**Practical No. 9 Roll No.13,14**

**Aim:Write a program to implement SSL**

**Server.Java**

import java.io.IOException;

import java.net.ServerSocket;

import javax.net.ssl.SSLServerSocketFactory;

public class clientserverins {

public static void main(String[] args)throws IOException {

System.setProperty("javax.net.ssl.keyStore", "za.store");

System.setProperty("javax.net.ssl.keyStorePassword", "123456");

ServerSocket serverSocket=((SSLServerSocketFactory)SSLServerSocketFactory.getDefault()).createServerSocket(4444);

System.out.println("Server up and ready for connection...");

while(true)

new ServerThread(serverSocket.accept()).start();

}

}

**ServerThread.java**

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.io.PrintWriter;

import java.net.Socket;

public class ServerThread extends Thread{

Socket socket;

ServerThread(Socket socket){

this.socket=socket;

}

public void run() {

try {

PrintWriter printwriter=new PrintWriter(socket.getOutputStream(),true);

BufferedReader bufferedreader=new BufferedReader(new InputStreamReader(socket.getInputStream()));

System.out.println("user"+bufferedreader.readLine()+"is now connected to server");

while(true) printwriter.println(bufferedreader.readLine()+"echo");

}

catch(Exception e) {

e.printStackTrace();

}

}

}

**Client.java**

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.io.PrintWriter;

import java.net.Socket;

import javax.net.ssl.SSLSocketFactory;

public class client {

public static void main(String [] args) throws IOException

{

System.setProperty("javax.net.ssl.trustStore", "za.store");

Socket socket=((SSLSocketFactory)SSLSocketFactory.getDefault()).createSocket("localhost",4444);

BufferedReader socketBufferedReader=new BufferedReader(new InputStreamReader(socket.getInputStream()));

PrintWriter printwriter=new PrintWriter(socket.getOutputStream(),true);

BufferedReader commandPromptBufferedReader=new BufferedReader(new InputStreamReader(System.in));

System.out.println("Plz Enter the user name");

printwriter.println(commandPromptBufferedReader.readLine());

String message=null;

while(true)

{

System.out.println("Enter the message to send to sever");

message=commandPromptBufferedReader.readLine();

if(message.equals("quit"))

{

socket.close();

break;

}

printwriter.println(message);

System.out.println("message reply from server");

System.out.println(socketBufferedReader.readLine());

}

}

}

