```java
import java.net.*;
import java.io.*;
import java.util.Scanner;

public class EavesdropperUDP{
    public static void main(String args[]){
        System.out.println("The eavesdropper is running."); // lab instructions
        DatagramSocket listenSocket = null;
        DatagramSocket masqueradeSocket = null;

        byte[] buffer = new byte[1000]; // create a buffer to receiver server packet

        try{
            InetAddress aHost = InetAddress.getByName("localhost");

            // listen port code
            System.out.print("Input a server port number to listen to: ");
            Scanner readline = new Scanner(System.in);
            int listenPort = readline.nextInt(); // convert to int

            listenSocket = new DatagramSocket(listenPort);
            listenSocket.setReuseAddress(true);

            // masquerade port code
            System.out.print("Input a server port number to masquerade as: ");
            int masqueradePort = readline.nextInt(); // convert to int

            masqueradeSocket = new DatagramSocket();
            masqueradeSocket.setReuseAddress(true);

            DatagramPacket request = new
```

```java
DatagramPacket(buffer, buffer.length); // syntax for buffer

            while(true){ // loop to continue until 'halt!' is sent
                // get response from  client
                listenSocket.receive(request);
                String requestString = new String(request.getData()).substring(0,request.getLength());
                System.out.println("\nFrom Client : "+ requestString);

                // send to server
                byte [] m = requestString.getBytes(); // convert console in into byte packets
                DatagramPacket reply = new DatagramPacket(m, m.length, aHost, masqueradePort);
                masqueradeSocket.send(reply); // send packet

                System.out.println("Sending to Server: "+ requestString);

                // get response from server
                DatagramPacket response = new DatagramPacket(buffer, buffer.length); // syntax for buffer
                masqueradeSocket.receive(response);
                String responseString = new String(response.getData()).substring(0,response.getLength());

                System.out.println("From Server: " + responseString);

                // send to client
                response = new DatagramPacket(response.getData(),  response.getLength(), request.getAddress(), request.getPort());
                listenSocket.send(response); //
send packet

                System.out.println("Sending to client: "+ responseString);

                // halt logic
                if(requestString.equalsIgnoreCase("halt!")){
                    System.out.print("**************");
                    System.out.print(" halt! message arrived **************");
                }


            }
            // catch potential exceptions
        }catch (SocketException e){System.out.println("Socket: " + e.getMessage());
        }catch (IOException e) {System.out.println("IO: " + e.getMessage());
        }finally {
            if(listenSocket != null) listenSocket.close();
            if(masqueradeSocket != null) masqueradeSocket.close();
        }
    }
}
```