PERCOBAAN 3

SIMULASI WIFI (Wireless AP-Client)

4.1 Tujuan :

Setelah melaksanakan praktikum ini mahasiswa diharapkan mampu :

- Mendesain dan memprogram jaringan WIFI menggunakan NS-3.
- Memahami dan membandingkan hasil trace output jaringan WIFI pada NS-3 menggunakan Tracemetrics.

4.2 Peralatan :

- 1 PC dilengkapi dengan OS Ubuntu 16.04
- Software NS-3 versi 3.25
- Software Tracemetrics

4.3 Teori :

Dalam simulasi Wireless Fidelity atau Wifi kita perlu melihat terlebih dahulu parameter yang digunakan untuk standar Wifi itu sendiri. Salah satu yang paling krusial adalah penggunaan Standar 802.11. Kemudian pada simulasi Wifi ini terdapat dua jenis node yaitu Station (Sta) dan Access Point (AP).

Selain itu, pada simulasi Wifi ini objek helper yang digunakan adalah WifiHelper dan WifiMacHelper yang dibuat untuk mengatur beberapa atribut lain.

Sama halnya seperti pada simulasi sebelumnya, pada simulasi Wifi ini output visualnya dapat dilihat pada Netanim. Skrip konfigurasi Netanim untuk simulasi ini adalah seperti yang ditunjukkan di bawah ini.

```
// Animation configuration lines
AnimationInterface anim ("mythird.xml");
anim.EnablePacketMetadata(true);
// End of animation configuration
```

Sedangkan, untuk melihat trace outputnya, masih sama yaitu dengan menggunakan Tracemetrics, dimana skrip konfigurasi ascii tracing yang perlu ditambahkan sebelumnya adalah seperti berikut.

```
//Ascii Format Tracing
AsciiTraceHelper ascii;
phy.EnableAsciiAll(ascii.CreateFileStream("mythird.tr"));
```

4.4 Prosedur Percobaan :

1. Buka direktori *ns-allinone-3.25/ns-3.25/scratch* melalui terminal.
2. Tuliskan perintah gedit dengan nama file *mythird.cc* seperti berikut.

   gedit mythird.cc

3. Tuliskan script di bawah ini, dimana script ini merupakan simulasi sederhana Wifi empat node.

```
/* Ilustrasi topologi
 *
 * Wifi 10.1.3.0
 *             AP
 *    *  *  *  *
 *    |  |  |  |
 *   n3  n2 n1 n0
 */

 #include "ns3/core-module.h"
 #include "ns3/point-to-point-module.h"
 #include "ns3/network-module.h"
 #include "ns3/applications-module.h"
 #include "ns3/wifi-module.h"
 #include "ns3/mobility-module.h"
 #include "ns3/csma-module.h"
 #include "ns3/internet-module.h"
 #include "ns3/netanim-module.h" // entered for animation
configuration and output file

 using namespace ns3;

 NS_LOG_COMPONENT_DEFINE ("MyThirdScript");

 int
 main (int argc, char *argv[])
 {
   bool verbose = true;
   uint32_t nWifi = 3;
   bool tracing = false;

   CommandLine cmd;

   cmd.AddValue ("nWifi", "Number of wifi STA devices", nWifi);
```

```
   cmd.AddValue ("verbose", "Tell echo applications to log if
true", verbose);
   cmd.AddValue ("tracing", "Enable pcap tracing", tracing);

   cmd.Parse (argc,argv);

   if (verbose)
   {
     LogComponentEnable ("UdpEchoClientApplication",
LOG_LEVEL_INFO);
     LogComponentEnable ("UdpEchoServerApplication",
LOG_LEVEL_INFO);
   }

   NodeContainer wifiStaNodes;
   wifiStaNodes.Create (nWifi);

   NodeContainer wifiApNode;
   wifiApNode.Create(1);

   YansWifiChannelHelper channel = YansWifiChannelHelper::Default
();
   YansWifiPhyHelper phy = YansWifiPhyHelper::Default ();
   phy.SetChannel (channel.Create ());

   WifiHelper wifi;
   wifi.SetRemoteStationManager ("ns3::AarfWifiManager");

   WifiMacHelper mac;
   Ssid ssid = Ssid ("ns-3-ssid");
   mac.SetType ("ns3::StaWifiMac", "Ssid", SsidValue (ssid),
"ActiveProbing", BooleanValue (false));

   NetDeviceContainer staDevices;
   staDevices = wifi.Install (phy, mac, wifiStaNodes);

   mac.SetType ("ns3::ApWifiMac", "Ssid", SsidValue (ssid));

   NetDeviceContainer apDevices;
   apDevices = wifi.Install (phy, mac, wifiApNode);

   MobilityHelper mobility;

   mobility.SetPositionAllocator ("ns3::GridPositionAllocator",
"MinX", DoubleValue (0.0), "MinY", DoubleValue (0.0), "DeltaX",
DoubleValue (5.0), "DeltaY", DoubleValue (10.0), "GridWidth",
UintegerValue (3), "LayoutType", StringValue ("RowFirst"));

   mobility.SetMobilityModel ("ns3::RandomWalk2dMobilityModel",
"Bounds", RectangleValue (Rectangle (-50, 50, -50, 50)));
   mobility.Install (wifiStaNodes);

   mobility.SetMobilityModel
("ns3::ConstantPositionMobilityModel");
   mobility.Install (wifiApNode);

   InternetStackHelper stack;
```

```
   stack.Install (wifiApNode);
   stack.Install (wifiStaNodes);

   Ipv4AddressHelper address;

   address.SetBase ("10.1.1.0", "255.255.255.0");
   Ipv4InterfaceContainer wifiStaInterfaces;
   wifiStaInterfaces = address.Assign (staDevices);
   address.Assign (apDevices);

   UdpEchoServerHelper echoServer (9);

   ApplicationContainer serverApps = echoServer.Install
(wifiStaNodes.Get(0));
   serverApps.Start (Seconds (1.0));
   serverApps.Stop (Seconds (10.0));

   UdpEchoClientHelper echoClient (wifiStaInterfaces.GetAddress
(2), 9);
   echoClient.SetAttribute ("MaxPackets", UintegerValue (10));
   echoClient.SetAttribute ("Interval", TimeValue (Seconds
(0.5)));
   echoClient.SetAttribute ("PacketSize", UintegerValue (1024));

   ApplicationContainer clientApps = echoClient.Install
(wifiStaNodes.Get (2));
   clientApps.Start (Seconds (2.0));
   clientApps.Stop (Seconds (10.0));

   // Ipv4GlobalRoutingHelper::PopulateRoutingTables ();

   Simulator::Stop (Seconds (10.0));

   if (tracing == true)
   {
     phy.EnablePcap ("mythird", apDevices.Get (0));;
   }

   // Animation configuration lines
   AnimationInterface anim ("mythird.xml");
   anim.EnablePacketMetadata(true);
   // End of animation configuration


   //Ascii Format Tracing
   AsciiTraceHelper ascii;
   phy.EnableAsciiAll(ascii.CreateFileStream("mythird.tr"));


   Simulator::Run ();
   Simulator::Destroy ();
   return 0;
 }
```

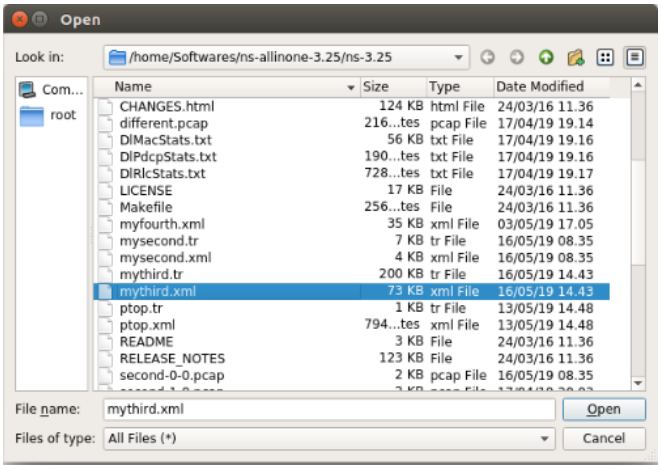
4. Buka direktori *ns-allinone-3.25/ns-3.25* dan jalankan program tersebut menggunakan *waf* dengan menuliskan perintah seperti di bawah ini.

```
root@dina-Inspiron-5458:/home/Softwares/ns-allinone-3.25/ns-3.25# ./waf --run scratch/mythird
Waf: Entering directory `/home/Softwares/ns-allinone-3.25/ns-3.25/build'
Waf: Leaving directory `/home/Softwares/ns-allinone-3.25/ns-3.25/build'
Build commands will be stored in build/compile_commands.json
'build' finished successfully (2.728s)
At time 2s client sent 1024 bytes to 10.1.1.3 port 9
At time 2.5s client sent 1024 bytes to 10.1.1.3 port 9
At time 3s client sent 1024 bytes to 10.1.1.3 port 9
At time 3.5s client sent 1024 bytes to 10.1.1.3 port 9
At time 4s client sent 1024 bytes to 10.1.1.3 port 9
At time 4.5s client sent 1024 bytes to 10.1.1.3 port 9
At time 5s client sent 1024 bytes to 10.1.1.3 port 9
At time 5.5s client sent 1024 bytes to 10.1.1.3 port 9
At time 6s client sent 1024 bytes to 10.1.1.3 port 9
At time 6.5s client sent 1024 bytes to 10.1.1.3 port 9
root@dina-Inspiron-5458:/home/Softwares/ns-allinone-3.25/ns-3.25#
```

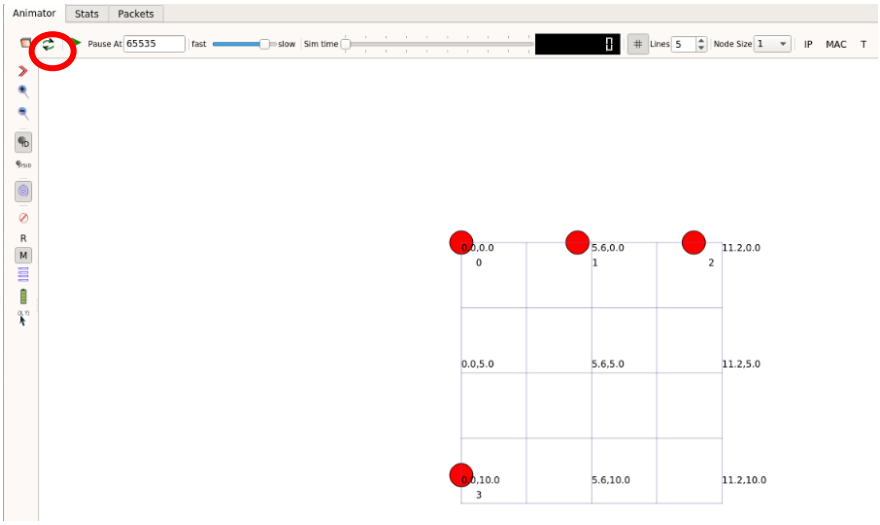
5. Kemudian, buka direktori *ns-allinone-3.25/netanim-3.107* dan buka Netanim dengan menuliskan command berikut.

```
root@dina-Inspiron-5458:/home/Softwares/ns-allinone-3.25/netanim-3.107# ./NetAnim
```

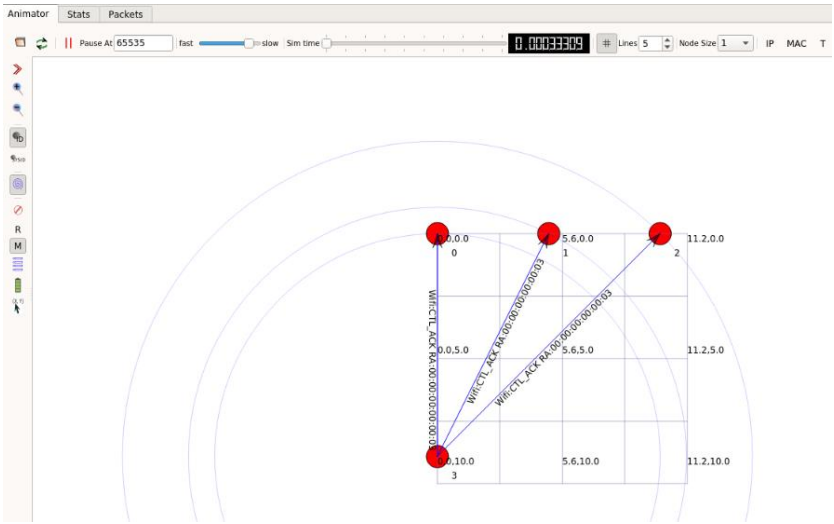
6. Buka file *mythird.xml* dengan cara klik icon Open File di pojok kiri atas dan akan muncul tampilan berikut.

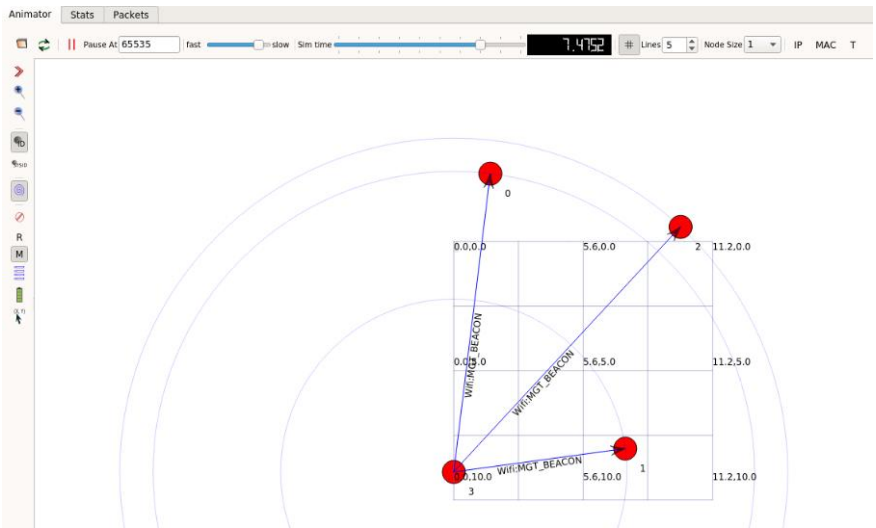


7. Jalankan simulasi tersebut dengan klik icon Play berwarna hijau di pojok kiri atas.



8. Ketika dijalankan, node – node yang ditampilkan seperti yang ditunjukkan pada gambar di bawah ini.

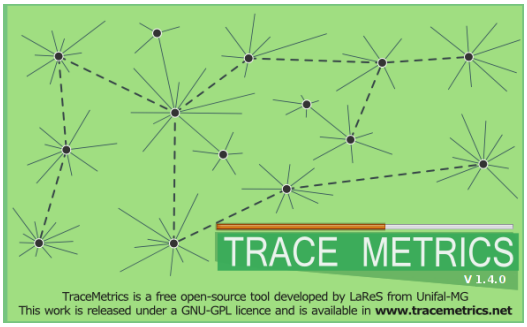




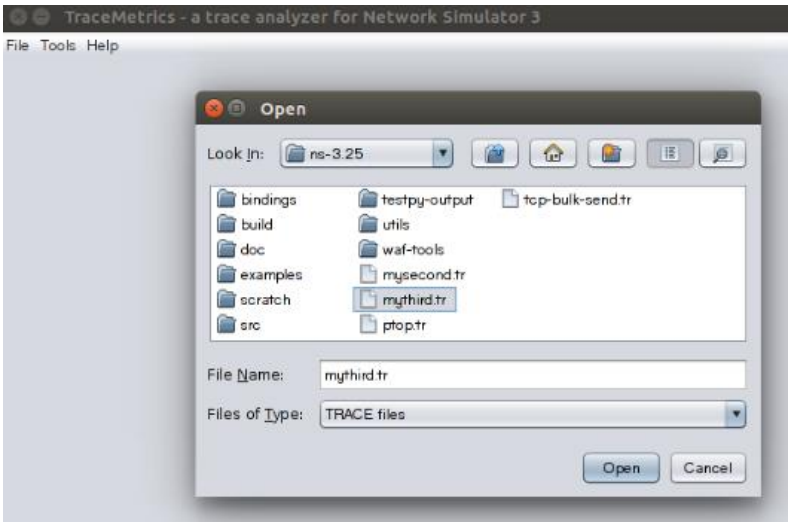
9. Untuk mengetahui trace outputnya, buka direktori *tracemetrics-1.4.0*. Kemudian, buka file *tracemetrics.jar* melalui terminal dengan menuliskan perintah berikut.

```
root@dina-Inspiron-5458:/home/Softwares/tracemetrics-1.4.0# java -jar "tracemetrics.jar"
```

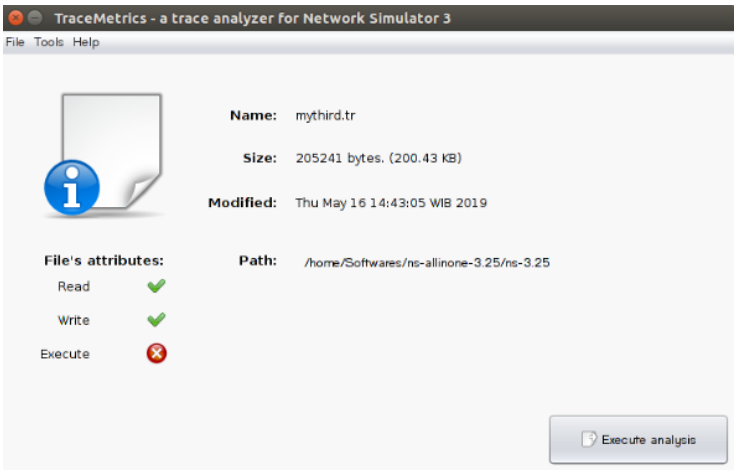
Maka, akan muncul tampilan seperti di bawah ini.



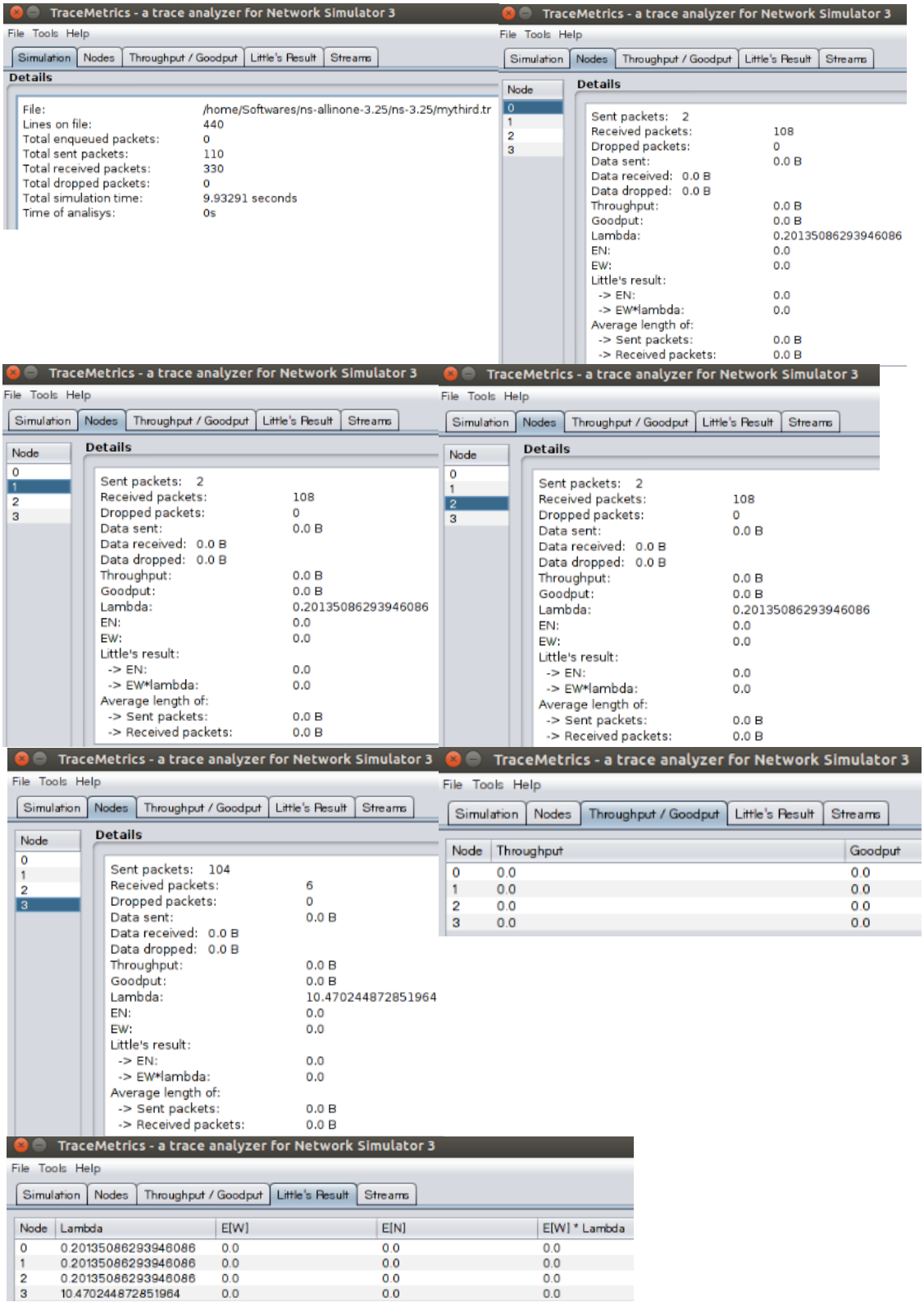
10. Buka file *mythird.tr* dengan cara klik File → Choose File.



11. Setelah itu, akan muncul tampilan seperti di bawah ini. Kemudian, klik Execute analysis.



12. Hasil trace output untuk simulasi Wifi ini adalah seperti berikut.



4.5 Analisa

- Pada percobaan ini, coba ubahlah nilai *Rectangle Value* menjadi (-100, 100, -100, 100).
- Amatilah perubahan yang terjadi.