

5. Simulate to implement the connection of 2 nodes and 4 router such that the extremes nodes act as client and server using Point-to-Point protocol. Apply NetAnim software to demonstrate the scenario graphically. Analyze packet parameters by creating trace file using Ascii trace metrics.

```

1 #include "ns3/core-module.h"
2 #include "ns3/network-module.h"
3 #include "ns3/internet-module.h"
4 #include "ns3/point-to-point-module.h"
5 #include "ns3/applications-module.h"
6 #include "ns3/metadata-module.h"
7 #include "ns3/ipv4-global-routing-helper.h"
8 using namespace ns3;
9 NS_LOG_COMPONENT_DEFINE ("PointToPointPPPEXample");
10 int main (int argc, char *argv[])
11 {
12     CommandLine cmd (_FILE_);
13     cmd.Parse (argc, argv);
14     Time::SetResolution (Time::NS);
15     LogComponentEnable ("UdpEchoClientApplication", LOG_LEVEL_INFO);
16     LogComponentEnable ("UdpEchoServerApplication", LOG_LEVEL_INFO);
17
18     NodeContainer nodes;
19     nodes.Create (2); // Client (n0) and Server (n1)
20     NodeContainer routers;
21     routers.Create (4); // Routers r0, r1, r2, r3
22
23     PointToPointHelper pointToPoint;
24     pointToPoint.SetDeviceAttribute ("DataRate", StringValue ("5Mbps"));
25     pointToPoint.SetChannelAttribute ("Delay", StringValue ("2ms"));
26
27     NetDeviceContainer devices0 = pointToPoint.Install (nodes.Get(0), routers.Get(0)); // n0-r0
28     NetDeviceContainer devices1 = pointToPoint.Install (routers.Get(0), routers.Get(1)); // r0-r1
29     NetDeviceContainer devices2 = pointToPoint.Install (routers.Get(1), routers.Get(2)); // r1-r2
30     NetDeviceContainer devices3 = pointToPoint.Install (routers.Get(2), routers.Get(3)); // r2-r3
31     NetDeviceContainer devices4 = pointToPoint.Install (routers.Get(3), nodes.Get(1)); // r3-n1
32
33     InternetStackHelper stack;
34     stack.Install (nodes);
35     stack.Install (routers);
36
37     Ipv4AddressHelper address;
38     address.SetBase ("10.1.1.0", "255.255.255.0");
39     Ipv4InterfaceContainer interfaces0 = address.Assign (devices0);
40     address.SetBase ("10.1.2.0", "255.255.255.0");
41     Ipv4InterfaceContainer interfaces1 = address.Assign (devices1);
42     address.SetBase ("10.1.3.0", "255.255.255.0");
43     Ipv4InterfaceContainer interfaces2 = address.Assign (devices2);
44     address.SetBase ("10.1.4.0", "255.255.255.0");
45     Ipv4InterfaceContainer interfaces3 = address.Assign (devices3);
46     address.SetBase ("10.1.5.0", "255.255.255.0");
47     Ipv4InterfaceContainer interfaces4 = address.Assign (devices4);
48
49     Ipv4GlobalRoutingHelper::PopulateRoutingTables ();
50
51     UdpEchoServerHelper echoServer (0);
52     ApplicationContainer serverApps = echoServer.Install (nodes.Get (1));
53     serverApps.Start (Seconds (1.0));
54     serverApps.Stop (Seconds (10.0));

```

```

55
56     UdpEchoClientHelper echoClient (interfaces4.GetAddress (1), 9);
57     echoClient.SetAttribute ("MaxPackets", UIntegerValue (1));
58     echoClient.SetAttribute ("Interval", TimeValue (Seconds (1.0)));
59     echoClient.SetAttribute ("PacketSize", UIntegerValue (1024));
60     ApplicationContainer clientApps = echoClient.Install (nodes.Get (0));
61     clientApps.Start (Seconds (2.0));
62     clientApps.Stop (Seconds (10.0));
63
64     system ("mkdir -p output");
65     Ptr<Node> n0 = nodes.Get(0);
66     Ptr<Node> n1 = nodes.Get(1);
67     Ptr<Node> r0 = routers.Get(0);
68     Ptr<Node> r1 = routers.Get(1);
69     Ptr<Node> r2 = routers.Get(2);
70     Ptr<Node> r3 = routers.Get(3);
71     AnimationInterface anim ("fish.xml");
72     anim.SetConstantPosition (n0, 100, 200);
73     anim.SetConstantPosition (r0, 200, 200);
74     anim.SetConstantPosition (r1, 300, 200);
75     anim.SetConstantPosition (r2, 400, 200);
76     anim.SetConstantPosition (r3, 500, 200);
77     anim.SetConstantPosition (n1, 600, 200);
78
79     AsciiTraceHelper ascii;
80     pointToPoint.EnableAsciiAll (ascii.CreateFileStream ("output/5th.tr"));
81     Simulator::Run ();
82     Simulator::Destroy ();
83     return 0;
84 }

```

