Assignment - 12

Title : TCP / UDP sockets

Problem Statement :
Write a program using TCP / UDP sockets for wired network to implement:

a. peer to peer chat
b. multiuser chat.

Demonstrate the packets captured using Wireshark.

Objectives:

- To study about TCP and UDP sockets.
- To implement peer to peer communication using sockets.
- To implement multichat communication using sockets.

Outcomes:

- Students will be implement communication using sockets in java.

Requirements:
Text editor, & JDK

Theory:
TCP (Transmission Control Protocol) is a standard that defines how to establish and maintain a network conversation through which

application programs can exchange data. TCP works with the Internet Protocol which ~~allows~~ defines how computer sends packets of data to each other. Together TCP and IP are the basic rules defining the internet

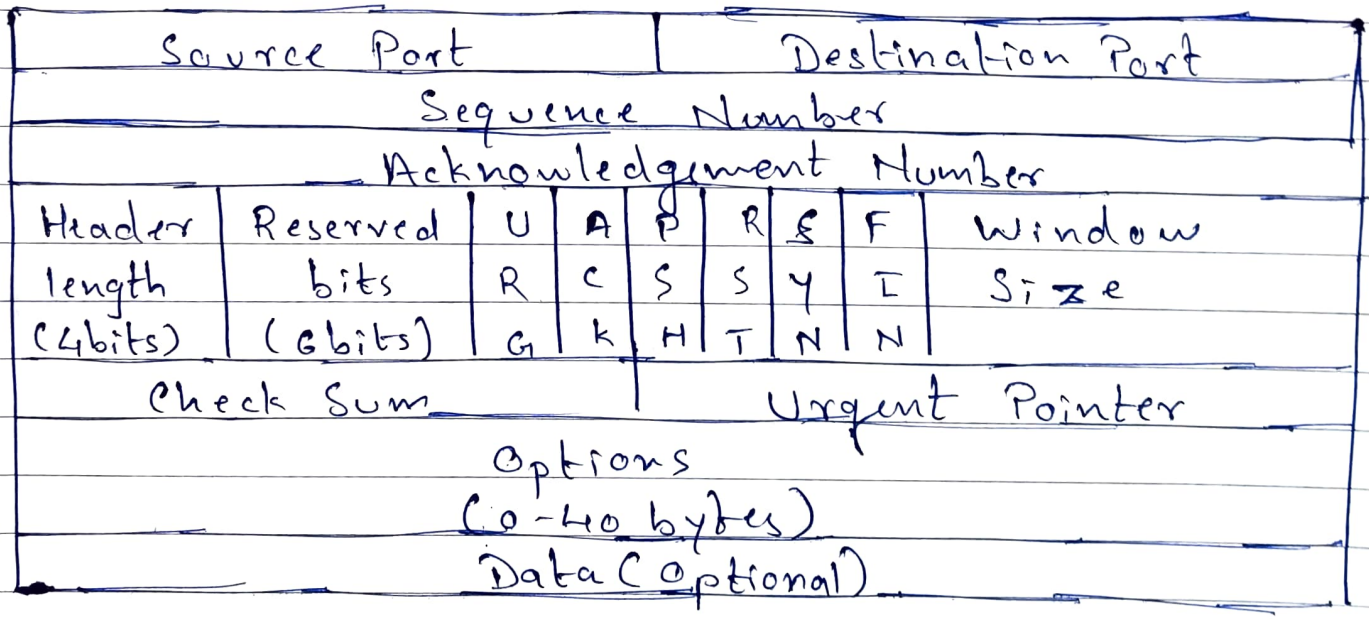
TCP is a connection oriented protocol which means a connection is established and maintained until the application programs at each end have finished exchanging messages. It determines how to break application data into packets that networks can deliver, sends packets to and accepts packets from the network layer, manages flow control and -- because it is meant to provide error-free data data transmission -- handles retransmission of dropped or garbled packets and acknowledges all the packets that arrive.

UDP (User Datagram Protocol) is a Transport Layer protocol. It is part of the IP suite ~~of~~ referred as UDP/IP suite. Unlike TCP, it is unreliable and connectionless protocol. So, there is no need to establish connection prior to

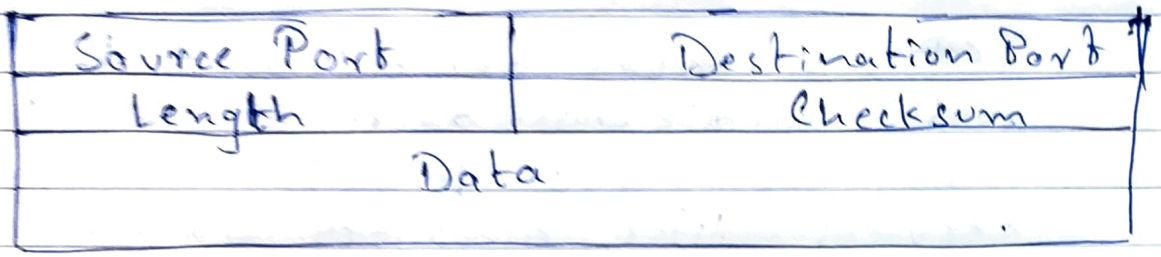
data transfer.

TCP is more widely used but ~~has~~ comes with additional overhead and latency. Here UDP comes into picture. For realtime services like computer gaming, voice or video communication, live conferences, etc, we need UDP since high performance is needed.

TCP Header:



UDP Header:



Conclusion:

Thus we have successfully implemented peer to peer and multichat server in java using TCP sockets.