LAPORAN TUGAS BESAR
MATA KULIAH JARINGAN KOMPUTER

Oleh :

Anyelir Belia Azzahra (1301200048)

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS INFORMATIKA

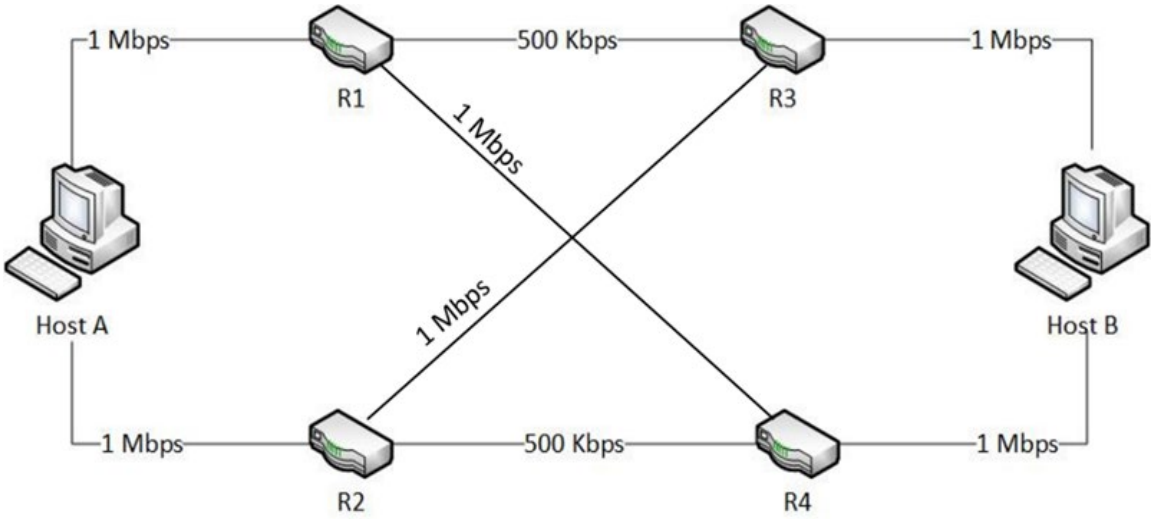
TELKOM UNIVERSITY

2021/2022

BAB 1
METODE TUGAS BESAR

1.1. Spesifikasi Tugas Besar

Perancangan yang akan dibuat adalah topologi dengan spesifikasi sebagai berikut :



1.1.1. CLO 1

Pada CLO ini terdapat spesifikasi pengerjaan dan kriteria penilaian yang akan dilakukan.

- **Goal :**
  - Build topology sesuai dengan soal.
  - Desain subnet masing-masing network.
  - Assign IP sesuai subnet.
  - Uji konektivitas dengan ping antara 2 host yang berada dalam 1 network.

1.1.2. CLO 2

Pada CLO ini terdapat spesifikasi pengerjaan dan kriteria penilaian yang akan dilakukan.

- **Goal :** Mengimplementasikan mekanisme Routing pada topologi yang ada.
  - Uji konektivitas menggunakan ping.

- Membuat table routing di semua host, dibuktikan dengan ping antar host.
- Menganalisis routing yang digunakan menggunakan traceroute

1.1.3. CLO 3

Pada CLO ini terdapat spesifikasi pengerjaan dan kriteria penilaian yang akan dilakukan.

- **Goal :** Membuktikan bahwa TCP telah diimplementasikan dengan benar pada topologi.
  - Generate *traffic* menggunakan iPerf.
  - Capture trafik menggunakan custom script atau Wireshark untuk diinspeksi, dibuktikan dengan trafik di Wireshark/tcpdump.

1.1.4. CLO 4

Pada CLO ini terdapat spesifikasi pengerjaan dan kriteria penilaian yang akan dilakukan.

- **Goal :** Menginspeksi penggunaan queue pada router jaringan.
  - Generate *traffic* menggunakan iPerf.
  - Set ukuran buffer pada router : 20, 40, 60 dan 100.
  - Capture pengaruh ukuran buffer terhadap *delay.*
  - Analisis eksperimen hasil variasi ukuran buffer.
  - Mahasiswa mengerti caranya mengubah buffer dan mengenai pengaruh besar buffer.

BAB 2
PEMBAHASAN

2.1. Tabel Subnetting

Tabel subnetting yang saya buat menggunakan prefix 24 sebagai berikut.

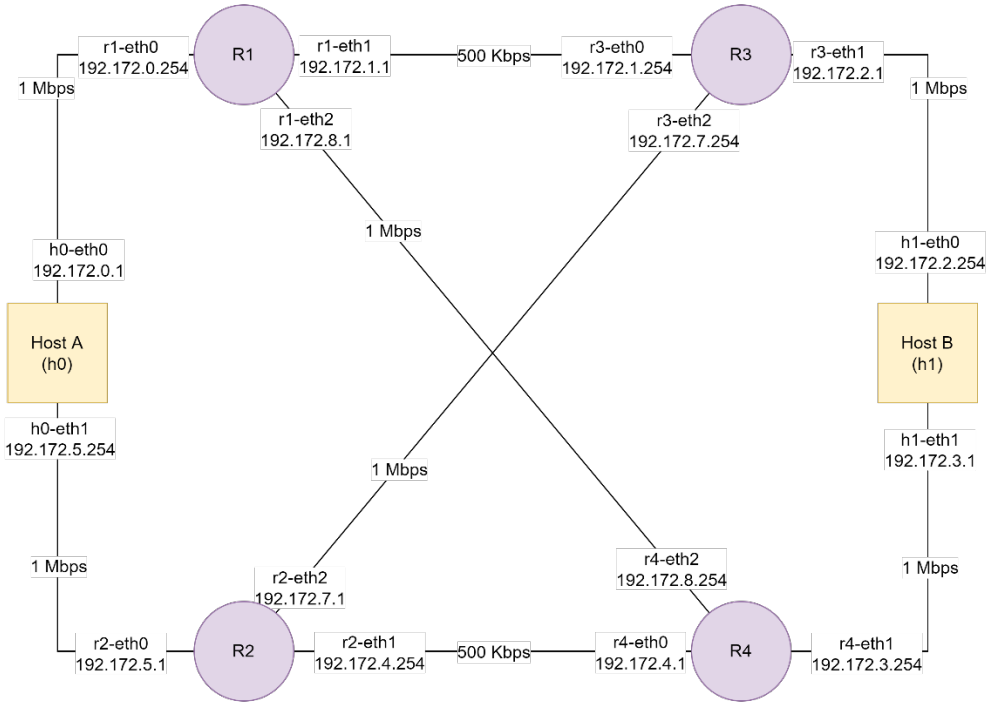
Name	Needs	Alokasi	Network ID	Host Range	Broadcast	Prefix	Subnet Mask
Net 1	2	256	192.172.0.0	192.172.0.1 - 192.172.0.254	192.172.0.255	/24	255.255.255.0
Net 2	2	256	192.172.1.0	192.172.1.1 - 192.172.1.254	192.172.1.255	/24	255.255.255.0
Net 3	2	256	192.172.2.0	192.172.2.1 - 192.172.2.254	192.172.2.255	/24	255.255.255.0
Net 4	2	256	192.172.3.0	192.172.3.1 - 192.172.3.254	192.172.3.255	/24	255.255.255.0
Net 5	2	256	192.172.4.0	192.172.4.1 - 192.172.4.254	192.172.4.255	/24	255.255.255.0
Net 6	2	256	192.172.5.0	192.172.5.1 - 192.172.5.254	192.172.5.255	/24	255.255.255.0
Net 7	2	256	192.172.6.0	192.172.6.1 - 192.172.6.254	192.172.6.255	/24	255.255.255.0
Net 8	2	256	192.172.7.0	192.172.7.1 - 192.172.7.254	192.172.7.255	/24	255.255.255.0

2.2. Implementasi dan Hasil

Tugas besar ini diimplementasikan menggunakan Ubuntu versi 20.04.4, tcpdump, dan wireshark dengan menggunakan Bahasa python.

2.2.1. CLO 1

1) Kesesuaian topologi sesuai dengan soal



2) Implementasi menggunakan Bahasa python

a. Import Library

```
#Anyelir Belia Azzahra  - 1301200048
#!/usr/bin/python
from mininet.net import Mininet
from mininet.topo import Topo
from mininet.node import Node
from mininet.log import setLogLevel
from mininet.cli import CLI
from mininet.link import TCLink
from datetime import datetime
import time
import os
import subprocess
```

b. Prosedur untuk membangun topologi

```
#CLO1
#Membangun Topologi
    # Add Router (Membangun objek untuk R1, R2, R3, R4)
    R1 = net.addHost( 'R1', ip='192.172.0.254/24')
    R2 = net.addHost( 'R2', ip='192.172.5.1/24')
    R3 = net.addHost( 'R3', ip='192.172.2.1/24')
    R4 = net.addHost( 'R4', ip='192.172.3.254/24')

    # Add Host (hostA dan hostB)
    hostA = net.addHost( 'hostA', ip='192.172.0.1/24')
    hostB = net.addHost( 'hostB', ip='192.172.2.254/24')

    # Add Link (Menghubungkan)
    net.addLink(hostA, R1, max_queue_size=100, intfName1='hostA-eth0',intfName2='R1-eth0', cls=>
    net.addLink(hostA, R2, max_queue_size=100, intfName1='hostA-eth1',intfName2='R2-eth0', cls=>
    net.addLink(hostB, R3, max_queue_size=100, intfName1='hostB-eth0',intfName2='R3-eth1', cls=>
    net.addLink(hostB, R4, max_queue_size=100, intfName1='hostB-eth1',intfName2='R4-eth1', cls=>
    net.addLink(R1, R3, max_queue_size=100, intfName1='R1-eth1',intfName2='R3-eth0', cls=TCLink>
    net.addLink(R1, R4, max_queue_size=100, intfName1='R1-eth2',intfName2='R4-eth2', cls=TCLink>
    net.addLink(R2, R4, max_queue_size=100, intfName1='R2-eth1',intfName2='R4-eth0', cls=TCLink>
    net.addLink(R2, R3, max_queue_size=100, intfName1='R2-eth2',intfName2='R3-eth2', cls=TCLink>
```

c. Configurasi IP

```
# Config IP

#Memasukkan IP pada hostA
hostA.cmd("ifconfig hostA-eth0 0")
hostA.cmd("ifconfig hostA-eth1 0")
hostA.cmd("ifconfig hostA-eth0 192.172.0.1 netmask 255.255.255.0")
hostA.cmd("ifconfig hostA-eth1 192.172.5.254 netmask 255.255.255.0")

#Memasukkan IP pada hostB
hostB.cmd("ifconfig hostB-eth0 0")
hostB.cmd("ifconfig hostB-eth1 0")
hostB.cmd("ifconfig hostB-eth0 192.172.2.254 netmask 255.255.255.0")
hostB.cmd("ifconfig hostB-eth1 192.172.3.1 netmask 255.255.255.0")

# Config router

R1.cmd( 'sysctl net.ipv4.ip_forward=1' )
R2.cmd( 'sysctl net.ipv4.ip_forward=1' )
R3.cmd( 'sysctl net.ipv4.ip_forward=1' )
R4.cmd( 'sysctl net.ipv4.ip_forward=1' )

# Add IP Address for Router

#Memasukkan IP router pada R1
R1.cmd( 'ip addr add 192.172.0.254/24 brd + dev R1-eth0' )
R1.cmd( 'ip addr add 192.172.1.1/24 brd + dev R1-eth1' )
R1.cmd( 'ip addr add 192.172.6.1/24 brd + dev R1-eth2' )

#Memasukkan IP router pada R2
R2.cmd( 'ip addr add 192.172.5.1/24 brd + dev R2-eth0' )
R2.cmd( 'ip addr add 192.172.4.254/24 brd + dev R2-eth1' )
R2.cmd( 'ip addr add 192.172.7.1/24 brd + dev R2-eth2' )

#Memasukkan IP router pada R3
R3.cmd( 'ip addr add 192.172.1.254/24 brd + dev R3-eth0' )
R3.cmd( 'ip addr add 192.172.2.1/24 brd + dev R3-eth1' )
R3.cmd( 'ip addr add 192.172.7.254/24 brd + dev R3-eth2' )

#Memasukkan IP router pada R4
R4.cmd( 'ip addr add 192.172.4.1/24 brd + dev R4-eth0' )
R4.cmd( 'ip addr add 192.172.3.254/24 brd + dev R4-eth1' )
R4.cmd( 'ip addr add 192.172.6.254/24 brd + dev R4-eth2' )
```

3) Hasil Uji Konektivitas

Hasil Uji konektivitas dengan ping antara 2 host yang berada dalam 1 network.

a. Menjalankan file.py

```
(1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (0.50Mbi
t) (0.50Mbit) (1.00Mbit) (1.00Mbit) (0.50Mbit) (0.50Mbit) (1.00Mbit) (1.00Mbit) *** Configuring
hosts
R1 R2 R3 R4 hostA hostB
*** Starting controller

*** Starting 0 switches

*** Configuring hosts
R1 R2 R3 R4 hostA hostB
*** Ping: testing ping reachability
R1 -> R2 R3 R4 hostA hostB
R2 -> R1 R3 R4 hostA hostB
R3 -> R1 R2 R4 hostA hostB
R4 -> R1 R2 R3 hostA hostB
hostA -> R1 R2 R3 R4 hostB
hostB -> R1 R2 R3 R4 hostA
*** Results: 0% dropped (30/30 received)

 0.0
*** Starting CLI:
```

b. Uji konektivitas Host A → Host B

```
mininet> hostA ping hostB
PING 192.172.2.254 (192.172.2.254) 56(84) bytes of data.
64 bytes from 192.172.2.254: icmp_seq=1 ttl=62 time=0.075 ms
64 bytes from 192.172.2.254: icmp_seq=2 ttl=62 time=0.087 ms
64 bytes from 192.172.2.254: icmp_seq=3 ttl=62 time=0.067 ms
64 bytes from 192.172.2.254: icmp_seq=4 ttl=62 time=0.071 ms
64 bytes from 192.172.2.254: icmp_seq=5 ttl=62 time=0.094 ms
64 bytes from 192.172.2.254: icmp_seq=6 ttl=62 time=0.070 ms
64 bytes from 192.172.2.254: icmp_seq=7 ttl=62 time=0.088 ms
64 bytes from 192.172.2.254: icmp_seq=8 ttl=62 time=0.066 ms
64 bytes from 192.172.2.254: icmp_seq=9 ttl=62 time=0.068 ms
64 bytes from 192.172.2.254: icmp_seq=10 ttl=62 time=0.101 ms
64 bytes from 192.172.2.254: icmp_seq=11 ttl=62 time=0.091 ms
64 bytes from 192.172.2.254: icmp_seq=12 ttl=62 time=0.066 ms
64 bytes from 192.172.2.254: icmp_seq=13 ttl=62 time=0.090 ms
64 bytes from 192.172.2.254: icmp_seq=14 ttl=62 time=0.068 ms
64 bytes from 192.172.2.254: icmp_seq=15 ttl=62 time=0.066 ms
64 bytes from 192.172.2.254: icmp_seq=16 ttl=62 time=0.071 ms
64 bytes from 192.172.2.254: icmp_seq=17 ttl=62 time=0.066 ms
64 bytes from 192.172.2.254: icmp_seq=18 ttl=62 time=0.067 ms
64 bytes from 192.172.2.254: icmp_seq=19 ttl=62 time=0.099 ms
64 bytes from 192.172.2.254: icmp_seq=20 ttl=62 time=0.147 ms
64 bytes from 192.172.2.254: icmp_seq=21 ttl=62 time=0.076 ms
64 bytes from 192.172.2.254: icmp_seq=22 ttl=62 time=0.190 ms
64 bytes from 192.172.2.254: icmp_seq=23 ttl=62 time=0.102 ms
64 bytes from 192.172.2.254: icmp_seq=24 ttl=62 time=0.069 ms
64 bytes from 192.172.2.254: icmp_seq=25 ttl=62 time=0.076 ms
64 bytes from 192.172.2.254: icmp_seq=26 ttl=62 time=0.096 ms
64 bytes from 192.172.2.254: icmp_seq=27 ttl=62 time=0.075 ms
64 bytes from 192.172.2.254: icmp_seq=28 ttl=62 time=0.078 ms
^C
--- 192.172.2.254 ping statistics ---
28 packets transmitted, 28 received, 0% packet loss, time 27631ms
rtt min/avg/max/mdev = 0.066/0.085/0.190/0.026 ms
```

c. Uji Konektivitas Host B → Host A

```
mininet> hostB ping hostA
PING 192.172.0.1 (192.172.0.1) 56(84) bytes of data.
64 bytes from 192.172.0.1: icmp_seq=1 ttl=62 time=5.90 ms
64 bytes from 192.172.0.1: icmp_seq=2 ttl=62 time=0.195 ms
64 bytes from 192.172.0.1: icmp_seq=3 ttl=62 time=0.090 ms
64 bytes from 192.172.0.1: icmp_seq=4 ttl=62 time=0.089 ms
64 bytes from 192.172.0.1: icmp_seq=5 ttl=62 time=0.067 ms
64 bytes from 192.172.0.1: icmp_seq=6 ttl=62 time=0.067 ms
64 bytes from 192.172.0.1: icmp_seq=7 ttl=62 time=0.074 ms
64 bytes from 192.172.0.1: icmp_seq=8 ttl=62 time=0.067 ms
64 bytes from 192.172.0.1: icmp_seq=9 ttl=62 time=0.085 ms
64 bytes from 192.172.0.1: icmp_seq=10 ttl=62 time=0.072 ms
64 bytes from 192.172.0.1: icmp_seq=11 ttl=62 time=0.086 ms
64 bytes from 192.172.0.1: icmp_seq=12 ttl=62 time=0.088 ms
^C
--- 192.172.0.1 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11244ms
rtt min/avg/max/mdev = 0.067/0.573/5.898/1.605 ms
```

2.2.2. CLO 2

1) Implementasi mekanisme routing

a. Routing Host

```
#CLO2

#Routing Host
hostA.cmd('ip rule add from 192.172.0.1 table 1')
hostA.cmd('ip rule add from 192.172.5.254 table 2')
hostA.cmd('ip route add 192.172.0.0/24 dev hostA-eth0 scope link table 1')
hostA.cmd('ip route add default via 192.172.0.254 dev hostA-eth0 table 1')
hostA.cmd('ip route add 192.172.5.0/24 dev hostA-eth1 scope link table 2')
hostA.cmd('ip route add default via 192.172.5.1 dev hostA-eth1 table 2')
hostA.cmd('ip route add default scope global nexthop via 192.172.0.254 dev hostA-eth0')
hostA.cmd('ip route add default scope global nexthop via 192.172.5.1 dev hostA-eth1')

hostB.cmd('ip rule add from 192.172.2.254 table 3')
hostB.cmd('ip rule add from 192.172.3.1 table 4')
hostB.cmd('ip route add 192.172.2.0/24 dev hostB-eth0 scope link table 3')
hostB.cmd('ip route add default via 192.172.2.1 dev hostB-eth0 table 3')
hostB.cmd('ip route add 192.172.3.0/24 dev hostB-eth1 scope link table 4')
hostB.cmd('ip route add default via 192.172.2.254 dev hostB-eth1 table 4')
hostB.cmd('ip route add default scope global nexthop via 192.172.2.1 dev hostB-eth0')
hostB.cmd('ip route add default scope global nexthop via 192.172.3.254 dev hostB-eth1')
```

b. Routing Router

```
# Static Routing (router)
R1.cmd('route add -net 192.172.2.0/24 gw 192.172.1.254')
R1.cmd('route add -net 192.172.3.0/24 gw 192.172.6.254')
R1.cmd('route add -net 192.172.4.0/24 gw 192.172.6.2')
R1.cmd('route add -net 192.172.5.0/24 gw 192.172.6.254')
R1.cmd('route add -net 192.172.7.0/24 gw 192.172.1.254')

R2.cmd('route add -net 192.172.0.0/24 gw 192.172.7.254')
R2.cmd('route add -net 192.172.1.0/24 gw 192.172.7.254')
R2.cmd('route add -net 192.172.2.0/24 gw 192.172.7.254')
R2.cmd('route add -net 192.172.3.0/24 gw 192.172.4.1')
R2.cmd('route add -net 192.172.6.0/24 gw 192.172.4.1')

R3.cmd('route add -net 192.172.0.0/24 gw 192.172.1.1')
R3.cmd('route add -net 192.172.3.0/24 gw 192.172.7.1')
R3.cmd('route add -net 192.172.4.0/24 gw 192.172.7.1')
R3.cmd('route add -net 192.172.5.0/24 gw 192.172.1.1')
R3.cmd('route add -net 192.172.6.0/24 gw 192.172.1.1')

R4.cmd('route add -net 192.172.0.0/24 gw 192.172.6.1')
R4.cmd('route add -net 192.172.1.0/24 gw 192.172.6.1')
R4.cmd('route add -net 192.172.2.0/24 gw 192.172.6.1')
R4.cmd('route add -net 192.172.5.0/24 gw 192.172.4.254')
R4.cmd('route add -net 192.172.7.0/24 gw 192.172.4.254')
```

Routing dilakukan dengan tujuan agar router dapat mengetahui dan dapat melalui jalur yang terhubung ataupun tidak terhubung langsung dengan router menggunakan perantara router lain.

2) Uji Konektivitas

a. Menguji Konektivitas (pingall)

```
mininet> pingall
*** Ping: testing ping reachability
R1 -> R2 R3 R4 hostA hostB
R2 -> R1 R3 R4 hostA hostB
R3 -> R1 R2 R4 hostA hostB
R4 -> R1 R2 R3 hostA hostB
hostA -> R1 R2 R3 R4 hostB
hostB -> R1 R2 R3 R4 hostA
*** Results: 0% dropped (30/30 received)
mininet>
```

b. HostA → R1

```
mininet> hostA ping R1
PING 192.172.0.254 (192.172.0.254) 56(84) bytes of data.
64 bytes from 192.172.0.254: icmp_seq=1 ttl=64 time=0.701 ms
64 bytes from 192.172.0.254: icmp_seq=2 ttl=64 time=0.062 ms
64 bytes from 192.172.0.254: icmp_seq=3 ttl=64 time=0.052 ms
64 bytes from 192.172.0.254: icmp_seq=4 ttl=64 time=0.050 ms
64 bytes from 192.172.0.254: icmp_seq=5 ttl=64 time=0.049 ms
64 bytes from 192.172.0.254: icmp_seq=6 ttl=64 time=0.075 ms
64 bytes from 192.172.0.254: icmp_seq=7 ttl=64 time=0.062 ms
64 bytes from 192.172.0.254: icmp_seq=8 ttl=64 time=0.060 ms
^C
--- 192.172.0.254 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7151ms
rtt min/avg/max/mdev = 0.049/0.138/0.701/0.212 ms
```

c. HostA → R2

```
mininet> hostA ping R2
PING 192.172.5.1 (192.172.5.1) 56(84) bytes of data.
64 bytes from 192.172.5.1: icmp_seq=1 ttl=64 time=0.123 ms
64 bytes from 192.172.5.1: icmp_seq=2 ttl=64 time=0.064 ms
64 bytes from 192.172.5.1: icmp_seq=3 ttl=64 time=0.060 ms
64 bytes from 192.172.5.1: icmp_seq=4 ttl=64 time=0.053 ms
64 bytes from 192.172.5.1: icmp_seq=5 ttl=64 time=0.077 ms
64 bytes from 192.172.5.1: icmp_seq=6 ttl=64 time=0.070 ms
64 bytes from 192.172.5.1: icmp_seq=7 ttl=64 time=0.074 ms
64 bytes from 192.172.5.1: icmp_seq=8 ttl=64 time=0.049 ms
^C
--- 192.172.5.1 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7148ms
rtt min/avg/max/mdev = 0.049/0.071/0.123/0.021 ms
mininet>
```

d. HostA → R3

```
mininet> hostA ping R3
PING 192.172.2.1 (192.172.2.1) 56(84) bytes of data.
64 bytes from 192.172.2.1: icmp_seq=1 ttl=63 time=1.16 ms
64 bytes from 192.172.2.1: icmp_seq=2 ttl=63 time=0.082 ms
64 bytes from 192.172.2.1: icmp_seq=3 ttl=63 time=0.115 ms
64 bytes from 192.172.2.1: icmp_seq=4 ttl=63 time=0.065 ms
64 bytes from 192.172.2.1: icmp_seq=5 ttl=63 time=0.063 ms
64 bytes from 192.172.2.1: icmp_seq=6 ttl=63 time=0.063 ms
64 bytes from 192.172.2.1: icmp_seq=7 ttl=63 time=0.064 ms
64 bytes from 192.172.2.1: icmp_seq=8 ttl=63 time=0.080 ms
^C
--- 192.172.2.1 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7140ms
rtt min/avg/max/mdev = 0.063/0.211/1.162/0.359 ms
mininet>
```

e. HostA → R4

```
mininet> hostA ping R4
PING 192.172.3.254 (192.172.3.254) 56(84) bytes of data.
64 bytes from 192.172.3.254: icmp_seq=1 ttl=63 time=0.688 ms
64 bytes from 192.172.3.254: icmp_seq=2 ttl=63 time=0.075 ms
64 bytes from 192.172.3.254: icmp_seq=3 ttl=63 time=0.135 ms
64 bytes from 192.172.3.254: icmp_seq=4 ttl=63 time=0.066 ms
64 bytes from 192.172.3.254: icmp_seq=5 ttl=63 time=0.089 ms
64 bytes from 192.172.3.254: icmp_seq=6 ttl=63 time=0.089 ms
64 bytes from 192.172.3.254: icmp_seq=7 ttl=63 time=0.064 ms
64 bytes from 192.172.3.254: icmp_seq=8 ttl=63 time=0.076 ms
^C
--- 192.172.3.254 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7171ms
rtt min/avg/max/mdev = 0.064/0.160/0.688/0.200 ms
mininet>
```

f. HostB → R1

```
mininet> hostB ping R1
PING 192.172.0.254 (192.172.0.254) 56(84) bytes of data.
64 bytes from 192.172.0.254: icmp_seq=1 ttl=63 time=0.102 ms
64 bytes from 192.172.0.254: icmp_seq=2 ttl=63 time=0.061 ms
64 bytes from 192.172.0.254: icmp_seq=3 ttl=63 time=0.063 ms
64 bytes from 192.172.0.254: icmp_seq=4 ttl=63 time=0.081 ms
64 bytes from 192.172.0.254: icmp_seq=5 ttl=63 time=0.084 ms
^C
--- 192.172.0.254 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4094ms
rtt min/avg/max/mdev = 0.061/0.078/0.102/0.015 ms
mininet>
```

g. HostB → R2

```
mininet> hostB ping R2
PING 192.172.5.1 (192.172.5.1) 56(84) bytes of data.
64 bytes from 192.172.5.1: icmp_seq=1 ttl=63 time=0.118 ms
64 bytes from 192.172.5.1: icmp_seq=2 ttl=63 time=0.072 ms
64 bytes from 192.172.5.1: icmp_seq=3 ttl=63 time=0.097 ms
64 bytes from 192.172.5.1: icmp_seq=4 ttl=63 time=0.077 ms
^C
--- 192.172.5.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3055ms
rtt min/avg/max/mdev = 0.072/0.091/0.118/0.018 ms
mininet>
```

h. HostB → R3

```
mininet> hostB ping R3
PING 192.172.2.1 (192.172.2.1) 56(84) bytes of data.
64 bytes from 192.172.2.1: icmp_seq=1 ttl=64 time=0.135 ms
64 bytes from 192.172.2.1: icmp_seq=2 ttl=64 time=0.053 ms
64 bytes from 192.172.2.1: icmp_seq=3 ttl=64 time=0.066 ms
^C
--- 192.172.2.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2043ms
rtt min/avg/max/mdev = 0.053/0.084/0.135/0.035 ms
mininet>
```

i. HostB → R4

```
mininet> hostB ping R4
PING 192.172.3.254 (192.172.3.254) 56(84) bytes of data.
64 bytes from 192.172.3.254: icmp_seq=1 ttl=64 time=0.055 ms
64 bytes from 192.172.3.254: icmp_seq=2 ttl=64 time=0.059 ms
64 bytes from 192.172.3.254: icmp_seq=3 ttl=64 time=0.049 ms
64 bytes from 192.172.3.254: icmp_seq=4 ttl=64 time=0.064 ms
^C
--- 192.172.3.254 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3055ms
rtt min/avg/max/mdev = 0.049/0.056/0.064/0.005 ms
mininet>
```

3) Table routing di semua host

a. Routing table hostA dan hostB

```
mininet> hostA route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.172.0.254   0.0.0.0         UG    0      0        0 hostA-eth0
192.172.0.0     0.0.0.0         255.255.255.0   U     0      0        0 hostA-eth0
192.172.5.0     0.0.0.0         255.255.255.0   U     0      0        0 hostA-eth1
mininet> hostB route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.172.2.1     0.0.0.0         UG    0      0        0 hostB-eth0
192.172.2.0     0.0.0.0         255.255.255.0   U     0      0        0 hostB-eth0
192.172.3.0     0.0.0.0         255.255.255.0   U     0      0        0 hostB-eth1
mininet>
```

b. Routing Table router

```
mininet> R1 route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.172.0.0     0.0.0.0         255.255.255.0   U     0      0        0 R1-eth0
192.172.1.0     0.0.0.0         255.255.255.0   U     0      0        0 R1-eth1
192.172.2.0     192.172.1.254   255.255.255.0   UG    0      0        0 R1-eth1
192.172.3.0     192.172.6.254   255.255.255.0   UG    0      0        0 R1-eth2
192.172.4.0     192.172.6.2     255.255.255.0   UG    0      0        0 R1-eth2
192.172.5.0     192.172.6.254   255.255.255.0   UG    0      0        0 R1-eth2
192.172.6.0     0.0.0.0         255.255.255.0   U     0      0        0 R1-eth2
192.172.7.0     192.172.1.254   255.255.255.0   UG    0      0        0 R1-eth1
mininet> R2 route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.172.0.0     192.172.7.254   255.255.255.0   UG    0      0        0 R2-eth2
192.172.1.0     192.172.7.254   255.255.255.0   UG    0      0        0 R2-eth2
192.172.2.0     192.172.7.254   255.255.255.0   UG    0      0        0 R2-eth2
192.172.3.0     192.172.4.1     255.255.255.0   UG    0      0        0 R2-eth1
192.172.4.0     0.0.0.0         255.255.255.0   U     0      0        0 R2-eth1
192.172.5.0     0.0.0.0         255.255.255.0   U     0      0        0 R2-eth0
192.172.6.0     192.172.4.1     255.255.255.0   UG    0      0        0 R2-eth1
192.172.7.0     0.0.0.0         255.255.255.0   U     0      0        0 R2-eth2
mininet> R3 route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.172.0.0     192.172.1.1     255.255.255.0   UG    0      0        0 R3-eth0
192.172.1.0     0.0.0.0         255.255.255.0   U     0      0        0 R3-eth0
192.172.2.0     0.0.0.0         255.255.255.0   U     0      0        0 R3-eth1
192.172.3.0     192.172.7.1     255.255.255.0   UG    0      0        0 R3-eth2
192.172.4.0     192.172.7.1     255.255.255.0   UG    0      0        0 R3-eth2
192.172.5.0     192.172.1.1     255.255.255.0   UG    0      0        0 R3-eth0
192.172.6.0     192.172.1.1     255.255.255.0   UG    0      0        0 R3-eth0
192.172.7.0     0.0.0.0         255.255.255.0   U     0      0        0 R3-eth2
mininet> R4 route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.172.0.0     192.172.6.1     255.255.255.0   UG    0      0        0 R4-eth2
192.172.1.0     192.172.6.1     255.255.255.0   UG    0      0        0 R4-eth2
192.172.2.0     192.172.6.1     255.255.255.0   UG    0      0        0 R4-eth2
192.172.3.0     0.0.0.0         255.255.255.0   U     0      0        0 R4-eth1
192.172.4.0     0.0.0.0         255.255.255.0   U     0      0        0 R4-eth0
192.172.5.0     192.172.4.254   255.255.255.0   UG    0      0        0 R4-eth0
192.172.6.0     0.0.0.0         255.255.255.0   U     0      0        0 R4-eth2
192.172.7.0     192.172.4.254   255.255.255.0   UG    0      0        0 R4-eth0
mininet>
```

4) Menganalisis routing yang digunakan menggunakan traceroute

a. Traceroute HostA → HostB

```
mininet> hostA traceroute hostB
traceroute to 192.172.2.254 (192.172.2.254), 30 hops max, 60 byte packets
 1  192.172.0.254 (192.172.0.254)  0.390 ms  0.017 ms  0.010 ms
 2  192.172.1.254 (192.172.1.254)  0.382 ms  0.050 ms  0.019 ms
 3  192.172.2.254 (192.172.2.254)  0.114 ms  0.026 ms  0.022 ms
mininet>
```

b. Traceroute HostA ke semua router

```
mininet> hostA traceroute R1
traceroute to 192.172.0.254 (192.172.0.254), 30 hops max, 60 byte packets
 1  192.172.0.254 (192.172.0.254)  0.460 ms  0.393 ms  0.368 ms
mininet> hostA traceroute R2
traceroute to 192.172.5.1 (192.172.5.1), 30 hops max, 60 byte packets
 1  192.172.5.1 (192.172.5.1)  1.083 ms  0.299 ms  0.219 ms
mininet> hostA traceroute R3
traceroute to 192.172.2.1 (192.172.2.1), 30 hops max, 60 byte packets
 1  192.172.0.254 (192.172.0.254)  0.055 ms  0.012 ms  0.009 ms
 2  192.172.2.1 (192.172.2.1)  0.028 ms  0.024 ms  0.016 ms
mininet> hostA traceroute R4
traceroute to 192.172.3.254 (192.172.3.254), 30 hops max, 60 byte packets
 1  192.172.0.254 (192.172.0.254)  3.487 ms  3.438 ms  3.380 ms
 2  192.172.3.254 (192.172.3.254)  3.359 ms  1.137 ms  1.104 ms
mininet>
```

c. Traceroute HostB → HostA

```
mininet> hostB traceroute hostA
traceroute to 192.172.0.1 (192.172.0.1), 30 hops max, 60 byte packets
 1  192.172.2.1 (192.172.2.1)  0.138 ms  0.018 ms  0.009 ms
 2  192.172.1.1 (192.172.1.1)  0.093 ms  0.022 ms  0.017 ms
 3  192.172.0.1 (192.172.0.1)  0.091 ms  0.019 ms  0.015 ms
mininet>
```

d. Traceroute HostB ke semua router

```
mininet> hostB traceroute R1
traceroute to 192.172.0.254 (192.172.0.254), 30 hops max, 60 byte packets
 1  192.172.2.1 (192.172.2.1)  0.344 ms  0.262 ms  0.244 ms
 2  192.172.0.254 (192.172.0.254)  0.231 ms  0.204 ms  0.188 ms
mininet> hostB traceroute R2
traceroute to 192.172.5.1 (192.172.5.1), 30 hops max, 60 byte packets
 1  192.172.2.1 (192.172.2.1)  0.447 ms  0.399 ms  0.382 ms
 2  192.172.1.1 (192.172.1.1)  0.364 ms * *
 3  192.172.6.254 (192.172.6.254)  0.311 ms  0.274 ms  0.256 ms
 4  192.172.5.1 (192.172.5.1)  0.237 ms  0.159 ms  0.136 ms
mininet> hostB traceroute R3
traceroute to 192.172.2.1 (192.172.2.1), 30 hops max, 60 byte packets
 1  192.172.2.1 (192.172.2.1)  0.385 ms  0.341 ms  0.321 ms
mininet> hostB traceroute R4
traceroute to 192.172.3.254 (192.172.3.254), 30 hops max, 60 byte packets
 1  192.172.3.254 (192.172.3.254)  0.290 ms  0.195 ms  0.178 ms
mininet>
```

e. Traceroute R1 ke hostA,hostB,R2,R3,R4

```
mininet> R1 traceroute hostA
traceroute to 192.172.0.1 (192.172.0.1), 30 hops max, 60 byte packets
 1  192.172.0.1 (192.172.0.1)  0.062 ms  0.016 ms  0.015 ms
mininet> R1 traceroute hostB
traceroute to 192.172.2.254 (192.172.2.254), 30 hops max, 60 byte packets
 1  192.172.1.254 (192.172.1.254)  0.505 ms  0.433 ms  0.378 ms
 2  192.172.2.254 (192.172.2.254)  0.357 ms  0.314 ms  0.284 ms
mininet> R1 traceroute R2
traceroute to 192.172.5.1 (192.172.5.1), 30 hops max, 60 byte packets
 1  192.172.6.254 (192.172.6.254)  0.296 ms  0.246 ms  0.234 ms
 2  192.172.5.1 (192.172.5.1)  0.217 ms  0.191 ms  0.175 ms
mininet> R1 traceroute R3
traceroute to 192.172.2.1 (192.172.2.1), 30 hops max, 60 byte packets
 1  192.172.2.1 (192.172.2.1)  0.248 ms  0.203 ms  0.189 ms
mininet> R1 traceroute R4
traceroute to 192.172.3.254 (192.172.3.254), 30 hops max, 60 byte packets
 1  192.172.3.254 (192.172.3.254)  0.240 ms  0.191 ms  0.178 ms
mininet>
```

f. Traceroute R2 ke hostA,hostB,R1,R3,R4

```
mininet> R2 traceroute hostA
traceroute to 192.172.0.1 (192.172.0.1), 30 hops max, 60 byte packets
 1  192.172.7.254 (192.172.7.254)  0.042 ms  0.008 ms  0.006 ms
 2  192.172.1.1 (192.172.1.1)  0.059 ms  0.020 ms  0.013 ms
 3  192.172.0.1 (192.172.0.1)  0.034 ms  0.015 ms  0.013 ms
mininet> R2 traceroute hostB
traceroute to 192.172.2.254 (192.172.2.254), 30 hops max, 60 byte packets
 1  192.172.7.254 (192.172.7.254)  0.491 ms  0.419 ms  0.396 ms
 2  192.172.2.254 (192.172.2.254)  0.379 ms  0.339 ms  0.308 ms
mininet> R2 traceroute R1
traceroute to 192.172.0.254 (192.172.0.254), 30 hops max, 60 byte packets
 1  192.172.7.254 (192.172.7.254)  0.421 ms  0.359 ms  0.338 ms
 2  192.172.0.254 (192.172.0.254)  0.322 ms  0.289 ms  0.267 ms
mininet> R2 traceroute R3
traceroute to 192.172.2.1 (192.172.2.1), 30 hops max, 60 byte packets
 1  192.172.2.1 (192.172.2.1)  0.235 ms  0.187 ms  0.174 ms
mininet> R2 traceroute R4
traceroute to 192.172.3.254 (192.172.3.254), 30 hops max, 60 byte packets
 1  192.172.3.254 (192.172.3.254)  0.418 ms  0.341 ms  0.314 ms
mininet>
```

g. Traceroute R3 ke hostA,hostB,R1,R2,R4

```
mininet> R3 traceroute hostA
traceroute to 192.172.0.1 (192.172.0.1), 30 hops max, 60 byte packets
 1  192.172.1.1 (192.172.1.1)  0.082 ms  0.010 ms  0.006 ms
 2  192.172.0.1 (192.172.0.1)  0.068 ms  0.015 ms  0.013 ms
mininet> R3 traceroute hostB
traceroute to 192.172.2.254 (192.172.2.254), 30 hops max, 60 byte packets
 1  192.172.2.254 (192.172.2.254)  0.390 ms  0.321 ms  0.295 ms
mininet> R3 traceroute R1
traceroute to 192.172.0.254 (192.172.0.254), 30 hops max, 60 byte packets
 1  192.172.0.254 (192.172.0.254)  0.044 ms  0.008 ms  0.006 ms
mininet> R3 traceroute R2
traceroute to 192.172.5.1 (192.172.5.1), 30 hops max, 60 byte packets
 1  192.172.1.1 (192.172.1.1)  0.596 ms  0.478 ms  0.454 ms
 2  192.172.6.254 (192.172.6.254)  0.438 ms  0.401 ms  0.377 ms
 3  192.172.5.1 (192.172.5.1)  0.354 ms  0.307 ms  0.281 ms
mininet> R3 traceroute R4
traceroute to 192.172.3.254 (192.172.3.254), 30 hops max, 60 byte packets
 1  192.172.7.1 (192.172.7.1)  1.397 ms  0.467 ms  0.393 ms
 2  192.172.3.254 (192.172.3.254)  0.370 ms  0.310 ms  0.277 ms
mininet>
```

h. Traceroute R4 ke hostA,hostB,R1,R2,R3

```
mininet> R4 traceroute hostA
traceroute to 192.172.0.1 (192.172.0.1), 30 hops max, 60 byte packets
 1  192.172.6.1 (192.172.6.1)  0.479 ms  0.018 ms  0.009 ms
 2  192.172.0.1 (192.172.0.1)  0.031 ms  0.018 ms  0.016 ms
mininet> R4 traceroute hostB
traceroute to 192.172.2.254 (192.172.2.254), 30 hops max, 60 byte packets
 1  192.172.6.1 (192.172.6.1)  1.103 ms  1.055 ms  1.043 ms
 2  192.172.1.254 (192.172.1.254)  1.031 ms  1.007 ms  0.993 ms
 3  192.172.2.254 (192.172.2.254)  0.976 ms  0.207 ms  0.171 ms
mininet> R4 traceroute R1
traceroute to 192.172.0.254 (192.172.0.254), 30 hops max, 60 byte packets
 1  192.172.0.254 (192.172.0.254)  0.251 ms  0.219 ms  0.206 ms
mininet> R4 traceroute R2
traceroute to 192.172.5.1 (192.172.5.1), 30 hops max, 60 byte packets
 1  192.172.5.1 (192.172.5.1)  0.379 ms  0.316 ms  0.294 ms
mininet> R4 traceroute R3
traceroute to 192.172.2.1 (192.172.2.1), 30 hops max, 60 byte packets
 1  192.172.6.1 (192.172.6.1)  0.289 ms  0.240 ms  0.225 ms
 2  192.172.2.1 (192.172.2.1)  0.215 ms  0.190 ms  0.173 ms
mininet>
```

Penggunaan Traceroute akan menunjukkan jalur router yang dilewati oleh paket yang kita kirimkan ke host tertentu. Contohnya hasil traceroute antara hostA dan hostB.

```
mininet> hostA traceroute hostB
traceroute to 192.172.2.254 (192.172.2.254), 30 hops max, 60 byte packets
 1  192.172.0.254 (192.172.0.254)  0.390 ms  0.017 ms  0.010 ms
 2  192.172.1.254 (192.172.1.254)  0.382 ms  0.050 ms  0.019 ms
 3  192.172.2.254 (192.172.2.254)  0.114 ms  0.026 ms  0.022 ms
mininet>
```

Pada traceroute tersebut terlihat paket dengan IP 192.172.2.254 berjalan melewati IP 192.172.0.254 dimana merupakan **R1**, kemudian melewati IP 192.172.1.254 dimana merupakan **R3**, dan berakhir diterima oleh host tujuan dengan IP 192.172.2.254 dimana merupakan **hostB**. Proses tersebut menandakan bahwa pengiriman paket ini sudah sesuai dengan topologi yang diberikan.

2.2.3. CLO 3

1) Generate *traffic* menggunakan iPerf

   a. Iperf hostB dengan hostA

```
mininet> xterm hostA
mininet> iperf hostB hostA
*** Iperf: testing TCP bandwidth between hostB and hostA
.*** Results: ['478 Kbits/sec', '1.06 Mbits/sec']
mininet> iperf hostB hostA
*** Iperf: testing TCP bandwidth between hostB and hostA
.*** Results: ['478 Kbits/sec', '1.08 Mbits/sec']
mininet>
```

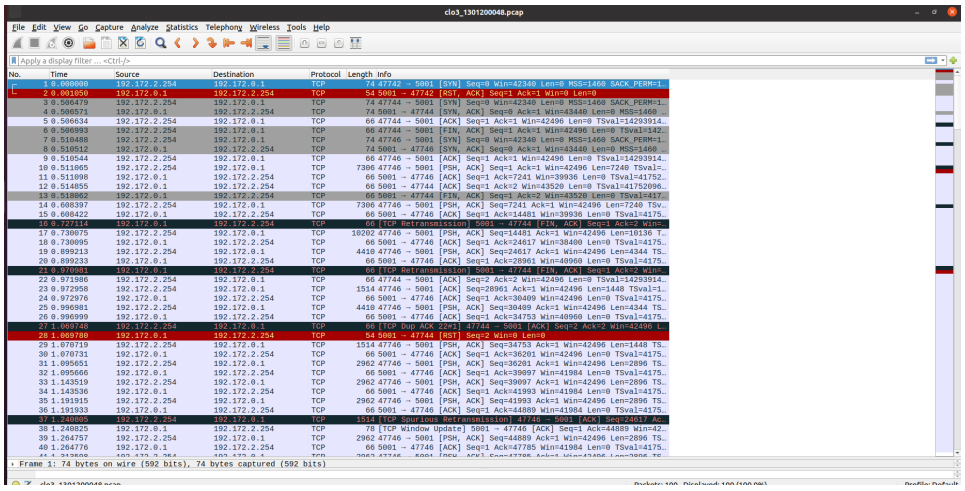
   b. Iperf hostA dengan hostB

```
mininet> xterm hostB
mininet> iperf hostA hostB
*** Iperf: testing TCP bandwidth between hostA and hostB
*** Results: ['478 Kbits/sec', '1.06 Mbits/sec']
```

2) Capture trafik

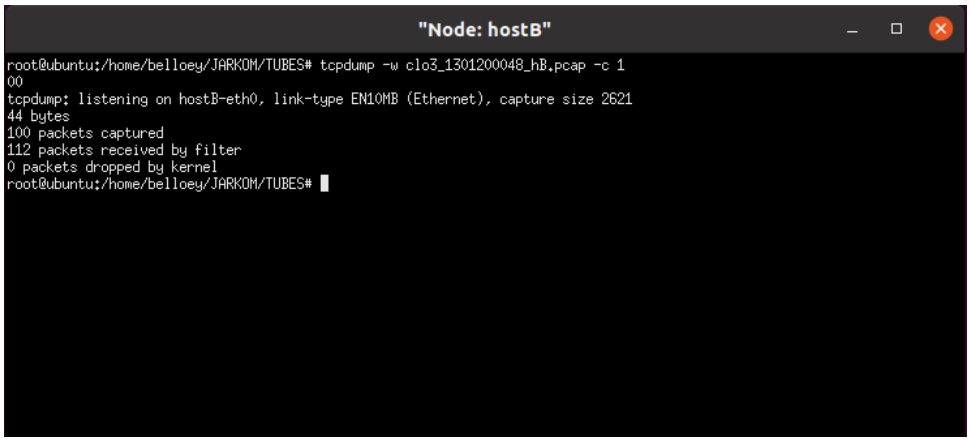
   a. Node : hostA

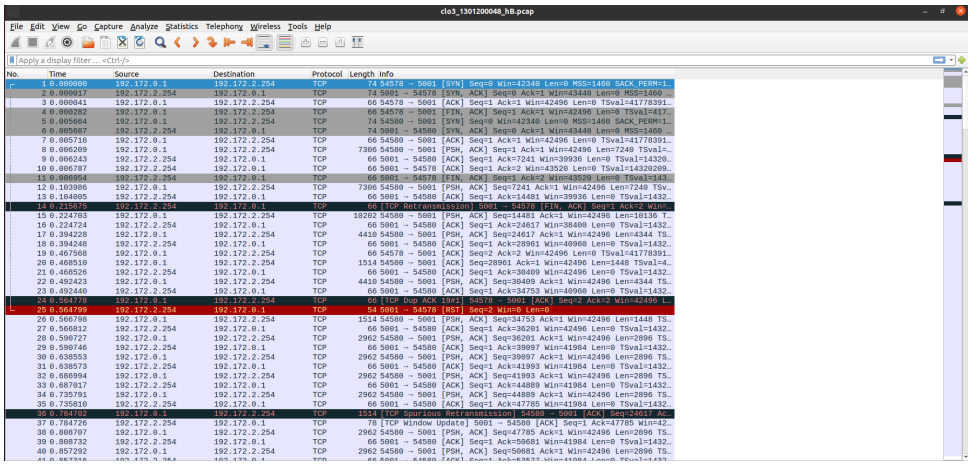
```
"Node: hostA"

root@ubuntu:/home/belloey/JARKOM/TUBES# tcpdump -w clo3_1301200048.pcap -c 100
tcpdump: listening on hostA-eth0, link-type EN10MB (Ethernet), capture size 2621
44 bytes
100 packets captured
127 packets received by filter
0 packets dropped by kernel
root@ubuntu:/home/belloey/JARKOM/TUBES#
```



Capture menggunakan wireshark

b. Node: hostB





Capture menggunakan wireshark

3) Penjelasan Trafik TCP

Trafik TCP merupakan Trafik yang bersifat connection-oriented, yang artinya akan melakukan handshaking terlebih dahulu sebelum mengirimkan paket. Paket yang dikirimkan juga sudah berurutan dengan susunan yang benar, serta mengecek adanya kesalahan dan akan melakukan pengiriman ulang apabila paket rusak atau hilang. Pada hasil capture menggunakan tcpdump dan wireshark di atas, terbukti bahwa trafik yang digunakan adalah TCP (terdapat konfirmasi paket telah sampai dengan flag ACK).

Adapun perbedaannya dengan UDP adalah UDP hanya menekankan kecepatan pengiriman dan tidak memperbaiki apabila terjadi kesalahan (connectionless).

UDP juga tidak terdapat konfirmasi paket telah sampai (karena tidak ada handshaking).

2.2.4. CLO 4

Menginspeksi penggunaan queue pada router jaringan. Ukuran Buffer (max_queue_size) dimanipulasi dengan ukuran 20,40,60, dan 100

1) Ukuran buffer pada router dimanipulasi atau diubah menjadi 20, 40, 60, dan 100.

a. Ukuran buffer = 20

```
# Add Link (Menghubungkan)
net.addLink(hostA, R1, max_queue_size=20, intfName1='hostA-eth0',intfName2='R1-eth0', cls=TCLink, bw=1 )
net.addLink(hostA, R2, max_queue_size=20, intfName1='hostA-eth1',intfName2='R2-eth0', cls=TCLink, bw=1 )
net.addLink(hostB, R3, max_queue_size=20, intfName1='hostB-eth0',intfName2='R3-eth1', cls=TCLink, bw=1 )
net.addLink(hostB, R4, max_queue_size=20, intfName1='hostB-eth1',intfName2='R4-eth1', cls=TCLink, bw=1 )
net.addLink(R1, R3, max_queue_size=20, intfName1='R1-eth1',intfName2='R3-eth0', cls=TCLink, bw=0.5 )
net.addLink(R1, R4, max_queue_size=20, intfName1='R1-eth2',intfName2='R4-eth2', cls=TCLink, bw=1 )
net.addLink(R2, R4, max_queue_size=20, intfName1='R2-eth1',intfName2='R4-eth0', cls=TCLink, bw=0.5 )
net.addLink(R2, R3, max_queue_size=20, intfName1='R2-eth2',intfName2='R3-eth2', cls=TCLink, bw=1 )
```

b. Ukuran buffer = 40

```
# Add Link (Menghubungkan)
net.addLink(hostA, R1, max_queue_size=40, intfName1='hostA-eth0',intfName2='R1-eth0', cls=TCLink, bw=1 )
net.addLink(hostA, R2, max_queue_size=40, intfName1='hostA-eth1',intfName2='R2-eth0', cls=TCLink, bw=1 )
net.addLink(hostB, R3, max_queue_size=40, intfName1='hostB-eth0',intfName2='R3-eth1', cls=TCLink, bw=1 )
net.addLink(hostB, R4, max_queue_size=40, intfName1='hostB-eth1',intfName2='R4-eth1', cls=TCLink, bw=1 )
net.addLink(R1, R3, max_queue_size=40, intfName1='R1-eth1',intfName2='R3-eth0', cls=TCLink, bw=0.5 )
net.addLink(R1, R4, max_queue_size=40, intfName1='R1-eth2',intfName2='R4-eth2', cls=TCLink, bw=1 )
net.addLink(R2, R4, max_queue_size=40, intfName1='R2-eth1',intfName2='R4-eth0', cls=TCLink, bw=0.5 )
net.addLink(R2, R3, max_queue_size=40, intfName1='R2-eth2',intfName2='R3-eth2', cls=TCLink, bw=1 )
```

c. Ukuran buffer = 60

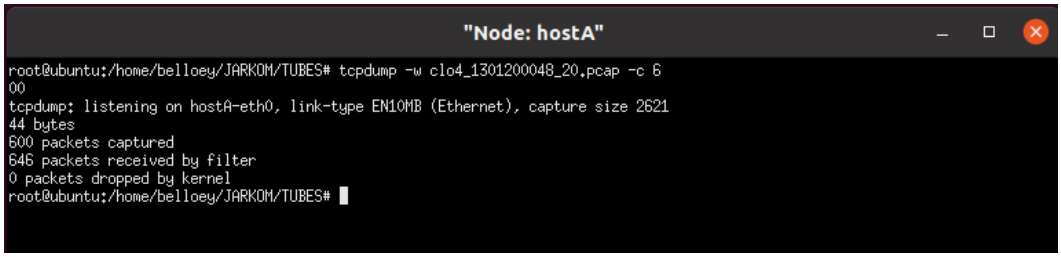
```
# Add Link (Menghubungkan)
net.addLink(hostA, R1, max_queue_size=60, intfName1='hostA-eth0',intfName2='R1-eth0', cls=TCLink, bw=1 )
net.addLink(hostA, R2, max_queue_size=60, intfName1='hostA-eth1',intfName2='R2-eth0', cls=TCLink, bw=1 )
net.addLink(hostB, R3, max_queue_size=60, intfName1='hostB-eth0',intfName2='R3-eth1', cls=TCLink, bw=1 )
net.addLink(hostB, R4, max_queue_size=60, intfName1='hostB-eth1',intfName2='R4-eth1', cls=TCLink, bw=1 )
net.addLink(R1, R3, max_queue_size=60, intfName1='R1-eth1',intfName2='R3-eth0', cls=TCLink, bw=0.5 )
net.addLink(R1, R4, max_queue_size=60, intfName1='R1-eth2',intfName2='R4-eth2', cls=TCLink, bw=1 )
net.addLink(R2, R4, max_queue_size=60, intfName1='R2-eth1',intfName2='R4-eth0', cls=TCLink, bw=0.5 )
net.addLink(R2, R3, max_queue_size=60, intfName1='R2-eth2',intfName2='R3-eth2', cls=TCLink, bw=1 )
```

d. Ukuran buffer = 100

```
# Add Link (Menghubungkan)
net.addLink(hostA, R1, max_queue_size=100, intfName1='hostA-eth0',intfName2='R1-eth0', cls=TCLink, bw=1 )
net.addLink(hostA, R2, max_queue_size=100, intfName1='hostA-eth1',intfName2='R2-eth0', cls=TCLink, bw=1 )
net.addLink(hostB, R3, max_queue_size=100, intfName1='hostB-eth0',intfName2='R3-eth1', cls=TCLink, bw=1 )
net.addLink(hostB, R4, max_queue_size=100, intfName1='hostB-eth1',intfName2='R4-eth1', cls=TCLink, bw=1 )
net.addLink(R1, R3, max_queue_size=100, intfName1='R1-eth1',intfName2='R3-eth0', cls=TCLink, bw=0.5 )
net.addLink(R1, R4, max_queue_size=100, intfName1='R1-eth2',intfName2='R4-eth2', cls=TCLink, bw=1 )
net.addLink(R2, R4, max_queue_size=100, intfName1='R2-eth1',intfName2='R4-eth0', cls=TCLink, bw=0.5 )
net.addLink(R2, R3, max_queue_size=100, intfName1='R2-eth2',intfName2='R3-eth2', cls=TCLink, bw=1 )
```

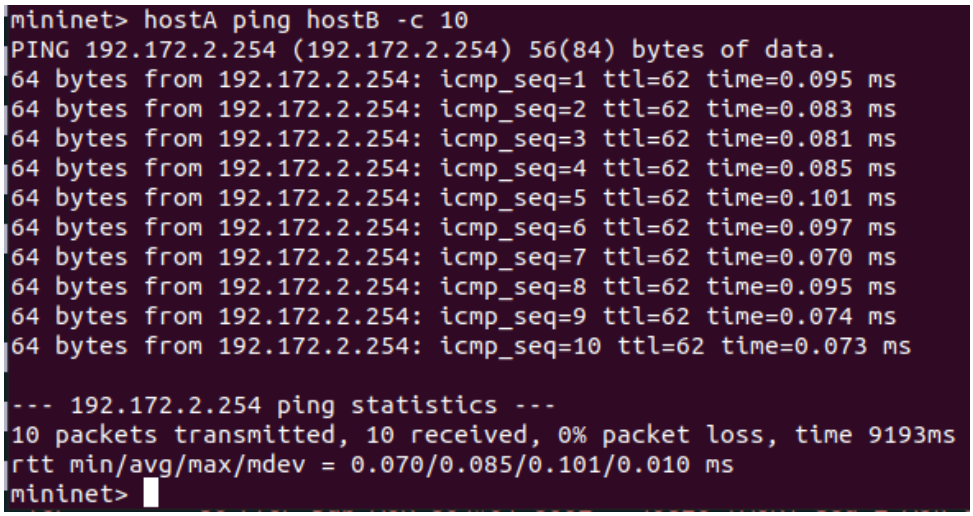
2) Generate *traffic* menggunakan iPerf

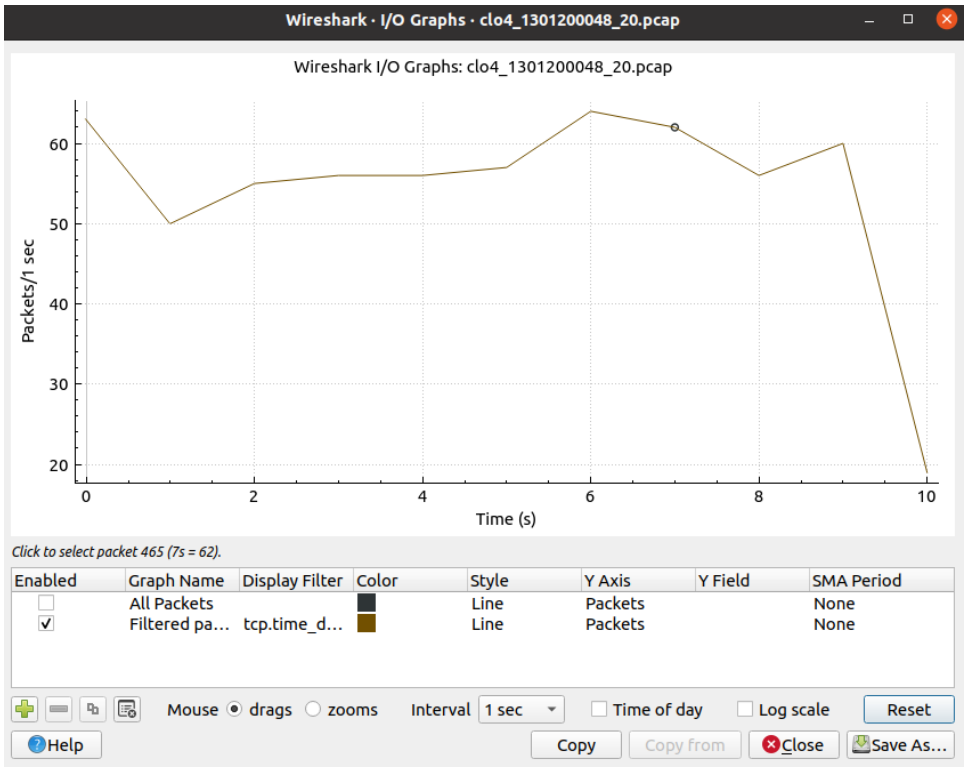
```
mininet> xterm hostA
mininet> iperf hostA hostB
*** Iperf: testing TCP bandwidth between hostA and hostB
*** Results: ['478 Kbits/sec', '1.15 Mbits/sec']
mininet> iperf hostA hostB
*** Iperf: testing TCP bandwidth between hostA and hostB
```



3) Capture pengaruh ukuran buffer terhadap *delay*

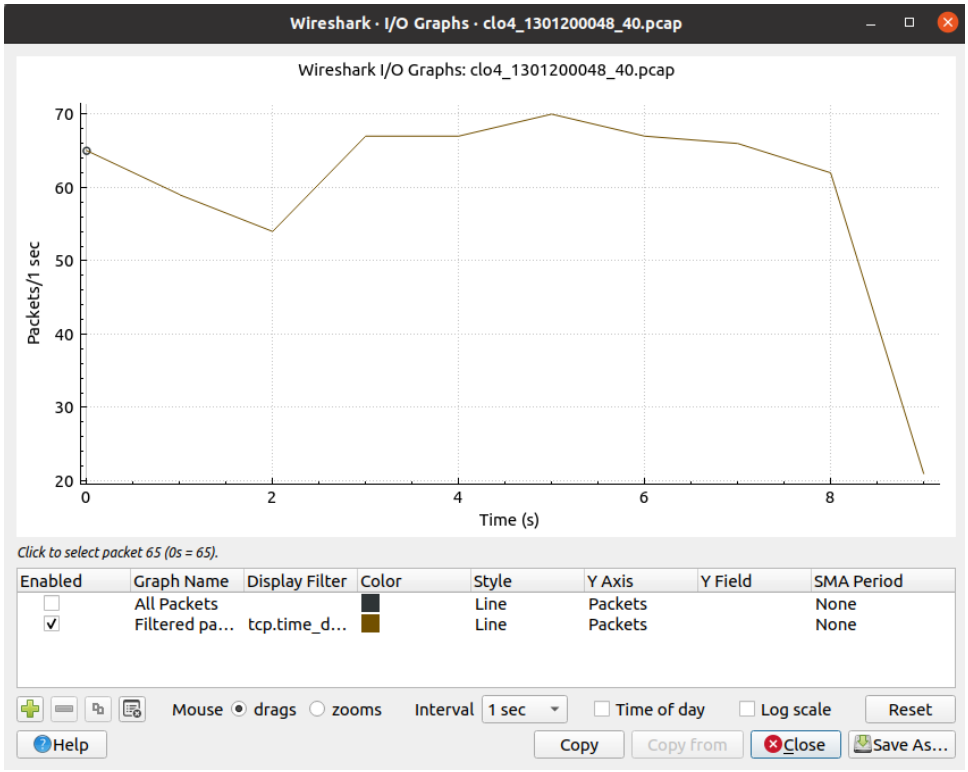
a. Ukuran buffer = 20





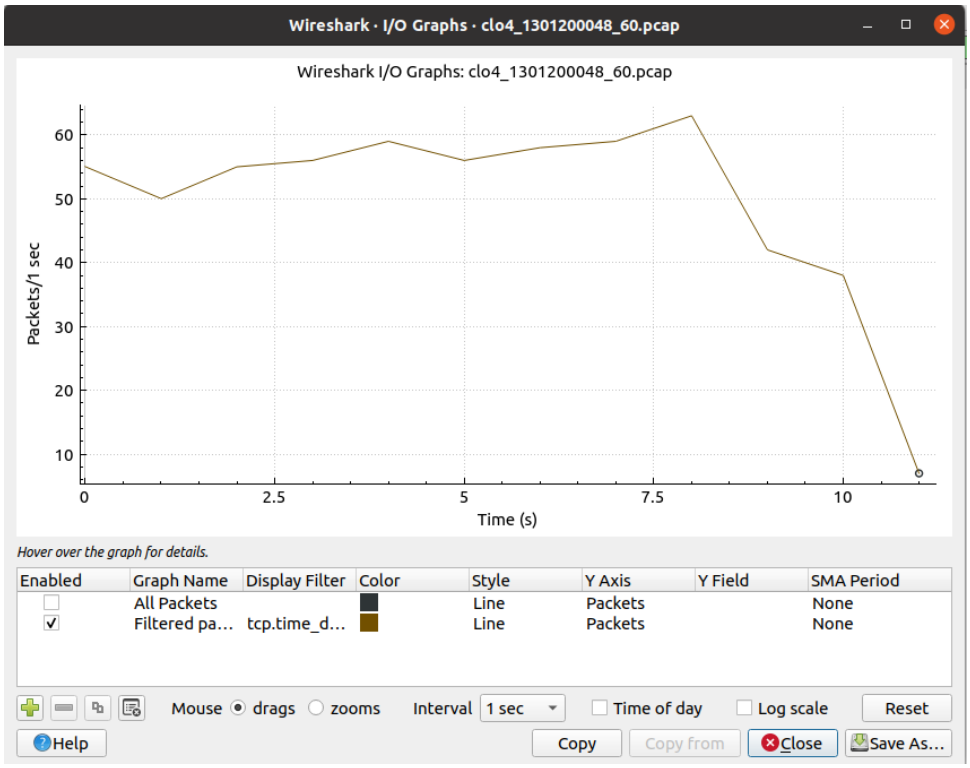
b. Ukuran buffer = 40

```
mininet> hostA ping hostB -c 10
PING 192.172.2.254 (192.172.2.254) 56(84) bytes of data.
64 bytes from 192.172.2.254: icmp_seq=1 ttl=62 time=0.075 ms
64 bytes from 192.172.2.254: icmp_seq=2 ttl=62 time=0.118 ms
64 bytes from 192.172.2.254: icmp_seq=3 ttl=62 time=0.094 ms
64 bytes from 192.172.2.254: icmp_seq=4 ttl=62 time=0.169 ms
64 bytes from 192.172.2.254: icmp_seq=5 ttl=62 time=0.077 ms
64 bytes from 192.172.2.254: icmp_seq=6 ttl=62 time=0.076 ms
64 bytes from 192.172.2.254: icmp_seq=7 ttl=62 time=0.071 ms
64 bytes from 192.172.2.254: icmp_seq=8 ttl=62 time=0.076 ms
64 bytes from 192.172.2.254: icmp_seq=9 ttl=62 time=0.117 ms
64 bytes from 192.172.2.254: icmp_seq=10 ttl=62 time=0.075 ms

--- 192.172.2.254 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9222ms
rtt min/avg/max/mdev = 0.071/0.094/0.169/0.029 ms
mininet>
```



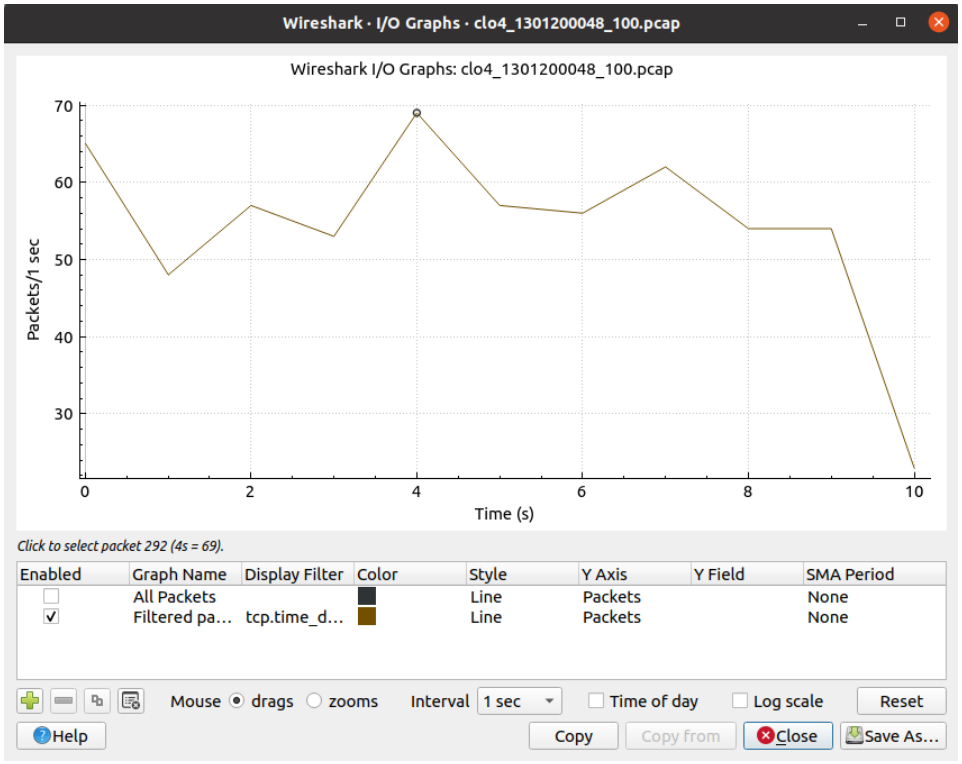
c. Ukuran buffer = 60

```
mininet> hostA ping hostB -c 10
PING 192.172.2.254 (192.172.2.254) 56(84) bytes of data.
64 bytes from 192.172.2.254: icmp_seq=1 ttl=62 time=0.101 ms
64 bytes from 192.172.2.254: icmp_seq=2 ttl=62 time=0.106 ms
64 bytes from 192.172.2.254: icmp_seq=3 ttl=62 time=0.104 ms
64 bytes from 192.172.2.254: icmp_seq=4 ttl=62 time=0.093 ms
64 bytes from 192.172.2.254: icmp_seq=5 ttl=62 time=0.075 ms
64 bytes from 192.172.2.254: icmp_seq=6 ttl=62 time=0.074 ms
64 bytes from 192.172.2.254: icmp_seq=7 ttl=62 time=0.074 ms
64 bytes from 192.172.2.254: icmp_seq=8 ttl=62 time=0.083 ms
64 bytes from 192.172.2.254: icmp_seq=9 ttl=62 time=0.078 ms
64 bytes from 192.172.2.254: icmp_seq=10 ttl=62 time=0.087 ms

--- 192.172.2.254 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9217ms
rtt min/avg/max/mdev = 0.074/0.087/0.106/0.012 ms
mininet>
```



d. Ukuran buffer = 100

```
mininet> hostA ping hostB -c 10
PING 192.172.2.254 (192.172.2.254) 56(84) bytes of data.
64 bytes from 192.172.2.254: icmp_seq=1 ttl=62 time=0.104 ms
64 bytes from 192.172.2.254: icmp_seq=2 ttl=62 time=0.068 ms
64 bytes from 192.172.2.254: icmp_seq=3 ttl=62 time=0.077 ms
64 bytes from 192.172.2.254: icmp_seq=4 ttl=62 time=0.072 ms
64 bytes from 192.172.2.254: icmp_seq=5 ttl=62 time=0.094 ms
64 bytes from 192.172.2.254: icmp_seq=6 ttl=62 time=0.082 ms
64 bytes from 192.172.2.254: icmp_seq=7 ttl=62 time=0.071 ms
64 bytes from 192.172.2.254: icmp_seq=8 ttl=62 time=0.085 ms
64 bytes from 192.172.2.254: icmp_seq=9 ttl=62 time=0.070 ms
64 bytes from 192.172.2.254: icmp_seq=10 ttl=62 time=0.098 ms

--- 192.172.2.254 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9212ms
rtt min/avg/max/mdev = 0.068/0.082/0.104/0.012 ms
mininet>
```



4) Hasil Analisis

Dari capture di atas, dapat disimpulkan bahwa jika buffer sizenya semakin besar maka packet loss time akan semakin tinggi dan jika buffer size semakin kecil maka packet loss time akan lebih rendah. Terlihat juga bahwa semakin besar ukuran buffer nya maka mempunyai kecenderungan lebih lama waktunya, tetapi tidak selalu. Kecepatan running atau lama waktu dipengaruhi juga oleh hardware. Apabila kita menjalankan aplikasi secara bersamaan, otomatis waktunya akan semakin lama.