receiver localhost 49155 Communications
Sending request to the sender
```

```
Requesting the file: Communications
Packet type: data packet.
Allocated space for 8 packets
Got packet number 4.
Sent ACK seqnum 4
```

```
Got packet number 3072
Packet type: data packet.
Sent ACK seqnum 3072
```

```
Got packet number 0
Packet type: data packet.
Sent ACK seqnum 0
```

```
Got packet number 4096
Packet type: data packet.
Sent ACK seqnum 4096
```

```
Got packet number 5120
Packet type: data packet.
Sent ACK seqnum 5120
```

```
Got packet number 6144
Packet type: data packet.
Sent ACK seqnum 6144
```

```
Got packet number 5120
Packet type: retransmitted data packet.
Sent ACK seqnum 5120
```

```
Got packet number 6144
Packet type: retransmitted data packet.
Sent ACK seqnum 6144
```

```
Got packet number 3072
Packet type: retransmitted data packet.
Sent ACK seqnum 3072
```

```
Got packet number 4096
Packet type: retransmitted data packet.
Sent ACK seqnum 4096
```

```
Got packet number 5120
Packet type: retransmitted data packet.
Sent ACK seqnum 5120
```

```
Got packet number 3072
Packet type: retransmitted data packet.
Sent ACK seqnum 3072
```

```
Got packet number 2048
Packet type: retransmitted data packet.
Sent ACK seqnum 2048
```

```
Got packet number 4096
Packet type: retransmitted data packet.
Sent ACK seqnum 4096
```

```
Got packet number 6144
Packet type: retransmitted data packet.
Sent ACK seqnum 6144
```

```
Got packet number 7168
Packet type: data packet.
Sent ACK seqnum 7168
```


2.4.3 Selective Repeat - Sim2

Sender

Receiver

```
jects/Project2/Selective Repeat$ ./sender 49155 file.txt
Waiting for receiver

Secret Pass key Matched!
Sending file.txt to the Receiver

Number of packets for the file file.txt is: 16
Packet Array initialized
Created array of packets with 16 packets

Window Size if 5 packets
Transmitting packet number 0
Transmitting packet number 1024
Transmitting packet number 2048
Transmitting packet number 3072
Transmitting packet number 4096
ACK for the packet 0 received
Transmitting packet number 5120
ACK for the packet 4096 received
ACK for the packet 3072 received
ACK for the packet 2048 received
ACK for the packet 1024 received
Transmitting packet number 6144
Transmitting packet number 7168
Transmitting packet number 8192
Transmitting packet number 9216
ACK for the packet 5120 received
Transmitting packet number 10240
ACK for the packet 6144 received
Transmitting packet number 11264
ACK for the packet 8192 received
ACK for the packet 9216 received
ACK for the packet 7168 received
Transmitting packet number 12288
Transmitting packet number 13312
Transmitting packet number 14336
ACK for the packet 10240 received
Transmitting packet number 15360
ACK for the packet 11264 received
ACK for the packet 14336 received
ACK for the packet 13312 received
ACK for the packet 12288 received
ACK for the packet 15360 received
ACK for the last packet received
Done with file transfer
```

```
ject2/Selective Repeat$ ./receiver localhost 49155 Communications
Sending request to the sender

The secret passkey : Communications
Sending the Secret Code: Communications
Packet type: data packet.
Allocated space for 16 packets
Got packet number 0.
Sent ACK seqnum 0

Got packet number 3072
Packet type: data packet.
Sent ACK seqnum 3072

Got packet number 4096
Packet type: data packet.
Sent ACK seqnum 4096

Got packet number 1024
Packet type: data packet.
Sent ACK seqnum 1024

Got packet number 2048
Packet type: data packet.
Sent ACK seqnum 2048

Got packet number 5120
Packet type: data packet.
Sent ACK seqnum 5120

Got packet number 9216
Packet type: data packet.
Sent ACK seqnum 9216

Got packet number 6144
Packet type: data packet.
Sent ACK seqnum 6144

Got packet number 7168
Packet type: data packet.
Sent ACK seqnum 7168

Got packet number 8192
Packet type: data packet.
Sent ACK seqnum 8192

Got packet number 10240
Packet type: data packet.
Sent ACK seqnum 10240

Got packet number 11264
Packet type: data packet.
Sent ACK seqnum 11264

Got packet number 14336
Packet type: data packet.
Sent ACK seqnum 14336

Got packet number 12288
Packet type: data packet.
Sent ACK seqnum 12288

Got packet number 13312
Packet type: data packet.
Sent ACK seqnum 13312

Got packet number 15360
Packet type: data packet.
Sent ACK seqnum 15360

Done receiving file packets and sending ACKs back
FILE COPY DONE - Exiting Gracefully
```

2.5 Simulation 3

Sudo tc qdisc add dev lo root netem delay 100ms

This configuration says:

1. Constant Delay of 100ms is added to outgoing packets of the external interface localhost

Simulation Results of the 3 methods are shown below

2.5.1 Stop N Wait - Sim3

Sender

Project2/StopNwait\$./sender 49155 file.txt
Waiting for receiver

Secret Pass key Matched!
Sending file.txt to the Receiver

Number of packets for the file file.txt is: 10
Packet Array initialized
Created array of packets with 10 packets

Transmitting packet number 0
ACK for the packet 0 received
Transmitting packet number 1024
Packet number 1024 has timed out.
Retransmitting packet number 1024
ACK for the packet 1024 received
Transmitting packet number 2048
Packet number 2048 has timed out. Repeated ~ 70 times
Retransmitting packet number 2048
ACK for the packet 2048 received
Transmitting packet number 3072
Packet number 3072 has timed out.
Retransmitting packet number 3072
ACK for the packet 3072 received
Transmitting packet number 4096
Packet number 4096 has timed out.
Retransmitting packet number 4096
ACK for the packet 4096 received
Transmitting packet number 5120
Packet number 5120 has timed out.
Retransmitting packet number 5120
ACK for the packet 5120 received
Transmitting packet number 6144
Packet number 6144 has timed out.
Retransmitting packet number 6144
ACK for the packet 6144 received
Transmitting packet number 7168
Packet number 7168 has timed out.
Retransmitting packet number 7168
ACK for the packet 7168 received
Transmitting packet number 8192
Packet number 8192 has timed out.
Retransmitting packet number 8192
ACK for the packet 8192 received
Transmitting packet number 9216
Packet number 9216 has timed out.
Retransmitting packet number 9216
ACK for the packet 9216 received
ACK for the last packet received
Done with file transfer

Receiver

saad@saad5810:~/GoogleDrive/Courses/ECE634Networking/Projects/Project2/
StopNwait\$./receiver localhost 49155 Communications
Sending request to the sender

The secret passkey : Communications
Packet type: data packet.
Allocated space for 10 packets
Got packet number 0.
Sent ACK seqnum 0

Packet type: data packet.
Got packet number 1024
Sent ACK seqnum 1024

Packet type: data packet.
Got packet number 2048
Sent ACK seqnum 2048

Packet type: data packet.
Got packet number 3072
Sent ACK seqnum 3072

Packet type: data packet.
Got packet number 4096
Sent ACK seqnum 4096

Packet type: data packet.
Got packet number 5120
Sent ACK seqnum 5120

Packet type: data packet.
Got packet number 6144
Sent ACK seqnum 6144

Packet type: data packet.
Got packet number 7168
Sent ACK seqnum 7168

Packet type: data packet.
Got packet number 8192
Sent ACK seqnum 8192

Packet type: data packet.
Got packet number 9216
Sent ACK seqnum 9216

Done receiving file packets and sending ACKs back
FILE COPY DONE - Exiting Gracefully

2.5.2 Go Back N - Sim3

Sender

```
jects/Project2/GoBackN$ ./sender 49155 file.txt  
Waiting for receiver
```

```
Secret Pass key Matched!  
Sending file.txt to the Receiver
```

```
Number of packets for the file file.txt is: 8  
Packet Array initialized  
Created array of packets with 8 packets
```

```
Window Size if 5 packets  
Transmitting packet number 0  
Transmitting packet number 1024  
Transmitting packet number 2048  
Transmitting packet number 3072  
Transmitting packet number 4096  
Packet number 0 has timed out.  
Retransmitting packet number 0  
Retransmitting packet number 1024  
Retransmitting packet number 2048  
Retransmitting packet number 3072  
Retransmitting packet number 4096  
ACK for the packet 0 received  
Transmitting packet number 5120  
ACK for the packet 1024 received  
ACK for the packet 2048 received  
Transmitting packet number 6144  
Transmitting packet number 7168  
ACK for the packet 3072 received  
ACK for the packet 4096 received  
ACK for the packet 5120 received  
ACK for the packet 6144 received  
ACK for the packet 7168 received  
ACK for the last packet received  
Done with file transfer
```

Receiver

```
ject2/GoBackN$ ./receiver localhost 49155 Communications  
Sending request to the sender
```

```
The secret passkey : Communications  
Packet type: data packet.  
Allocated space for 8 packets  
Got packet number 0.  
Sent ACK seqnum 0
```

```
Got packet number 1024  
Packet type: data packet.  
Sent ACK seqnum 1024
```

```
Got packet number 2048  
Packet type: data packet.  
Sent ACK seqnum 2048
```

```
Got packet number 3072  
Packet type: data packet.  
Sent ACK seqnum 3072
```

```
Got packet number 4096  
Packet type: data packet.  
Sent ACK seqnum 4096
```

```
Got packet number 0  
Packet type: retransmitted data packet.  
Sent ACK seqnum 0
```

```
Got packet number 1024  
Packet type: retransmitted data packet.  
Sent ACK seqnum 1024
```

```
Got packet number 2048  
Packet type: retransmitted data packet.  
Sent ACK seqnum 2048
```

```
Got packet number 3072  
Packet type: retransmitted data packet.  
Sent ACK seqnum 3072
```

```
Got packet number 4096  
Packet type: retransmitted data packet.  
Sent ACK seqnum 4096
```

```
Got packet number 5120  
Packet type: data packet.  
Sent ACK seqnum 5120
```

```
Got packet number 6144  
Packet type: data packet.  
Sent ACK seqnum 6144
```

```
Got packet number 7168  
Packet type: data packet.  
Sent ACK seqnum 7168
```

```
Done receiving file packets and sending ACKs back  
FILE COPY DONE - Exiting Gracefully
```


2.5.3 Selective Repeat - Sim3

Sender

```
jects/Project2/Selective Repeat$ ./sender 49155 file.txt  
Waiting for receiver
```

```
Secret Pass key Matched!  
Sending file.txt to the Receiver
```

```
Number of packets for the file file.txt is: 8  
Packet Array initialized  
Created array of packets with 8 packets
```

```
Window Size if 5 packets  
Transmitting packet number 0  
Transmitting packet number 1024  
Transmitting packet number 2048  
Transmitting packet number 3072  
Transmitting packet number 4096  
ACK for the packet 0 received  
Transmitting packet number 5120  
ACK for the packet 1024 received  
ACK for the packet 2048 received  
Transmitting packet number 6144  
Transmitting packet number 7168  
ACK for the packet 3072 received  
ACK for the packet 4096 received  
ACK for the packet 5120 received  
ACK for the packet 6144 received  
ACK for the packet 7168 received  
ACK for the last packet received  
Done with file transfer
```

Receiver

```
ject2/Selective Repeat$ ./receiver localhost 49155 Communications  
Sending request to the sender
```

```
The secret passkey : Communications  
Packet type: data packet.  
Allocated space for 8 packets  
Got packet number 0.  
Sent ACK seqnum 0
```

```
Got packet number 1024  
Packet type: data packet.  
Sent ACK seqnum 1024
```

```
Got packet number 2048  
Packet type: data packet.  
Sent ACK seqnum 2048
```

```
Got packet number 3072  
Packet type: data packet.  
Sent ACK seqnum 3072
```

```
Got packet number 4096  
Packet type: data packet.  
Sent ACK seqnum 4096
```

```
Got packet number 5120  
Packet type: data packet.  
Sent ACK seqnum 5120
```

```
Got packet number 6144  
Packet type: data packet.  
Sent ACK seqnum 6144
```

```
Got packet number 7168  
Packet type: data packet.  
Sent ACK seqnum 7168
```

```
Done receiving file packets and sending ACKs back  
FILE COPY DONE - Exiting Gracefully
```

2.6 Simulation 4.

Sudo tc qdisc add dev lo root netem delay 20ms 30ms 40% loss 0.5% duplicate 5% reorder 25% 50%

This configuration says:

5. Delay is 20ms \pm 30ms with the next random element depending 40% on the last one.
6. Packet loss rate is 0.5%
7. Packet duplicate rate is 5%
8. 25% of packets (with a correlation of 50%) will get sent immediately, others will be delayed.

Simulation Results of the 3 methods are shown below

2.6.1 Stop N Wait - Sim4

Sender

Receiver

<pre>Project2/StopNwait\$./sender 49155 file.txt Waiting for receiver Secret Pass key Matched! Sending file.txt to the Receiver Number of packets for the file file.txt is: 10 Packet Array initialized Created array of packets with 10 packets Transmitting packet number 0 ACK for the packet 0 received Transmitting packet number 1024 Packet number 1024 has timed out. Retransmitting packet number 1024 ACK for the packet 1024 received Transmitting packet number 2048 Packet number 2048 has timed out. Retransmitting packet number 2048 ACK for the packet 2048 received Transmitting packet number 3072 Packet number 3072 has timed out. Retransmitting packet number 3072 ACK for the packet 3072 received Transmitting packet number 4096 Packet number 4096 has timed out. Retransmitting packet number 4096 ACK for the packet 4096 received Transmitting packet number 5120 Packet number 5120 has timed out. Retransmitting packet number 5120 ACK for the packet 5120 received Transmitting packet number 6144 Packet number 6144 has timed out. Repeated ~ 50 times Retransmitting packet number 6144 ACK for the packet 6144 received Transmitting packet number 7168 Packet number 7168 has timed out. Retransmitting packet number 7168 ACK for the packet 7168 received Transmitting packet number 8192 Packet number 8192 has timed out. Retransmitting packet number 8192 ACK for the packet 8192 received Transmitting packet number 9216 Packet number 9216 has timed out. Retransmitting packet number 9216 ACK for the packet 9216 received ACK for the last packet received Done with file transfer</pre>	<pre>saad@saad5810:~/GoogleDrive/Courses/ECE634Networking/Projects/Project 2/StopNwait\$./receiver localhost 49155 Communications Sending request to the sender The secret passkey : Communications Packet type: data packet. Allocated space for 10 packets Got packet number 0. Sent ACK seqnum 0 Packet type: data packet. Got packet number 1024 Sent ACK seqnum 1024 Packet type: data packet. Got packet number 2048 Sent ACK seqnum 2048 Packet type: data packet. Got packet number 3072 Sent ACK seqnum 3072 Packet type: data packet. Got packet number 4096 Sent ACK seqnum 4096 Packet type: data packet. Got packet number 5120 Sent ACK seqnum 5120 Packet type: data packet. Got packet number 6144 Sent ACK seqnum 6144 Packet type: data packet. Got packet number 7168 Sent ACK seqnum 7168 Packet type: data packet. Got packet number 8192 Sent ACK seqnum 8192 Packet type: data packet. Got packet number 9216 Sent ACK seqnum 9216 Done receiving file packets and sending ACKs back FILE COPY DONE - Exiting Gracefully</pre>
---	---

2.6.2 Go-Back-N - Sim 4

Sender

```
jects/Project2/GoBackN$ ./sender 49155 file.txt
Waiting for receiver

Secret Pass key Matched!
Sending file.txt to the Receiver

Number of packets for the file file.txt is: 8
Packet Array initialized
Created array of packets with 8 packets

Window Size if 5 packets
Transmitting packet number 0
Transmitting packet number 1024
Transmitting packet number 2048
Transmitting packet number 3072
Transmitting packet number 4096
ACK for the packet 4096 received
ACK for the packet 2048 received
ACK for the packet 0 received
Transmitting packet number 5120
ACK for the packet 3072 received
ACK for the packet 5120 received
Packet number 1024 has timed out.
Retransmitting packet number 1024
Retransmitting packet number 2048
Retransmitting packet number 3072
Retransmitting packet number 4096
Retransmitting packet number 5120
ACK for the packet 4096 received
ACK for the packet 1024 received
Transmitting packet number 6144
ACK for the packet 2048 received
Transmitting packet number 7168
ACK for the packet 3072 received
ACK for the packet 5120 received
ACK for the packet 7168 received
ACK for the packet 6144 received
ACK for the last packet received
Done with file transfer
```

Receiver

```
ject2/GoBackN$ ./receiver localhost 49155 Communications
Sending request to the sender

The secret passkey : Communications
Sending the Secret Code: Communications
Packet type: data packet.
Allocated space for 8 packets
Got packet number 1.
Sent ACK seqnum 1

Got packet number 2048
Packet type: data packet.
Sent ACK seqnum 2048

Got packet number 4096
Packet type: data packet.
Sent ACK seqnum 4096

Got packet number 0
Packet type: data packet.
Sent ACK seqnum 0

Got packet number 3072
Packet type: data packet.
Sent ACK seqnum 3072

Got packet number 5120
Packet type: data packet.
Sent ACK seqnum 5120

Got packet number 4096
Packet type: retransmitted data packet.
Sent ACK seqnum 4096

Got packet number 1024
Packet type: retransmitted data packet.
Sent ACK seqnum 1024

Got packet number 2048
Packet type: retransmitted data packet.
Sent ACK seqnum 2048

Got packet number 3072
Packet type: retransmitted data packet.
Sent ACK seqnum 3072

Got packet number 5120
Packet type: retransmitted data packet.
Sent ACK seqnum 5120

Got packet number 7168
Packet type: data packet.
Sent ACK seqnum 7168

Got packet number 6144
Packet type: data packet.
Sent ACK seqnum 6144

Done receiving file packets and sending ACKs back
FILE COPY DONE - Exiting Gracefully
```


2.6.3 Selective Repeat - Sim 4

Sender

```
jects/Project2/Selective Repeat$ ./sender 49155 file.txt
Waiting for receiver

Secret Pass key Matched!
Sending file.txt to the Receiver

Number of packets for the file file.txt is: 8
Packet Array initialized
Created array of packets with 8 packets

Window Size if 5 packets
Transmitting packet number 0
Transmitting packet number 1024
Transmitting packet number 2048
Transmitting packet number 3072
Transmitting packet number 4096
ACK for the packet 0 received
Transmitting packet number 5120
ACK for the packet 2048 received
ACK for the packet 3072 received
ACK for the packet 4096 received
ACK for the packet 5120 received
Packet number 1024 has timed out.
Retransmitting packet number 1024
ACK for the packet 1024 received
Transmitting packet number 6144
Transmitting packet number 7168
ACK for the packet 7168 received
ACK for the packet 6144 received
ACK for the last packet received
Done with file transfer
```

Receiver

```
ject2/Selective Repeat$ ./receiver localhost 49155 Communications
Sending request to the sender

The secret passkey : Communications
Packet type: data packet.
Allocated space for 8 packets
Got packet number 1.
Sent ACK seqnum 1

Got packet number 0
Packet type: data packet.
Sent ACK seqnum 0

Got packet number 2048
Packet type: data packet.
Sent ACK seqnum 2048

Got packet number 4096
Packet type: data packet.
Sent ACK seqnum 4096

Got packet number 3072
Packet type: data packet.
Sent ACK seqnum 3072

Got packet number 5120
Packet type: data packet.
Sent ACK seqnum 5120

Got packet number 1024
Packet type: retransmitted data packet.
Sent ACK seqnum 1024

Got packet number 6144
Packet type: data packet.
Sent ACK seqnum 6144

Got packet number 7168
Packet type: data packet.
Sent ACK seqnum 7168

Done receiving file packets and sending ACKs back
FILE COPY DONE - Exiting Gracefully
```

2.7 Ending the simulation

Finally the simulations were ended by disabling the netem delay product on the localhost interface by deleting the rules using the following.

```
Sudo tc qdisc del dev lo root netem
```

3. Comparing Performances of the three Protocols

Stop & Wait

The major shortcoming of the stop-and-wait protocol is that requires that the sender to have only one outgoing frame on the sending media at any given time. The sender waits till it gets the ACK back of sent frame before sending the next frame. This causes substantial amount of network bandwidth wastage. To improve efficiency while providing reliability, "sliding window" protocols are a better option.

Go-Back-N vs Sliding Window

The sliding window protocols does not waste any network bandwidth as compared to the Stop-N-Wait protocol both in normal and in congested conditions. Both Sliding window protocols GBN and SR show better performance than Stop-N-Wait as shown in the comparison graph below.

1. It was observed that delay is the biggest contributing factor to the performance in all three implemented protocols.
2. The Stop & Wait protocol is good for small file sizes under perfect network conditions but it performed the worst under deteriorated network conditions.
3. Since netem emulated delays and corruption in bursts, some packets of Stop-N-Wait kept delaying for as long as 150 cycles, whereas the other ones were quicker.
4. Selective Repeat protocol was the best performing of the three

4. Conclusion

The objective of this project was to implement several different transport protocols using C language in a Linux based environment and emulate real transmission conditions. To achieve the desired objectives the following three Transport Protocols were implemented

1. Stop N Wait
2. Go Back N
3. Selective Repeat

Their efficiencies were also compared using a virtual network environment created using the netem emulator that we used. For the performance comparison of the developed protocols all three protocols were compared in both normal conditions and real-life emulations. In the normal condition there is no packet loss and the protocols performed fast and reliably. In the real life emulation or otherwise also called the congested condition there were some packet loss some packet delay and duplication as controlled by our emulator. All three protocols were compared in the given conditions and results were reported.