NAME: Bhavik Ransubhe
CLASS : TE (B) COMP
ROLL NO : 39055

Problem Statement:

Write a program using UDP Sockets to enable file transfer (Script, Text, Audio and Video one file each) between two machines. Analyze the packets captured traces
using Wireshark Packet Analyzer Tool for peer to peer mode.

CODE:-

Client side:

```java
package com.company;

import java.io.*;

import java.net.DatagramPacket; import java.net.DatagramSocket;
import java.net.InetAddress;

import java.util.Arrays;

import java.util.Scanner;


public class Client {


  static DatagramSocket clientSocket;

  static InetAddress ip;

  static byte[] sendData;

  static byte[] receiveData;


  public Client() {

    try {


      clientSocket = new DatagramSocket();

      ip = InetAddress.getLocalHost();

      sendData = new byte[1024];
```

```
        receiveData = new byte[1024];

    } catch (Exception e) {

        System.out.println("Socket could not be connected");

    }

}

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

    Client client = new Client();

    String choice;

    char doYouWantToContinue;


    Scanner sc = new Scanner(System.in);


    do {


        System.out.print("\na. Download a File\n"

                + "b. Download a Song\n"

                + "c. Download a Video\n"

                + "Enter your choice : ");


        choice = sc.nextLine();

        sendData = choice.getBytes();

        DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, ip, 6000);
        clientSocket.send(sendPacket);


        switch (choice) {


            case "a":

                client.receiveAFile();

                break;
```

```java
        case "b":

          client.receiveASong();

          break;

        case "c":

          client.receiveAVideo();

          break;

        default:

          System.out.println("\nOops! Try again!!!\n");
          break;

      }

      System.out.print("\nDo you want to continue ? \nAns: ");

      doYouWantToContinue = sc.next().charAt(0);

      sc.nextLine();

    } while (doYouWantToContinue == 'y' || doYouWantToContinue == 'Y');


  }

  private void receiveAFile() throws IOException, InterruptedException {



    byte b[] = new byte[3072];

    String packet;

    DatagramSocket fileSocket = new DatagramSocket(1000);

    FileOutputStream fis = new FileOutputStream("D:/file from server.txt");

    Thread.sleep(4000);

    System.out.println("\nReceiving...");

    DatagramPacket dp = new DatagramPacket(b, b.length);

    fileSocket.receive(dp);

    packet = new String(dp.getData(), 0, dp.getLength());

    System.out.println("-----Contents of File-----");

        System.out.println(packet);

    System.out.println("----End of File----");
    packet.getBytes();

    fis.write(b);

    fileSocket.close();

    Thread.sleep(2000);

    System.out.println("\nFile saved Successfully!");


  }




  private void receiveASong() throws IOException, InterruptedException {


    DatagramSocket audioSocket = new DatagramSocket(2000);

    int packetsize = 1024;

    FileOutputStream fos = null;

    BufferedOutputStream bos=null;

    try {

      fos = new FileOutputStream("D:\\song from server.wav");

      bos = new BufferedOutputStream(fos);

      double nosofpackets = Math.ceil(((int) (new File("D:\\DEMOSONG.wav")).length()) / packetsize);

      byte[] mybytearray = new byte[packetsize];

    DatagramPacket receivePacket = new DatagramPacket(mybytearray,mybytearray.length);
    System.out.println(nosofpackets + " " + Arrays.toString(mybytearray) + " " + packetsize);




      for (double i = 0; i < nosofpackets + 1; i++) {


        audioSocket.receive(receivePacket);

        byte[] audioData = receivePacket.getData();

        System.out.println("Packet:" + (i + 1));
```

```
            System.out.println(Arrays.toString(audioData));

            bos.write(audioData, 0, audioData.length);


        }

        System.out.println("\nFile saved Successfully!");

        bos.close();

        audioSocket.close();


    }catch (IOException e) {

        e.printStackTrace();

    }

}


private void receiveAVideo() throws IOException {


    DatagramSocket videoSocket = new DatagramSocket(3000);

    int packetsize = 1024;

    FileOutputStream fos = null;

    BufferedOutputStream bos=null;

    try {

        fos = new FileOutputStream("D:\\video from server.mp4");

        bos = new BufferedOutputStream(fos);

        double nosofpackets = Math.ceil(((int) (new File("D:\\Butterfly.mp4")).length()) / packetsize);

        byte[] mybytearray = new byte[packetsize];

        DatagramPacket receivePacket = new DatagramPacket(mybytearray, mybytearray.length);


        System.out.println(nosofpackets + " " + Arrays.toString(mybytearray) + " " + packetsize);


        for (double i = 0; i < nosofpackets + 1; i++) {
                videoSocket.receive(receivePacket);

                byte[] audioData = receivePacket.getData();

                System.out.println("Packet:" + (i + 1));

                System.out.println(Arrays.toString(audioData));

                bos.write(audioData, 0, audioData.length);


            }

            System.out.println("\nFile saved Successfully!");

            bos.close();

            videoSocket.close();

        }catch (IOException e) {

            e.printStackTrace();

        }

    }

}
```

Server side:

```
package com.company;

import java.io.*;

import java.net.DatagramPacket;

import java.net.DatagramSocket;

import java.net.InetAddress;

import java.net.SocketException;

import java.util.Scanner;


public class Server {



    static DatagramSocket serverSocket;
```

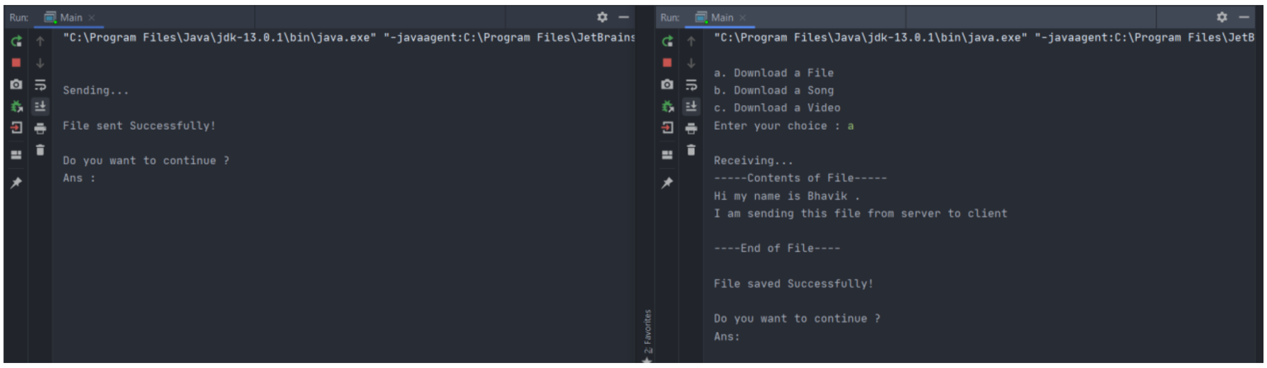
```java
  static byte[] sendData;

  static byte[] receiveData;

  private static InetAddress ip;


  public Server(){


    try {

      serverSocket = new DatagramSocket(6000);

      ip = InetAddress.getLocalHost();

      sendData = new byte[1024];

      receiveData = new byte[1024];


    }catch (Exception e){

      System.out.println("Socket could not be connected");

    }

  }

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


    Server server = new Server();

    char doYouWantToContinue;


    Scanner sc = new Scanner(System.in);


    do {

      DatagramPacket receivePacket = new DatagramPacket(receiveData,receiveData.length);

      serverSocket.receive(receivePacket);

      String userChoice = new
String(receivePacket.getData(),receivePacket.getOffset(),receivePacket.getLength());
```

```
      if (userChoice.equals("a")) {

        server.sendAFile();

      } else if (userChoice.equals("b")) {

        server.sendASong();

      } else if (userChoice.equals("c")) {

        server.sendAVideo();

      } else {

        System.out.println("\nOops! Try again!!!\n");

      }


      System.out.print("\nDo you want to continue ? \nAns : ");

      doYouWantToContinue = sc.next().charAt(0);


    }while(doYouWantToContinue == 'y' || doYouWantToContinue == 'Y');

  }

  private void sendAFile() throws IOException, InterruptedException {


    byte[] b = new byte[10000];

    FileInputStream fos=new FileInputStream("D:\\To Client.txt");

    int i=0;

    System.out.println("\n\nSending...");

    Thread.sleep(2000);

    while(fos.available()!=0)

    {

      b[i]=(byte)fos.read();

      i++;

    }

    fos.close();

    serverSocket.send(new DatagramPacket(b,i,InetAddress.getLocalHost(),1000));
    System.out.println("\nFile sent Successfully!");
  }

  private void sendASong() throws IOException, InterruptedException {

    File myFile = new File("D:\\DEMOSONG.wav");

    BufferedInputStream bis = null;
    try {
      DatagramSocket audioSocket = new DatagramSocket();
      DatagramPacket dp;
      int packetsize = 1024;
      double nosofpackets;
      nosofpackets = Math.ceil(((int) myFile.length()) / packetsize);

      bis = new BufferedInputStream(new FileInputStream(myFile));
      for (double i = 0; i < nosofpackets + 1; i++) {
        byte[] mybytearray = new byte[packetsize];
        bis.read(mybytearray, 0, mybytearray.length);
        System.out.println("Packet:" + (i + 1));
        dp = new DatagramPacket(mybytearray,mybytearray.length, InetAddress.getLocalHost(), 2000);
        Thread.sleep(10L);
        audioSocket.send(dp);
      }

      System.out.println("\nFile sent Successfully!");
   bis.close();
      audioSocket.close();
    } catch (IOException e) {
        e.printStackTrace();

    } catch (InterruptedException e) {

        e.printStackTrace();

    }

}

private void sendAVideo() throws IOException, InterruptedException {


    File myFile = new File("D:\\Butterfly.mp4");


    BufferedInputStream bis = null;

    try {

        DatagramSocket videoSocket = new DatagramSocket();

        DatagramPacket dp;

        int packetsize = 1024;

        double nosofpackets;

        nosofpackets = Math.ceil(((int) myFile.length()) / packetsize);


        bis = new BufferedInputStream(new FileInputStream(myFile));

        for (double i = 0; i < nosofpackets + 1; i++) {


            byte[] mybytearray = new byte[packetsize];

            bis.read(mybytearray, 0, mybytearray.length);

            System.out.println("Packet:" + (i + 1));

            dp = new DatagramPacket(mybytearray,mybytearray.length, InetAddress.getLocalHost(), 3000);

            Thread.sleep(10L);

            videoSocket.send(dp);

        }

        System.out.println("\nFile sent Successfully!");
```

```
            bis.close();

            videoSocket.close();


        } catch (IOException | InterruptedException e) {

            e.printStackTrace();

        }

    }

}
```

OUTPUT:-

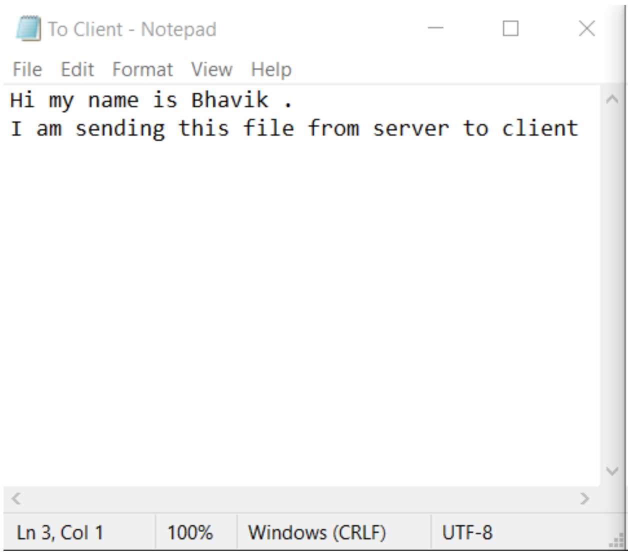
1)Text File Transfer :-



```
Run:   Main
"C:\Program Files\Java\jdk-13.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains


Sending...

File sent Successfully!

Do you want to continue ?
Ans :
```

```
Run:   Main
"C:\Program Files\Java\jdk-13.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetB

a. Download a File
b. Download a Song
c. Download a Video
Enter your choice : a

Receiving...
-----Contents of File-----
Hi my name is Bhavik .
I am sending this file from server to client

----End of File----

File saved Successfully!

Do you want to continue ?
Ans:
```

- After Files Transfer

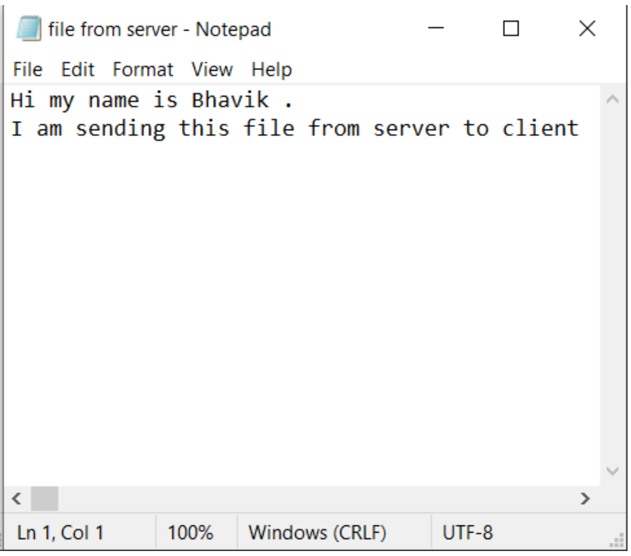


To Client - Notepad

File Edit Format View Help

```
Hi my name is Bhavik .
I am sending this file from server to client
```

Ln 3, Col 1 | 100% | Windows (CRLF) | UTF-8



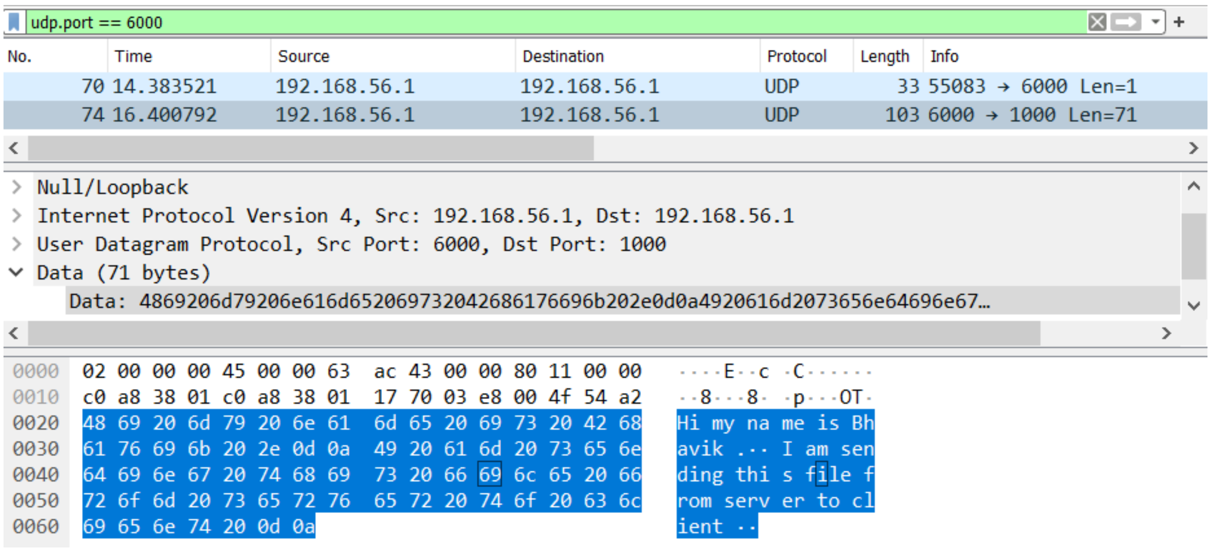
file from server - Notepad

File Edit Format View Help

```
Hi my name is Bhavik .
I am sending this file from server to client
```

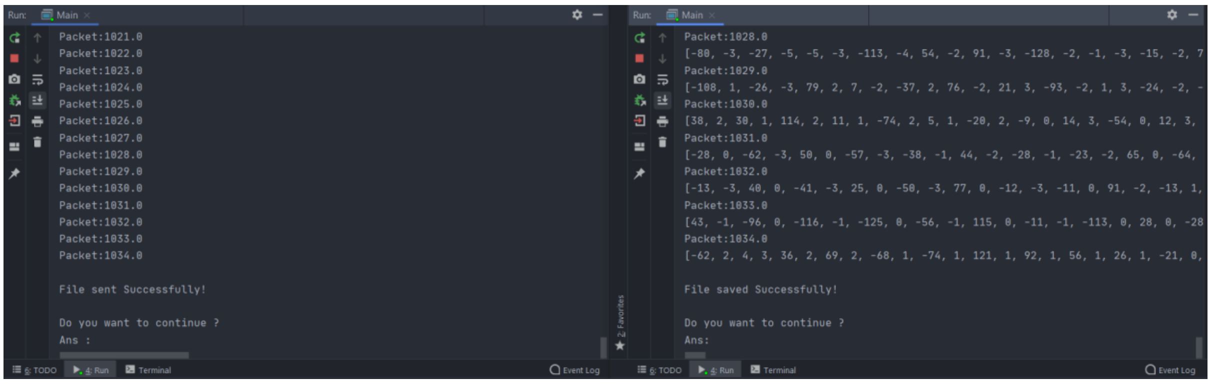
Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

Wireshark:-



```
udp.port == 6000

No.   Time        Source         Destination    Protocol  Length  Info
   70 14.383521   192.168.56.1   192.168.56.1   UDP           33  55083 → 6000 Len=1
   74 16.400792   192.168.56.1   192.168.56.1   UDP          103  6000 → 1000 Len=71

> Null/Loopback
> Internet Protocol Version 4, Src: 192.168.56.1, Dst: 192.168.56.1
> User Datagram Protocol, Src Port: 6000, Dst Port: 1000
v Data (71 bytes)
    Data: 4869206d79206e616d652069732042686176696b202e0d0a4920616d2073656e64696e67…

0000  02 00 00 00 45 00 00 63  ac 43 00 00 80 11 00 00   ····E··c ·C······
0010  c0 a8 38 01 c0 a8 38 01  17 70 03 e8 00 4f 54 a2   ··8···8· ·p···OT·
0020  48 69 20 6d 79 20 6e 61  6d 65 20 69 73 20 42 68   Hi my na me is Bh
0030  61 76 69 6b 20 2e 0d 0a  49 20 61 6d 20 73 65 6e   avik .·· I am sen
0040  64 69 6e 67 20 74 68 69  73 20 66 69 6c 65 20 66   ding thi s file f
0050  72 6f 6d 20 73 65 72 76  65 72 20 74 6f 20 63 6c   rom serv er to cl
0060  69 65 6e 74 20 0d 0a                               ient ··
```

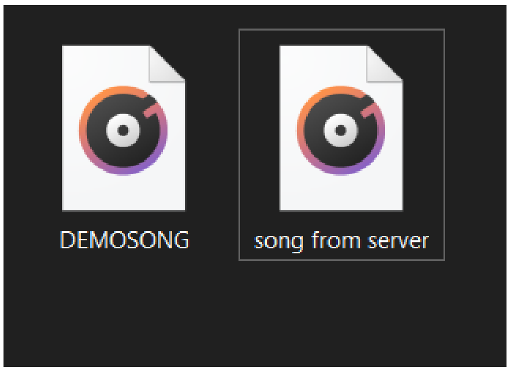
2)Audio File Transfer :-



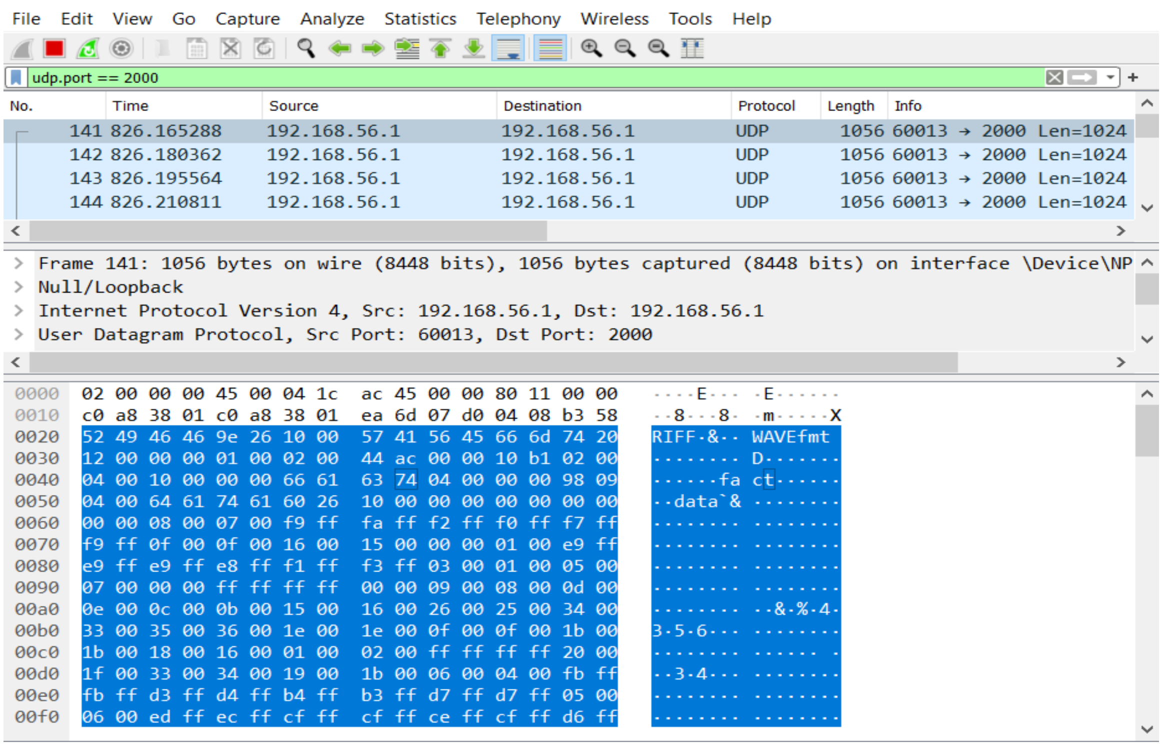
```
Packet:1021.0
Packet:1022.0
Packet:1023.0
Packet:1024.0
Packet:1025.0
Packet:1026.0
Packet:1027.0
Packet:1028.0
Packet:1029.0
Packet:1030.0
Packet:1031.0
Packet:1032.0
Packet:1033.0
Packet:1034.0

File sent Successfully!

Do you want to continue ?
Ans :
```

```
Packet:1028.0
[-80, -3, -27, -5, -5, -3, -113, -4, 54, -2, 91, -3, -128, -2, -1, -3, -15, -2, 7
Packet:1029.0
[-108, 1, -26, -3, 79, 2, 7, -2, -37, 2, 76, -2, 21, 3, -93, -2, 1, 3, -24, -2, -
Packet:1030.0
[38, 2, 30, 1, 114, 2, 11, 1, -74, 2, 5, 1, -20, 2, -9, 0, 14, 3, -54, 0, 12, 3,
Packet:1031.0
[-28, 0, -62, -3, 50, 0, -57, -3, -38, -1, 44, -2, -28, -1, -23, -2, 65, 0, -64,
Packet:1032.0
[-13, -3, 40, 0, -41, -3, 25, 0, -50, -3, 77, 0, -12, -3, -11, 0, 91, -2, -13, 1,
Packet:1033.0
[43, -1, -96, 0, -116, -1, -125, 0, -56, -1, 115, 0, -11, -1, -113, 0, 28, 0, -28
Packet:1034.0
[-62, 2, 4, 3, 36, 2, 69, 2, -68, 1, -74, 1, 121, 1, 92, 1, 56, 1, 26, 1, -21, 0,

File saved Successfully!

Do you want to continue ?
Ans:
```

- After Audio Transfer

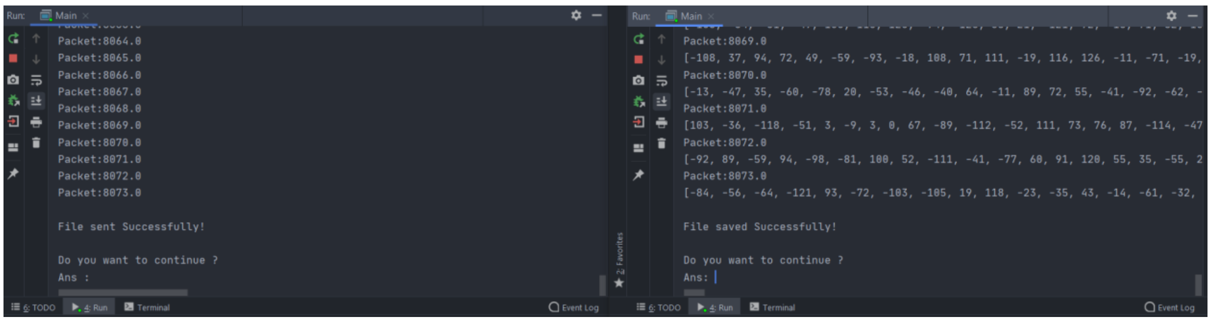


Wireshark:-


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
udp.port == 2000

No.   Time         Source          Destination     Protocol  Length  Info
  141 826.165288   192.168.56.1    192.168.56.1    UDP       1056 60013 → 2000 Len=1024
  142 826.180362   192.168.56.1    192.168.56.1    UDP       1056 60013 → 2000 Len=1024
  143 826.195564   192.168.56.1    192.168.56.1    UDP       1056 60013 → 2000 Len=1024
  144 826.210811   192.168.56.1    192.168.56.1    UDP       1056 60013 → 2000 Len=1024

> Frame 141: 1056 bytes on wire (8448 bits), 1056 bytes captured (8448 bits) on interface \Device\NP
> Null/Loopback
> Internet Protocol Version 4, Src: 192.168.56.1, Dst: 192.168.56.1
> User Datagram Protocol, Src Port: 60013, Dst Port: 2000

0000  02 00 00 00 45 00 04 1c  ac 45 00 00 80 11 00 00   ····E··· ·E······
0010  c0 a8 38 01 c0 a8 38 01  ea 6d 07 d0 04 08 b3 58   ··8···8· ·m·····X
0020  52 49 46 46 9e 26 10 00  57 41 56 45 66 6d 74 20   RIFF·&·· WAVEfmt 
0030  12 00 00 00 01 00 02 00  44 ac 00 00 10 b1 02 00   ········ D·······
0040  04 00 10 00 00 00 66 61  63 74 04 00 00 00 98 09   ······fa ct······
0050  04 00 64 61 74 61 60 26  10 00 00 00 00 00 00 00   ··data`& ········
0060  00 00 08 00 07 00 f9 ff  fa ff f2 ff f0 ff f7 ff   ········ ········
0070  f9 ff 0f 00 0f 00 16 00  15 00 00 00 01 00 e9 ff   ········ ········
0080  e9 ff e9 ff e8 ff f1 ff  f3 ff 03 00 01 00 05 00   ········ ········
0090  07 00 00 00 ff ff ff ff  00 00 09 00 08 00 0d 00   ········ ········
00a0  0e 00 0c 00 0b 00 15 00  16 00 26 00 25 00 34 00   ········ ··&·%·4·
00b0  33 00 35 00 36 00 1e 00  1e 00 0f 00 0f 00 1b 00   3·5·6··· ········
00c0  1b 00 18 00 16 00 01 00  02 00 ff ff ff ff 20 00   ········ ······ ·
00d0  1f 00 33 00 34 00 19 00  1b 00 06 00 04 00 fb ff   ··3·4··· ········
00e0  fb ff d3 ff d4 ff b4 ff  b3 ff d7 ff d7 ff 05 00   ········ ········
00f0  06 00 ed ff ec ff cf ff  cf ff ce ff cf ff d6 ff   ········ ········
```

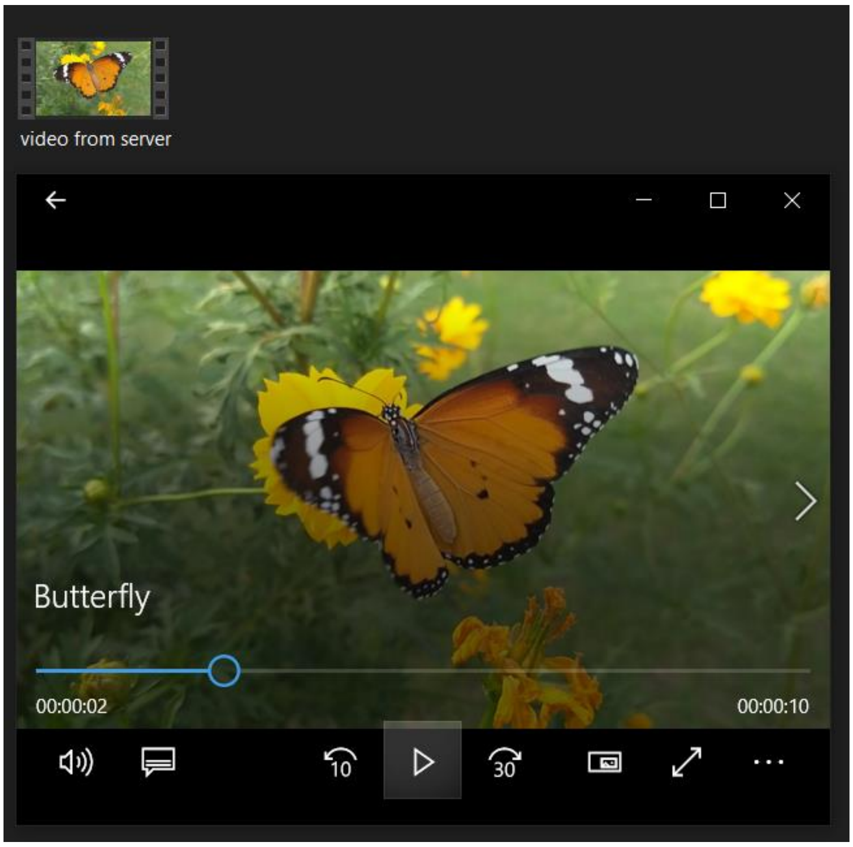
3)Video file transfer :-



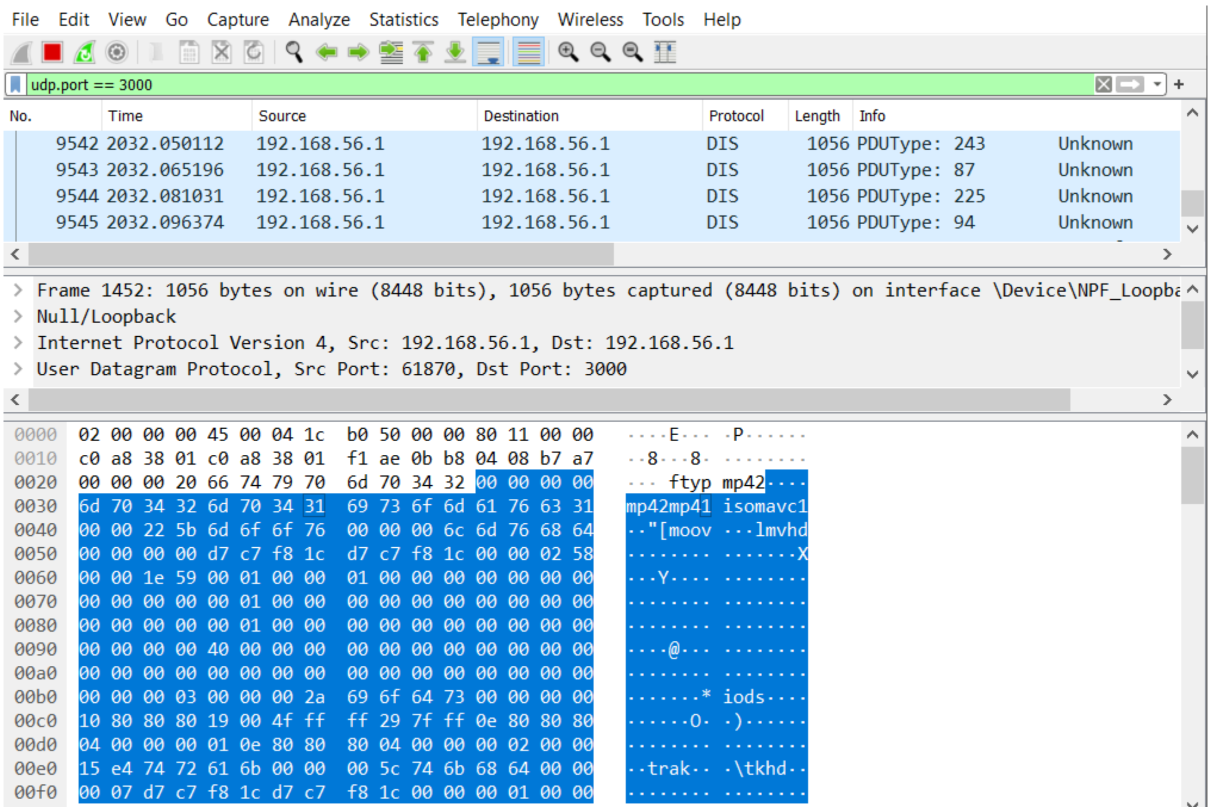
```
Packet:8064.0
Packet:8065.0
Packet:8066.0
Packet:8067.0
Packet:8068.0
Packet:8069.0
Packet:8070.0
Packet:8071.0
Packet:8072.0
Packet:8073.0

File sent Successfully!

Do you want to continue ?
Ans :
```

```
Packet:8069.0
[-108, 37, 94, 72, 49, -59, -93, -18, 108, 71, 111, -19, 116, 126, -11, -71, -19,
Packet:8070.0
[-13, -47, 35, -60, -78, 20, -53, -46, -40, 64, -11, 89, 72, 55, -41, -92, -62, -
Packet:8071.0
[103, -36, -118, -51, 3, -9, 3, 0, 67, -89, -112, -52, 111, 73, 76, 87, -114, -47
Packet:8072.0
[-92, 89, -59, 94, -98, -81, 100, 52, -111, -41, -77, 60, 91, 120, 55, 35, -55, 2
Packet:8073.0
[-84, -56, -64, -121, 93, -72, -103, -105, 19, 118, -23, -35, 43, -14, -61, -32,

File saved Successfully!

Do you want to continue ?
Ans:
```

- After video Transfer



Wireshark:-


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
udp.port == 3000

No.   Time          Source          Destination     Protocol  Length Info
 9542 2032.050112   192.168.56.1    192.168.56.1    DIS       1056 PDUType: 243      Unknown
 9543 2032.065196   192.168.56.1    192.168.56.1    DIS       1056 PDUType: 87       Unknown
 9544 2032.081031   192.168.56.1    192.168.56.1    DIS       1056 PDUType: 225      Unknown
 9545 2032.096374   192.168.56.1    192.168.56.1    DIS       1056 PDUType: 94       Unknown

> Frame 1452: 1056 bytes on wire (8448 bits), 1056 bytes captured (8448 bits) on interface \Device\NPF_Loopba
> Null/Loopback
> Internet Protocol Version 4, Src: 192.168.56.1, Dst: 192.168.56.1
> User Datagram Protocol, Src Port: 61870, Dst Port: 3000

0000  02 00 00 00 45 00 04 1c  b0 50 00 00 80 11 00 00   ····E··· ·P······
0010  c0 a8 38 01 c0 a8 38 01  f1 ae 0b b8 04 08 b7 a7   ··8···8· ········
0020  00 00 00 20 66 74 79 70  6d 70 34 32 00 00 00 00   ··· ftyp mp42····
0030  6d 70 34 32 6d 70 34 31  69 73 6f 6d 61 76 63 31   mp42mp41 isomavc1
0040  00 00 22 5b 6d 6f 6f 76  00 00 00 6c 6d 76 68 64   ··"[moov ···lmvhd
0050  00 00 00 00 d7 c7 f8 1c  d7 c7 f8 1c 00 00 02 58   ········ ·······X
0060  00 00 1e 59 00 01 00 00  01 00 00 00 00 00 00 00   ···Y···· ········
0070  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00   ········ ········
0080  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00   ········ ········
0090  00 00 00 00 40 00 00 00  00 00 00 00 00 00 00 00   ····@··· ········
00a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00b0  00 00 00 03 00 00 00 2a  69 6f 64 73 00 00 00 00   ·······* iods····
00c0  10 80 80 80 19 00 4f ff  ff 29 7f ff 0e 80 80 80   ······O· ·)······
00d0  04 00 00 00 01 0e 80 80  80 04 00 00 00 02 00 00   ········ ········
00e0  15 e4 74 72 61 6b 00 00  00 5c 74 6b 68 64 00 00   ··trak·· ·\tkhd··
00f0  00 07 d7 c7 f8 1c d7 c7  f8 1c 00 00 00 01 00 00   ········ ········
```