IJARIIT INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 7, Issue 3 - V7I3-1422)
Available online at: https://www.ijariit.com

Implementation of Sliding Window Protocol over lossy connections

Yash Deshpande
yashsd99@gmail.com
Terna College of Engineering, Mumbai, Maharashtra

Rucha Tirodkar
ruchatirodkar@ternaengg.ac.in
Terna College of Engineering, Mumbai, Maharashtra

Piyush Verma
piyushverma@ternaengg.ac.in
Terna College of Engineering, Mumbai, Maharashtra

Arjun Bhardwaj
arjunbhardwaj@ternaengg.ac.in
Terna College of Engineering, Mumbai, Maharashtra

Nayana Vaity
nayanavaity@ternaengg.ac.in
Terna College of Engineering, Mumbai, Maharashtra

ABSTRACT

In the Lossy Networks, the loss of data packets is very prominent and concerns of worries for efficiency and sustainability of the system. However, by using Sliding Window Protocol both the sender and receiver keeps a window of acknowledgement. The sender keeps the knowledge about expected acknowledgement and receivers keeps the knowledge about receiving frames. Due to which, loss of transmission of data is minimal and efficiency of the system is improved.

Keywords: *Lossy Networks, Data Packets, Acknowledgements, Sliding Window Protocol.*

1. INTRODUCTION

In the age of internet data (information) is considered to be the next oil. People are now exchanging data from the tip of the fingers. People depends upon the flawless connection to a great content. When a user sends the data, it is important that the sender and receiver will receive the data in the same form without attenuation.

Implementing a stable connection is a must. To achieve this many techniques are used. Transmission Control Protocol & User Datagram Protocol are commonly used protocols used for transmission of data.

TCP is connection-based Protocol, we have to first negotiate a connection before we can do anything. UDP is connectionless Protocol, so need to justify a connection is removed.

Main aim of the Project is to Implement Sliding Window Protocol over UDP layer.

- To build a file sharing application where a user will be able to share files with the server.
- To implement sliding window protocol to ensure reliable and in order transmission of data (packets).

2. DRAWBACKS OF EXISTING SYSTEM

The current & previous file sharing platforms are having some drawbacks which are as follows:

- One Bit Sliding Window Protocol also called as Stop- And Wait Protocol, Sender sends the data and waits until the acknowledgement arrives.
- Stop- And- Wait protocol is very inefficient and time consuming.
- At the moment only one Data packet is in transmission, sender has to wait until one round to send the next data.
- Existing System wait time is longer for slow networks.

3. SUGGESTED IMPROVEMENTS

• We are using Go-Back-N Protocol for transmission of Data in the system.

• Every Data is sent with a Ack, if the Ack is lost in between transmission then sender resents the Ack after the timeout.

• Each and Every Data will be received by the receiver with the help of Ack methodology.

4. METHODOLOGY AND IMPLEMENTATION

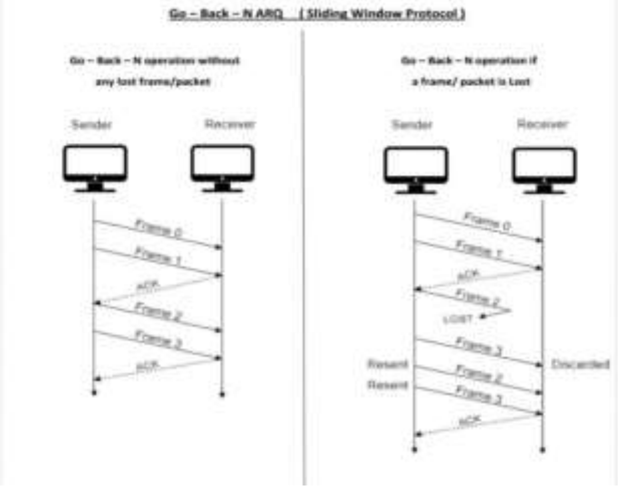

Go – Back – N ARQ (Sliding Window Protocol)
Go – Back – N operation without any lost frame/packet
Go – Back – N operation if a frame/ packet is Lost
Sender
Receiver
Sender
Receiver
Frame 0
Frame 1
ACK
Frame 2
Frame 3
ACK
Frame 0
Frame 1
ACK
Frame 2
LOST
Frame 3
Discarded
Resent
Frame 2
Resent
Frame 3
ACK

Fig. 1: Implementation of Sliding Window Protocol

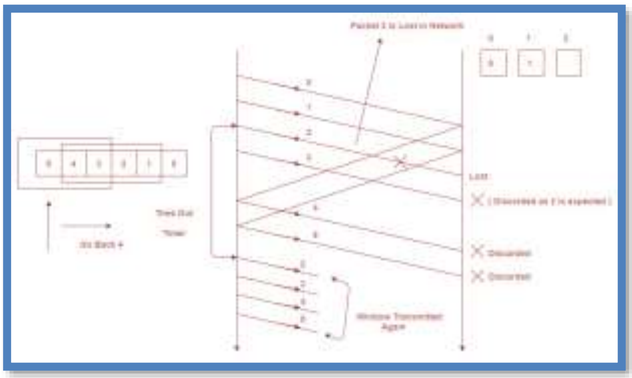
The project aimed towards developing a file sharing application over the UDP layer. As we were dealing with problem that we need to build a file sharing system, the most convenient way to approach this would be to use a Linux based system as file sharing can be simulated using server and a client. The language of preference here is C as socket programming in C is very much compatible with the Linux systems and also due to abundance of resources on the internet with regards to socket programming in C, this way the errors could easily be rectified and also learn new techniques, functions, key words in the process.

Sockets are used in the program that are going to act as an endpoint of the node for receiving and sending data. The client sends the request to the server, the server then checks if the request made is valid. First step for both the client as well as the server side will be to bind the socket with the concurrent port number for transferring of data to take place. As the name suggests this process of binding or matching the socket with port number is called binding. Every port number has a different service to provide, different ports are used for different purposes. We can also say the process of assigning the port address to a socket is known as binding.

There are functions that are being used to handle the errors that might occur during the execution of the whole code, these errors could be binding errors, connection errors, listening errors etc. Different strategies were used to handle all these errors. The server starts and then waits for the client to send the request in terms of a file name. Additional information that is being sent from the client side to server side is about window size and data to be transferred in bytes.

Once the window size and size of data to be delivered is fixed by the user a request is sent to the server. The server then checks request, if the name of the file matches with any of the file in the server side of the data, the file is transferred. The file is not transferred all at once instead the data is being sent packet by packet from the server to the client. The packet size is already determined by the user at the beginning of the execution taking place. The file is sliced into many packets and sent one by one and received at the client end in the same order.

The client also sends an acknowledgement after sending two to three consecutive packets. The acknowledgement signal gives the assurance that the data packet has been received at the client side without any problem. If the acknowledgment signal is not received after certain time has passed (let's say 1ms), then the server asks the client to send all the data packets again. The data packets that have already been received are discarded. The files here are seen as continuous sequence of data bytes. The data is also streamed from server to client in similar fashion. Continuous stream of data is fragmented into small packets, packets in turn are transferred which contain all the data.



5. OVERALL DESCRIPTION

1.2 System Perspective

- We have implemented a Sliding Window Protocol in our file sharing application over UDP layer. It is developed to allow user to send and receive files over a loss network. It is a Windows Application which uses Socket Programming to implement the Protocol and transferring of Data over the Network.

5.2 System Functions:

- Select the file
- Define the Size
- Define the transmission rate of Data
- Acknowledgements of receiving files

5.3 System Features:

- Removing Attenuation from the Lossy Connection.
- Description: This feature aims to remove the redundancy in the attenuation occurs while transferring of the files.
- Priority: High
- Response Sequences: When a user attempts to send a file, it is given a certain acknowledgment by the receiver stating that the file has arrived. File will be resent, if the acknowledgment is not received.
- Achievements
- Description: User can set the transmission rate and size of the file.
- Priority: Medium
- Response Sequences: For example, User has the capability to set different limit of transmission size depending upon the files.

6. CONCLUSION

In Today's World, sharing of data has become so easy and swift. You can send a piece of information from one continent to another with in a fraction of secs via Email. But the main problem exist today is unreliable connection.

We have developed an implementation of Sliding Window Protocol Over UDP Layer. In this we used GO-BACK-N ARQ technique which is under Sliding Window Protocol. Each packet here, is sent with the adjoining acknowledgement to the receiver. If the Acknowledgement is lost in between the transmissions, the sender resends the data packets after timeout. This will ensure that every packet Is received without any loss by the receiver.

In the current system, they are no provision of sending two different data files in a single transmission round. As part of the future work, developing an algorithm in which user can vary the size of files after each transmission without need to restart the whole algorithm, is important. Multiple data types files can be able to send via the algorithm.

7. REFERENCES

[1] Slobodan Beliga University of Rijeka, Department of Informatics Radmile Matejčić 2, 51 000 Rijeka, Croatia
[2] JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants, M. Jones, B.
[3] Campbell, C. Mortimore, May 2015
[4] Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content, R. Fielding, J. Reschke
[5] A Critical Review of Recurrent Neural Networks for Sequence Learning by Zachary C. Lipton, John Berkowitz
[6] Sequence to Sequence Learning with Neural Networks by Ilya Sutskever, Oriol Vinyals, Quoc V. Le