

Network Simulation is done in ns-3 in this lab.

In first.cc file, two nodes are set to transfer data at a rate of 8Mbps, and transmission delay is 4ms. So, I set

```
pointToPoint.SetDeviceAttribute("DataRate",StringValue("8Mbps"));
pointToPoint.SetChannelAttribute("Delay",StringValue("4ms"));
```

IP addresses are 192.168.40.0 and 192.168.40.24.

```
Address.SetBase("192.168.40.0","192.168.40.24");
```

UdpEchoServer is on port 93. The packet size is 256 Bytes.

```
UdpEchoServerHelper echoServer(93);
echoClient.SetAttribute("PacketSize",UIntegerValue(256));
```

To output the .xml file for NetAnim, the codes as the following are added to first.cc:

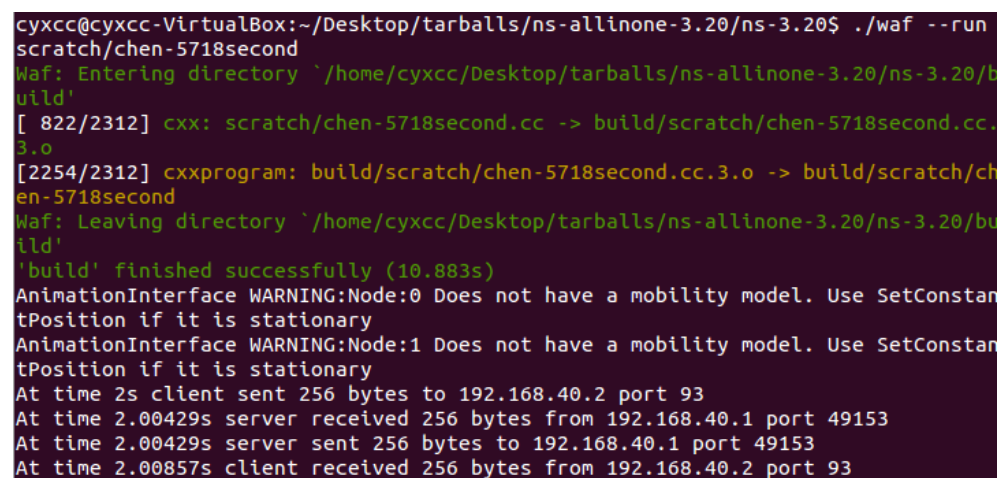
```
AnimationInterface anim("chen-5718second.xml");
Anim.SetConstantPosition(nodes.Get(0),1.0,2.0);
Anim.SetConstantPosition(nodes.Get(1),2.0,3.0);
```

The coordinates of the two nodes are set. One is 1.0 on x, 2.0 on y. The other is 2.0 on x, 3.0 on y.

To enable ASCII and pcap tracing, the codes as the following are added to create .tr file and pcap file:

```
AsciiTraceHelper ascii;
pointToPoint.EnableAsciiAll(ascii.CreateFileStream("chen-5718second.tr"));
pointToPoint.EnablePcapAll("chen-5718second");
```

Then to simulate first.cc, the command `./waf --run scratch/first` is implemented in Ubuntu. The result is shown as the following:



```
cyxcc@cyxcc-VirtualBox:~/Desktop/tarballs/ns-allinone-3.20/ns-3.20$ ./waf --run
scratch/chen-5718second
waf: Entering directory `/home/cyxcc/Desktop/tarballs/ns-allinone-3.20/ns-3.20/build'
[ 822/2312] cxx: scratch/chen-5718second.cc -> build/scratch/chen-5718second.cc.3.o
[2254/2312] cxxprogram: build/scratch/chen-5718second.cc.3.o -> build/scratch/chen-5718second
waf: Leaving directory `/home/cyxcc/Desktop/tarballs/ns-allinone-3.20/ns-3.20/build'
'build' finished successfully (10.883s)
AnimationInterface WARNING:Node:0 Does not have a mobility model. Use SetConstantPosition if it is stationary
AnimationInterface WARNING:Node:1 Does not have a mobility model. Use SetConstantPosition if it is stationary
At time 2s client sent 256 bytes to 192.168.40.2 port 93
At time 2.00429s server received 256 bytes from 192.168.40.1 port 49153
At time 2.00429s server sent 256 bytes to 192.168.40.1 port 49153
At time 2.00857s client received 256 bytes from 192.168.40.2 port 93
```

The screenshots of pcap tracing in WireShark of the two nodes are shown as:

Node 1:

[illegible]

## Node 0:

The image shows a Wireshark packet capture analysis of a UDP packet. The packet list at the top shows a single packet (No. 1) of type UDP, source 192.168.40.1, destination 192.168.40.2, length 286, and info "286 dcp(93) Len=256". The packet details pane shows the packet structure: Ethernet II (II), Internet Protocol Version 4 (IPv4), and User Datagram Protocol (UDP). The packet bytes pane shows the raw data in hexadecimal and ASCII. The packet is a UDP packet from 192.168.40.1 to 192.168.40.2, length 286, and info "286 dcp(93) Len=256".

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.40.1	192.168.40.2	UDP	286	49153 → dcp(93) Len=256
2	0.000572	192.168.40.2	192.168.40.1	UDP	286	dcp(93) → 49153 Len=256

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: 286 bytes (2288 bits)  
Capture Length: 286 bytes (2288 bits)  
[Frame is marked: False]  
[Frame is ignored: False]  
[Protocols in frame: ppp-ipv4:udp:data]  
[Coloring Rule Name: UDP]  
[Coloring Rule String: udp]  
Point-to-Point Protocol  
Internet Protocol Version 4, Src: 192.168.40.1, Dst: 192.168.40.2

0000 00 21 00 00 01 1c 00 00 00 00 40 11 00 00 c0 a0 -1.....  
0010 00 00 00 00 00 00 c0 01 00 5d 01 08 00 00 00 00 [.....]  
0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
0030 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 .....  
0050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
0070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
0080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
0090 00 00 00 00 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 .....  
00b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
00c0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
00d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
00e0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
00f0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

Internet Protocol Version 4 (Ip), 20 bytes  
Packets: 2 · BinLoad: 2 (100.00)  
Profile Info

Other files such as flow diagrams, output files are in the corresponding folders.