Name – Pranav Tiwari
Batch – Cse
Roll no. - 201127

Network Laboratory

Programming Assignment-4

Ques:

Simple client-server communication (NS-3 Warmup)

- **Level**: Introductory
- **Expected learning outcome**: NS-3 simulation basics. Basic client server paradigm. Reading pcap traces.
- **Experiment**:
  1. Create a simple topology of two nodes (Node1, Node2) separated by a point-to-point link.
  2. Setup a UdpClient on one Node1 and a UdpServer on Node2. Let it be of a fixed data rate Rate1.
  3. Start the client application, and measure end to end throughput whilst varying the latency of the link.
  4. Now add another client application to Node1 and a server instance to Node2. What do you need to configure to ensure that there is no conflict?
  5. Repeat step 3 with the extra client and server application instances. Show screenshots of pcap traces which indicate that delivery is made to the appropriate server instance.

Source Code

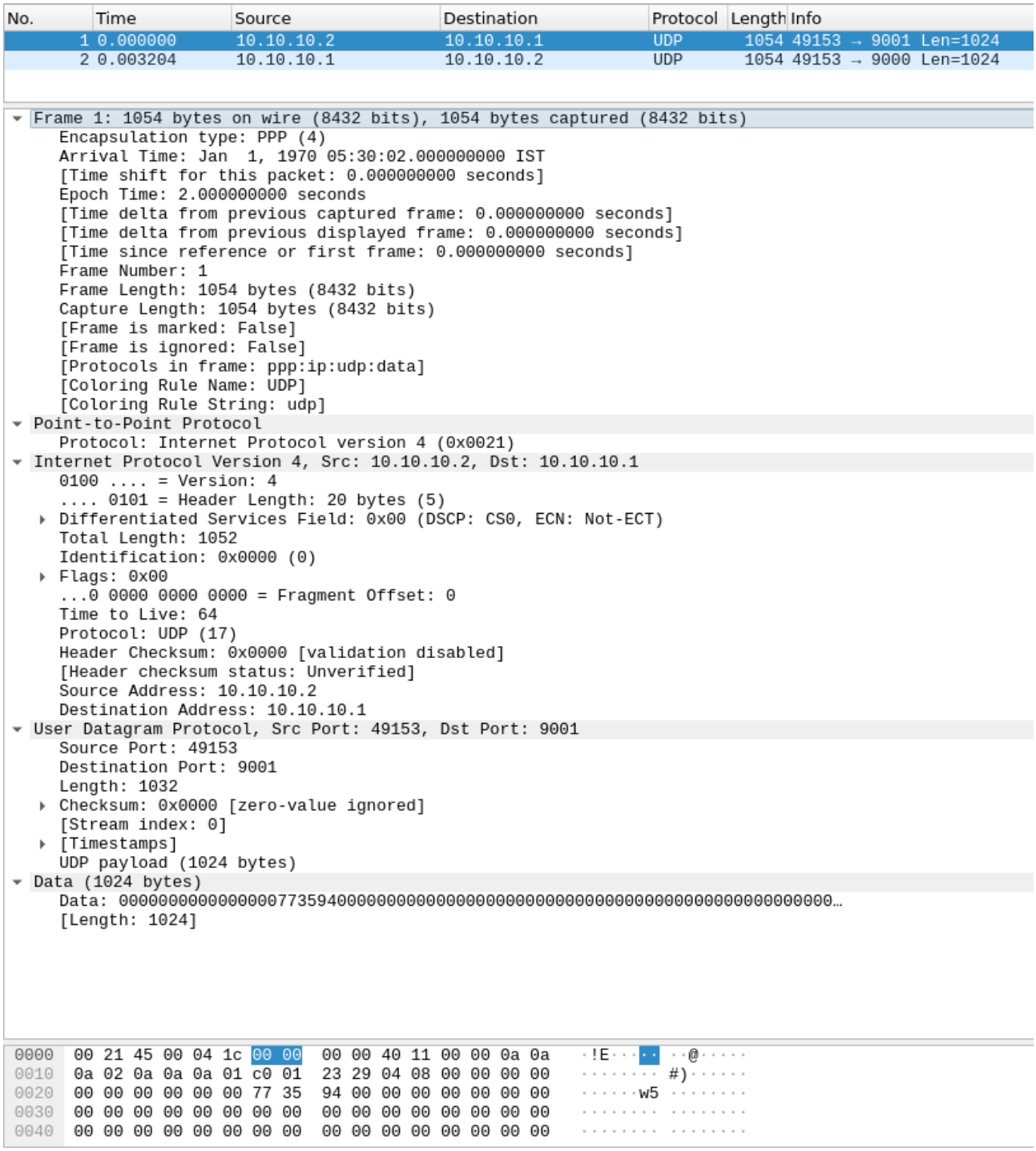
q1.cc

```
#include "ns3/applications-module.h"
#include "ns3/core-module.h"
#include "ns3/internet-module.h"
#include "ns3/network-module.h"
#include "ns3/point-to-point-module.h"

using namespace ns3;

NS_LOG_COMPONENT_DEFINE("Client_Server_Communication");

int main(int argc, char* argv[])
{
CommandLine cmd;
cmd.Parse(argc, argv);

Time::SetResolution(Time::NS);

NodeContainer nodes;
nodes.Create(2);

PointToPointHelper pointToPoint;
pointToPoint.SetDeviceAttribute("DataRate", StringValue("7Mbps"));
pointToPoint.SetChannelAttribute("Delay", StringValue("2ms"));

NetDeviceContainer devices;
devices = pointToPoint.Install(nodes);

InternetStackHelper stack;
stack.Install(nodes);
```

```
Ipv4AddressHelper address;
address.SetBase("10.10.10.0", "255.255.255.0");
Ipv4InterfaceContainer interfaces = address.Assign(devices);

UdpServerHelper server1(9000);
ApplicationContainer serverApps1 = server1.Install(nodes.Get(1));
serverApps1.Start(Seconds(1.0));
serverApps1.Stop(Seconds(10.0));

UdpClientHelper client1(interfaces.GetAddress(1), 9000);
client1.SetAttribute("MaxPackets", UintegerValue(1));
client1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
client1.SetAttribute("PacketSize", UintegerValue(1024));

UdpServerHelper server2(9001);
ApplicationContainer serverApps2 = server2.Install(nodes.Get(0));
serverApps2.Start(Seconds(1.0));
serverApps2.Stop(Seconds(10.0));

UdpClientHelper client2(interfaces.GetAddress(0), 9001);
client2.SetAttribute("MaxPackets", UintegerValue(1));
client2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
client2.SetAttribute("PacketSize", UintegerValue(1024));

ApplicationContainer clientApps;
clientApps.Add(client1.Install(nodes.Get(0)));
clientApps.Add(client2.Install(nodes.Get(1)));
clientApps.Start(Seconds(2.0));

pointToPoint.EnablePcapAll("simple-client-server-communication");

Simulator::Stop(Seconds(10.0));
Simulator::Run();
Simulator::Destroy();

return 0;
}
```

Terminal:

Wire-shark:



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.10.10.2	10.10.10.1	UDP	1054	49153 → 9001 Len=1024
2	0.003204	10.10.10.1	10.10.10.2	UDP	1054	49153 → 9000 Len=1024

```
▾ Frame 1: 1054 bytes on wire (8432 bits), 1054 bytes captured (8432 bits)
    Encapsulation type: PPP (4)
    Arrival Time: Jan  1, 1970 05:30:02.000000000 IST
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 2.000000000 seconds
    [Time delta from previous captured frame: 0.000000000 seconds]
    [Time delta from previous displayed frame: 0.000000000 seconds]
    [Time since reference or first frame: 0.000000000 seconds]
    Frame Number: 1
    Frame Length: 1054 bytes (8432 bits)
    Capture Length: 1054 bytes (8432 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: ppp:ip:udp:data]
    [Coloring Rule Name: UDP]
    [Coloring Rule String: udp]
▾ Point-to-Point Protocol
    Protocol: Internet Protocol version 4 (0x0021)
▾ Internet Protocol Version 4, Src: 10.10.10.2, Dst: 10.10.10.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1052
    Identification: 0x0000 (0)
  ▸ Flags: 0x00
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: UDP (17)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 10.10.10.2
    Destination Address: 10.10.10.1
▾ User Datagram Protocol, Src Port: 49153, Dst Port: 9001
    Source Port: 49153
    Destination Port: 9001
    Length: 1032
  ▸ Checksum: 0x0000 [zero-value ignored]
    [Stream index: 0]
  ▸ [Timestamps]
    UDP payload (1024 bytes)
▾ Data (1024 bytes)
    Data: 00000000000000007735940000000000000000000000000000000000000000000000000000…
    [Length: 1024]
```

```
0000  00 21 45 00 04 1c 00 00  00 00 40 11 00 00 0a 0a   ·!E·····  ··@·····
0010  0a 02 0a 0a 0a 01 c0 01  23 29 04 08 00 00 00 00   ········  #)······
0020  00 00 00 00 00 00 77 35  94 00 00 00 00 00 00 00   ······w5  ········
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········  ········
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········  ········
```