

LAPORAN TUGAS BESAR JARINGAN KOMPUTER

Ananda Affan Fattahila

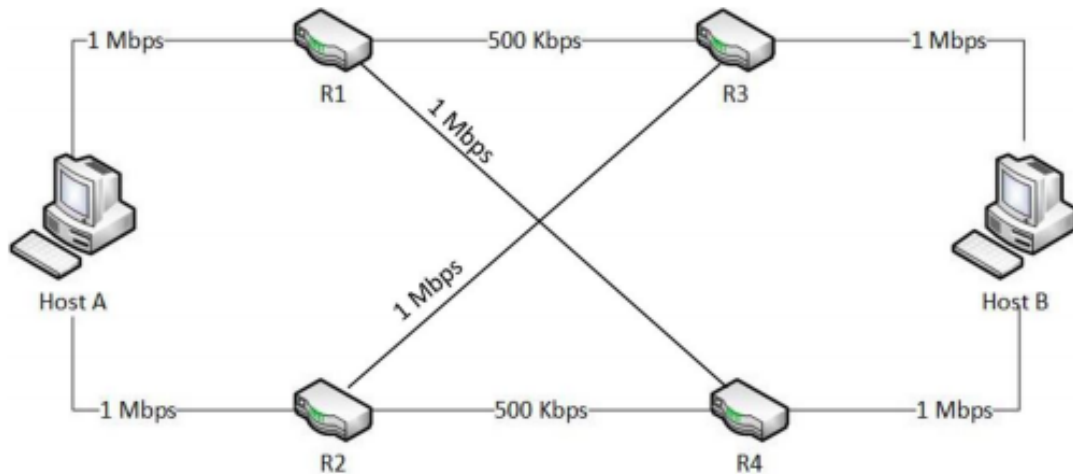
affanfattahila@student.telkomuniversity.ac.id

1301194175

Jaringan Komputer IF-43-02 (CII2J4-IF-43-02)

Informatika, FT, Universitas Telkom. 2021.

Jl. Telekomunikasi Terusan Buah Batu Indonesia 40257, Bandung, Indonesia



Gambar 1 Topologi MPTCP

Note : Kode Program Ada di Point 6 Dokumentasi Tugas Besar (Untuk melihat subnetting,static routing dan juga penerapan penggunaan buffer dengan iperf traffic)

1. GAMBARAN TUGAS

Mata Kuliah Jaringan Komputer terdapat sebuah tugas besar yang memiliki spesifikasi dibawah ini dari Topologi diatas :

CLO 1 GOAL :

- Build topology sesuai dengan soal.
- Desain subnet masing2 network.
- Assign IP sesuai subnet.
- Uji konektivitas dengan ping (yang berada di 1 network)

Penilaian : Penjelasan topologi,Uji konektivitas.

CLO 2 GOAL :

- Mengimplementasikan mekanisme Routing pada topologi yang ada

- Uji konektivitas menggunakan ping. Membuat tabel routing di semua host! Dibuktikan dengan ping antar host. Pake traceroute.

Penilaian : Penjelasan proses routing menggunakan traceroute, Uji konektivitas.

CLO 3 GOAL :

- Membuktikan bahwa TCP telah diimplementasikan dengan benar pada topologi
- Generate trafik dari h1 ke h2 menggunakan iperf.
- Inspeksi trafik pakai wireshark, dibuktikan dengan trafik TCP di wireshark/tcpdump.

Penilaian : Penjelasan generate traffic TCP, Uji konektivitas.

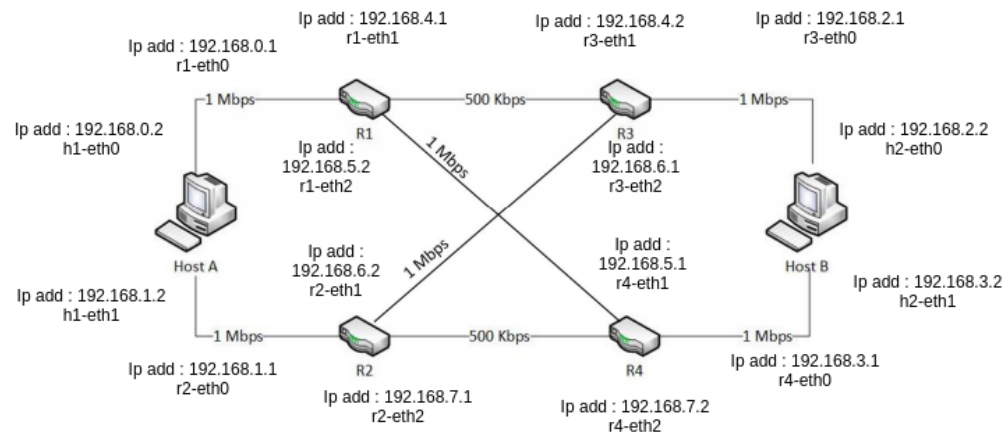
CLO 4 GOAL :

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

Penilaian : Ketepatan konfigurasi besar buffer, Penjelasan pengaruh ukuran buffer.

2. CLO 1

2.1 Penjelasan Desain Network dan Assign Ip (Subnetting)



Looks like those subnets will not fit into that network, but here is something else that may work for you:

Name	Hosts Needed	Hosts Available	Unused Hosts	Network Address	Slash	Mask	Usable Range	Broadcast	Wildcard
Host1-In1	128	254	126	192.168.0.0	/24	255.255.255.0	192.168.0.1 - 192.168.0.254	192.168.0.255	0.0.0.255
Host1-In2	128	254	126	192.168.1.0	/24	255.255.255.0	192.168.1.1 - 192.168.1.254	192.168.1.255	0.0.0.255
Host2-In1	128	254	126	192.168.2.0	/24	255.255.255.0	192.168.2.1 - 192.168.2.254	192.168.2.255	0.0.0.255
Host2-In2	128	254	126	192.168.3.0	/24	255.255.255.0	192.168.3.1 - 192.168.3.254	192.168.3.255	0.0.0.255
net1	128	254	126	192.168.4.0	/24	255.255.255.0	192.168.4.1 - 192.168.4.254	192.168.4.255	0.0.0.255
net2	128	254	126	192.168.5.0	/24	255.255.255.0	192.168.5.1 - 192.168.5.254	192.168.5.255	0.0.0.255
net3	128	254	126	192.168.6.0	/24	255.255.255.0	192.168.6.1 - 192.168.6.254	192.168.6.255	0.0.0.255
net4	128	254	126	192.168.7.0	/24	255.255.255.0	192.168.7.1 - 192.168.7.254	192.168.7.255	0.0.0.255

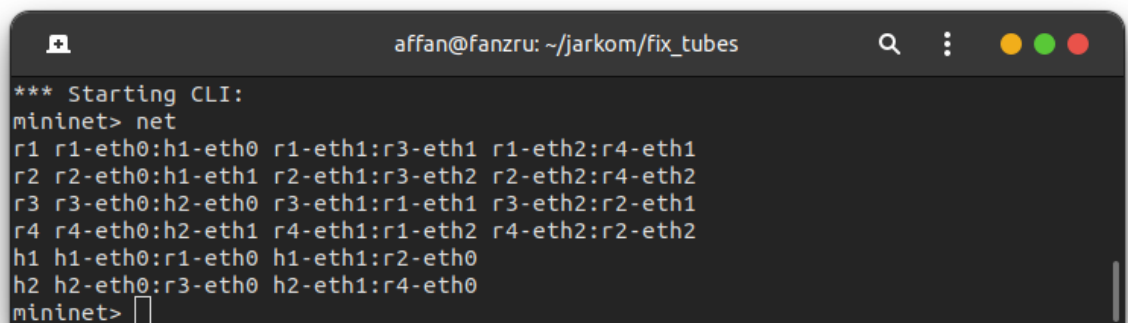
Sub-jaringan, atau subnet, adalah pembagian logis dari suatu jaringan IP. Praktik membagi jaringan menjadi dua atau lebih jaringan disebut subnetting [1]. Berdasarkan hasil konfigurasi yang saya lakukan berdasarkan tabel route. Host A dan B dibuatkan masing - masing dua interface. Untuk menghubungkan antar router di buatkan 4 net.

2.2 Uji Konektivitas Topologi

Uji konektivitas dilakukan untuk membuktikan bahwa koneksi tiap netting sudah berhasil dimasukan pada masing - masing host dengan ethernet tertentu dari setiap host.

- Cek Link

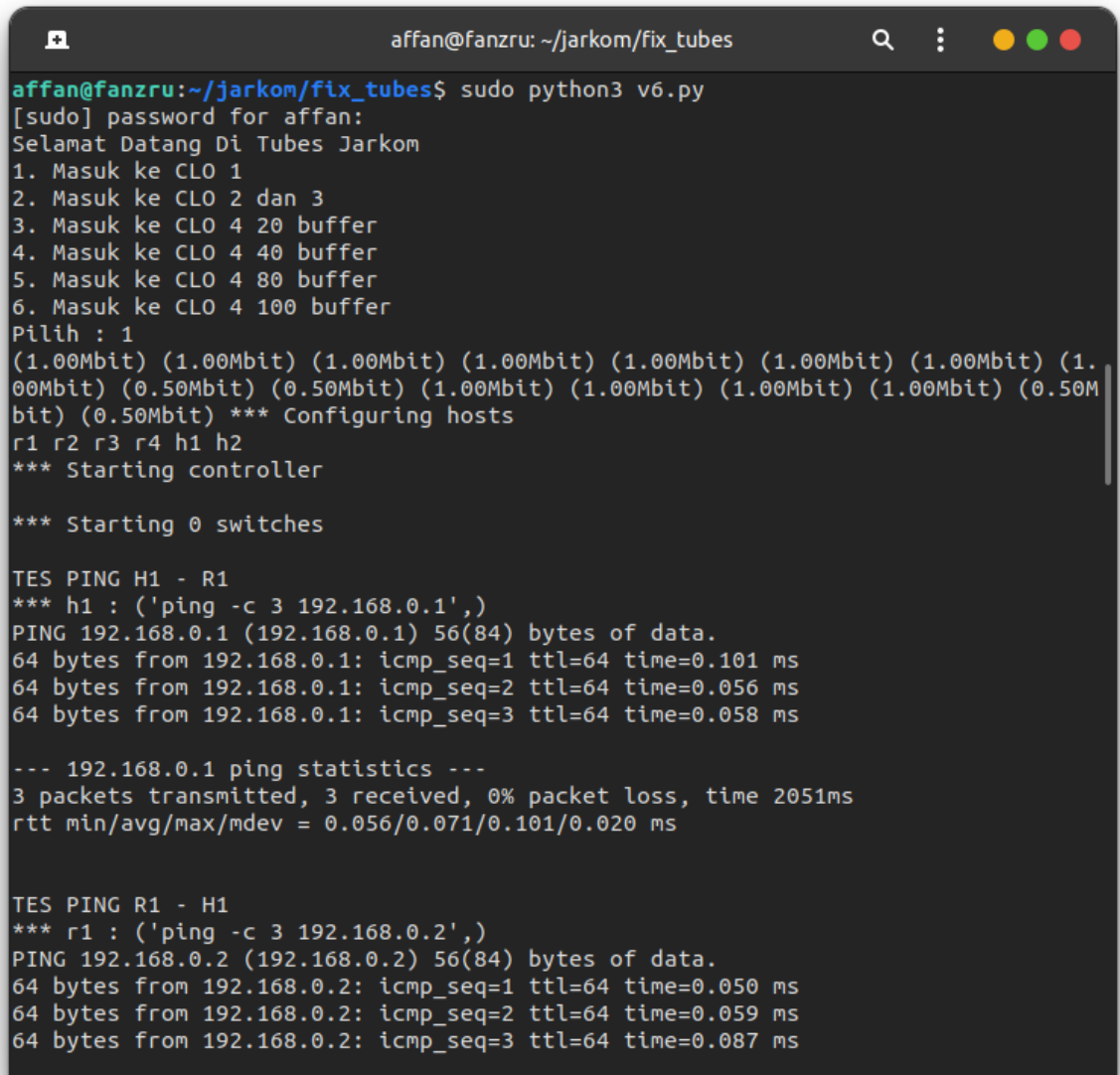
Net berguna untuk melihat link yang menghubungkan antar host network yang sudah di buat. Hasil dari net saya sebagai berikut :



```
affan@fanzru: ~/jarkom/fix_tubes
*** Starting CLI:
mininet> net
r1 r1-eth0:h1-eth0 r1-eth1:r3-eth1 r1-eth2:r4-eth1
r2 r2-eth0:h1-eth1 r2-eth1:r3-eth2 r2-eth2:r4-eth2
r3 r3-eth0:h2-eth0 r3-eth1:r1-eth1 r3-eth2:r2-eth1
r4 r4-eth0:h2-eth1 r4-eth1:r1-eth2 r4-eth2:r2-eth2
h1 h1-eth0:r1-eth0 h1-eth1:r2-eth0
h2 h2-eth0:r3-eth0 h2-eth1:r4-eth0
mininet> 
```

- Tes Ping Setelah AddLink

Screenshot membuktikan hasil ping terhadap topologi yang saya buat dan sudah dilakukan subnetting. Adapun hasilnya sebagai berikut :



```
affan@fanzru: ~/jarkom/fix_tubes
affan@fanzru:~/jarkom/fix_tubes$ sudo python3 v6.py
[sudo] password for affan:
Selamat Datang Di Tubes Jarkom
1. Masuk ke CLO 1
2. Masuk ke CLO 2 dan 3
3. Masuk ke CLO 4 20 buffer
4. Masuk ke CLO 4 40 buffer
5. Masuk ke CLO 4 80 buffer
6. Masuk ke CLO 4 100 buffer
Pilih : 1
(1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (0.50Mbit) (0.50Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (0.50Mbit) (0.50Mbit) *** Configuring hosts
r1 r2 r3 r4 h1 h2
*** Starting controller

*** Starting 0 switches

TES PING H1 - R1
*** h1 : ('ping -c 3 192.168.0.1',)
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp_seq=1 ttl=64 time=0.101 ms
64 bytes from 192.168.0.1: icmp_seq=2 ttl=64 time=0.056 ms
64 bytes from 192.168.0.1: icmp_seq=3 ttl=64 time=0.058 ms

--- 192.168.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2051ms
rtt min/avg/max/mdev = 0.056/0.071/0.101/0.020 ms

TES PING R1 - H1
*** r1 : ('ping -c 3 192.168.0.2',)
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=1 ttl=64 time=0.050 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=64 time=0.059 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=64 time=0.087 ms
```

```
affan@fanzru: ~/jarkom/fix_tubes

TES PING H1 - R2
*** h1 : ('ping -c 3 192.168.1.1',)
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.098 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.060 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.058 ms

--- 192.168.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.058/0.072/0.098/0.018 ms

TES PING R2 - H1
*** r2 : ('ping -c 3 192.168.1.2',)
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=0.062 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.059 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.057 ms

--- 192.168.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.057/0.059/0.062/0.002 ms

TES PING H2 - R3
*** h2 : ('ping -c 3 192.168.2.1',)
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data.
64 bytes from 192.168.2.1: icmp_seq=1 ttl=64 time=0.108 ms
64 bytes from 192.168.2.1: icmp_seq=2 ttl=64 time=0.057 ms
64 bytes from 192.168.2.1: icmp_seq=3 ttl=64 time=0.073 ms

--- 192.168.2.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2049ms
rtt min/avg/max/mdev = 0.057/0.079/0.108/0.021 ms
```

```
affan@fanzru: ~/jarkom/fix_tubes

TES PING R3 - H2
*** r3 : ('ping -c 3 192.168.2.2',)
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=64 time=0.057 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=64 time=0.058 ms
64 bytes from 192.168.2.2: icmp_seq=3 ttl=64 time=0.058 ms

--- 192.168.2.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2041ms
rtt min/avg/max/mdev = 0.057/0.057/0.058/0.000 ms

TES PING H2 - R4
*** h2 : ('ping -c 3 192.168.3.1',)
PING 192.168.3.1 (192.168.3.1) 56(84) bytes of data.
64 bytes from 192.168.3.1: icmp_seq=1 ttl=64 time=0.100 ms
64 bytes from 192.168.3.1: icmp_seq=2 ttl=64 time=0.060 ms
64 bytes from 192.168.3.1: icmp_seq=3 ttl=64 time=0.061 ms

--- 192.168.3.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2046ms
rtt min/avg/max/mdev = 0.060/0.073/0.100/0.018 ms

TES PING R4 - H2
*** r4 : ('ping -c 3 192.168.3.2',)
PING 192.168.3.2 (192.168.3.2) 56(84) bytes of data.
64 bytes from 192.168.3.2: icmp_seq=1 ttl=64 time=0.052 ms
64 bytes from 192.168.3.2: icmp_seq=2 ttl=64 time=0.060 ms
64 bytes from 192.168.3.2: icmp_seq=3 ttl=64 time=0.077 ms

--- 192.168.3.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.052/0.063/0.077/0.010 ms
```

```
affan@fanzru: ~/jarkom/fix_tubes

TES PING R1 - R3
*** r1 : ('ping -c 3 192.168.5.1',)
PING 192.168.5.1 (192.168.5.1) 56(84) bytes of data.
64 bytes from 192.168.5.1: icmp_seq=1 ttl=64 time=0.078 ms
64 bytes from 192.168.5.1: icmp_seq=2 ttl=64 time=0.070 ms
64 bytes from 192.168.5.1: icmp_seq=3 ttl=64 time=0.058 ms

--- 192.168.5.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.058/0.068/0.078/0.008 ms

TES PING R3 - R1
*** r3 : ('ping -c 3 192.168.5.2',)
ping: connect: Network is unreachable

TES PING R1 - R4
*** r1 : ('ping -c 3 192.168.4.2',)
PING 192.168.4.2 (192.168.4.2) 56(84) bytes of data.
64 bytes from 192.168.4.2: icmp_seq=1 ttl=64 time=0.080 ms
64 bytes from 192.168.4.2: icmp_seq=2 ttl=64 time=0.074 ms
64 bytes from 192.168.4.2: icmp_seq=3 ttl=64 time=0.059 ms

--- 192.168.4.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2043ms
rtt min/avg/max/mdev = 0.059/0.071/0.080/0.008 ms

TES PING R4 - R1
*** r4 : ('ping -c 3 192.168.4.1',)
ping: connect: Network is unreachable

TES PING R2 - R3
```

```
affan@fanzru: ~/jarkom/fix_tubes

TES PING R2 - R3
*** r2 : ('ping -c 3 192.168.6.1',)
PING 192.168.6.1 (192.168.6.1) 56(84) bytes of data.
64 bytes from 192.168.6.1: icmp_seq=1 ttl=64 time=0.094 ms
64 bytes from 192.168.6.1: icmp_seq=2 ttl=64 time=0.078 ms
64 bytes from 192.168.6.1: icmp_seq=3 ttl=64 time=0.060 ms

--- 192.168.6.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2043ms
rtt min/avg/max/mdev = 0.060/0.077/0.094/0.013 ms

TES PING R3 - R2
*** r3 : ('ping -c 3 192.168.6.2',)
PING 192.168.6.2 (192.168.6.2) 56(84) bytes of data.
64 bytes from 192.168.6.2: icmp_seq=1 ttl=64 time=0.052 ms
64 bytes from 192.168.6.2: icmp_seq=2 ttl=64 time=0.072 ms
64 bytes from 192.168.6.2: icmp_seq=3 ttl=64 time=0.076 ms

--- 192.168.6.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.052/0.066/0.076/0.010 ms

TES PING R2 - R4
*** r2 : ('ping -c 3 192.168.7.1',)
PING 192.168.7.1 (192.168.7.1) 56(84) bytes of data.
64 bytes from 192.168.7.1: icmp_seq=1 ttl=64 time=0.044 ms
64 bytes from 192.168.7.1: icmp_seq=2 ttl=64 time=0.057 ms
64 bytes from 192.168.7.1: icmp_seq=3 ttl=64 time=0.041 ms

--- 192.168.7.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.041/0.047/0.057/0.007 ms

TES PING R4 - R2
```



```
affan@fanzru: ~/jarkom/fix_tubes

3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.052/0.066/0.076/0.010 ms

TES PING R2 - R4
*** r2 : ('ping -c 3 192.168.7.1',)
PING 192.168.7.1 (192.168.7.1) 56(84) bytes of data.
64 bytes from 192.168.7.1: icmp_seq=1 ttl=64 time=0.044 ms
64 bytes from 192.168.7.1: icmp_seq=2 ttl=64 time=0.057 ms
64 bytes from 192.168.7.1: icmp_seq=3 ttl=64 time=0.041 ms

--- 192.168.7.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.041/0.047/0.057/0.007 ms

TES PING R4 - R2
*** r4 : ('ping -c 3 192.168.7.2',)
PING 192.168.7.2 (192.168.7.2) 56(84) bytes of data.
64 bytes from 192.168.7.2: icmp_seq=1 ttl=64 time=0.043 ms
64 bytes from 192.168.7.2: icmp_seq=2 ttl=64 time=0.048 ms
64 bytes from 192.168.7.2: icmp_seq=3 ttl=64 time=0.063 ms

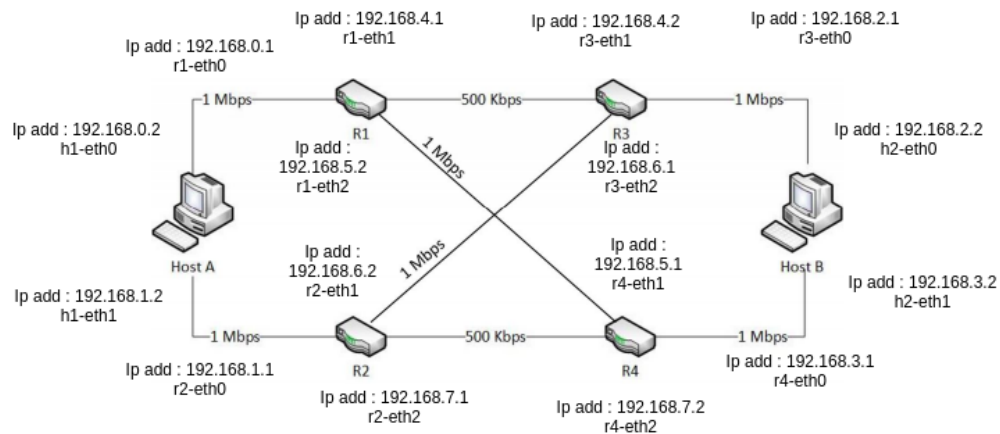
--- 192.168.7.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.043/0.051/0.063/0.008 ms

*** Starting CLI:
mininet> net
r1 r1-eth0:h1-eth0 r1-eth1:r3-eth1 r1-eth2:r4-eth1
r2 r2-eth0:h1-eth1 r2-eth1:r3-eth2 r2-eth2:r4-eth2
r3 r3-eth0:h2-eth0 r3-eth1:r1-eth1 r3-eth2:r2-eth1
r4 r4-eth0:h2-eth1 r4-eth1:r1-eth2 r4-eth2:r2-eth2
h1 h1-eth0:r1-eth0 h1-eth1:r2-eth0
h2 h2-eth0:r3-eth0 h2-eth1:r4-eth0
mininet> 
```

3. CLO 2

2.1 Tabel Routing

Static Routing adalah routing yang dilakukan secara manual. Setiap jaringan yang akan di routing harus dikonfigurasi satu persatu oleh administrator jaringan. Kelebihan dari static routing adalah lebih aman serta tidak membutuhkan sumber daya yang besar [2]. Tabel dibawah ini menjadi acuan untuk melakukan routing agar bisa menjangkau host yang lain tidak hanya host yang terhubung dengan 1 net.



Looks like those subnets will not fit into that network, but here is something else that may work for you:

Name	Hosts Needed	Hosts Available	Unused Hosts	Network Address	Slash	Mask	Usable Range	Broadcast	Wildcard
Host1-In1	128	254	126	192.168.0.0	/24	255.255.255.0	192.168.0.1 - 192.168.0.254	192.168.0.255	0.0.0.255
Host1-In2	128	254	126	192.168.1.0	/24	255.255.255.0	192.168.1.1 - 192.168.1.254	192.168.1.255	0.0.0.255
Host2-In1	128	254	126	192.168.2.0	/24	255.255.255.0	192.168.2.1 - 192.168.2.254	192.168.2.255	0.0.0.255
Host2-In2	128	254	126	192.168.3.0	/24	255.255.255.0	192.168.3.1 - 192.168.3.254	192.168.3.255	0.0.0.255
net1	128	254	126	192.168.4.0	/24	255.255.255.0	192.168.4.1 - 192.168.4.254	192.168.4.255	0.0.0.255
net2	128	254	126	192.168.5.0	/24	255.255.255.0	192.168.5.1 - 192.168.5.254	192.168.5.255	0.0.0.255
net3	128	254	126	192.168.6.0	/24	255.255.255.0	192.168.6.1 - 192.168.6.254	192.168.6.255	0.0.0.255
net4	128	254	126	192.168.7.0	/24	255.255.255.0	192.168.7.1 - 192.168.7.254	192.168.7.255	0.0.0.255

2.2 Penjelasan Proses Routing Menggunakan Traceroute

Traceroute (Tracert) adalah perintah untuk menunjukkan rute yang dilewati paket untuk mencapai tujuan. Ini dilakukan dengan mengirim pesan Internet Control Message Protocol (ICMP) Echo Request Ke tujuan dengan nilai Time to Live yang semakin meningkat. Rute yang ditampilkan adalah daftar interface router (yang paling dekat dengan host) yang terdapat pada jalur antara host dan tujuan [3]. Berikut adalah hasil traceroute pada topologi dengan static routing yang saya buat pada h1 terhadap h2 :

```

affan@fanzru: ~/jarkom/fix_tubes
mininet> h1 traceroute h2
traceroute to 192.168.2.2 (192.168.2.2), 30 hops max, 60 byte packets
 1  192.168.0.1 (192.168.0.1)  0.064 ms  0.015 ms  0.013 ms
 2  192.168.4.2 (192.168.4.2)  0.037 ms  0.022 ms  0.021 ms
 3  192.168.2.2 (192.168.2.2)  0.048 ms  0.037 ms  0.033 ms
mininet>

```

Dimana untuk menjangkau h2 dari h1 harus melalui IP 192.168.0.1 lalu IP 192.168.4.2 dan berakhir di IP 192.168.2.2. Dan artinya traceroute berhasil.

2.3 Uji konektivitas Setelah Routing

Berikut ini merupakan hasil ping dari topologi dengan static routing yang sudah saya buat :

```
affan@fanzru: ~/jarkom/fix_tubes
*** Done
affan@fanzru:~/jarkom/fix_tubes$ sudo python3 v6.py
[sudo] password for affan:
Selamat Datang Di Tubes Jarkom
1. Masuk ke CLO 1
2. Masuk ke CLO 2 dan 3
3. Masuk ke CLO 4 20 buffer
4. Masuk ke CLO 4 40 buffer
5. Masuk ke CLO 4 80 buffer
6. Masuk ke CLO 4 100 buffer
Pilih : 2
(1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (0.50Mbit) (0.50Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (1.00Mbit) (0.50Mbit) (0.50Mbit) *** Configuring hosts
r1 r2 r3 r4 h1 h2
*** Starting controller

*** Starting 0 switches

TES PING R1 - R2
*** r1 : ('ping -c 3 192.168.6.2',)
PING 192.168.6.2 (192.168.6.2) 56(84) bytes of data.
64 bytes from 192.168.6.2: icmp_seq=1 ttl=63 time=0.141 ms
64 bytes from 192.168.6.2: icmp_seq=2 ttl=63 time=0.079 ms
64 bytes from 192.168.6.2: icmp_seq=3 ttl=63 time=0.079 ms

--- 192.168.6.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2050ms
rtt min/avg/max/mdev = 0.079/0.099/0.141/0.029 ms

TES PING R3 - R4
*** r3 : ('ping -c 3 192.168.4.2',)
PING 192.168.4.2 (192.168.4.2) 56(84) bytes of data.
64 bytes from 192.168.4.2: icmp_seq=1 ttl=64 time=0.055 ms
64 bytes from 192.168.4.2: icmp_seq=2 ttl=64 time=0.043 ms
64 bytes from 192.168.4.2: icmp_seq=3 ttl=64 time=0.069 ms
```

```
affan@fanzru: ~/jarkom/fix_tubes

rtt min/avg/max/mdev = 0.043/0.055/0.069/0.010 ms

TES PING H1 - H2 interface Atas
*** h1 : ('ping -c 3 192.168.2.2',)
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=62 time=0.233 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=62 time=0.112 ms
64 bytes from 192.168.2.2: icmp_seq=3 ttl=62 time=0.106 ms

--- 192.168.2.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.106/0.150/0.233/0.058 ms

TES PING H1- H2 interface Bawah
*** h1 : ('ping -c 3 192.168.3.2',)
PING 192.168.3.2 (192.168.3.2) 56(84) bytes of data.
64 bytes from 192.168.3.2: icmp_seq=1 ttl=62 time=0.125 ms
64 bytes from 192.168.3.2: icmp_seq=2 ttl=62 time=0.119 ms
64 bytes from 192.168.3.2: icmp_seq=3 ttl=62 time=0.120 ms

--- 192.168.3.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2049ms
rtt min/avg/max/mdev = 0.119/0.121/0.125/0.002 ms

TES PING H2 - H1 interface Atas
*** h2 : ('ping -c 3 192.168.0.2',)
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=1 ttl=62 time=0.120 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=62 time=0.083 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=62 time=0.129 ms

--- 192.168.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2041ms
rtt min/avg/max/mdev = 0.083/0.110/0.129/0.019 ms
```

```
affan@fanzru: ~/jarkom/fix_tubes

*** h1 : ('ping -c 3 192.168.3.2',)
PING 192.168.3.2 (192.168.3.2) 56(84) bytes of data.
64 bytes from 192.168.3.2: icmp_seq=1 ttl=62 time=0.125 ms
64 bytes from 192.168.3.2: icmp_seq=2 ttl=62 time=0.119 ms
64 bytes from 192.168.3.2: icmp_seq=3 ttl=62 time=0.120 ms

--- 192.168.3.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2049ms
rtt min/avg/max/mdev = 0.119/0.121/0.125/0.002 ms

TES PING H2 - H1 interface Atas
*** h2 : ('ping -c 3 192.168.0.2',)
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=1 ttl=62 time=0.120 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=62 time=0.083 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=62 time=0.129 ms

--- 192.168.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2041ms
rtt min/avg/max/mdev = 0.083/0.110/0.129/0.019 ms

TES PING H2- H1 interface Bawah
*** h2 : ('ping -c 3 192.168.1.2',)
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=62 time=0.111 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=62 time=0.161 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=62 time=0.100 ms

--- 192.168.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.100/0.124/0.161/0.026 ms

*** Starting CLI:
mininet>
```

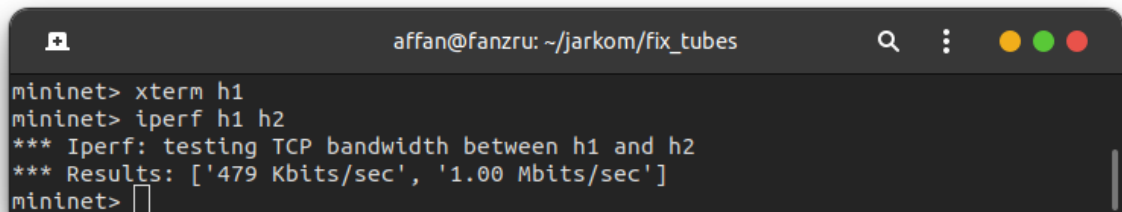
```
affan@fanzru: ~/jarkom/fix_tubes

mininet> pingall
*** Ping: testing ping reachability
r1 -> r2 r3 r4 h1 h2
r2 -> r1 r3 r4 h1 h2
r3 -> r1 r2 r4 h1 h2
r4 -> r1 r2 r3 h1 h2
h1 -> r1 r2 r3 r4 h2
h2 -> r1 r2 r3 r4 h1
*** Results: 0% dropped (30/30 received)
mininet>
```

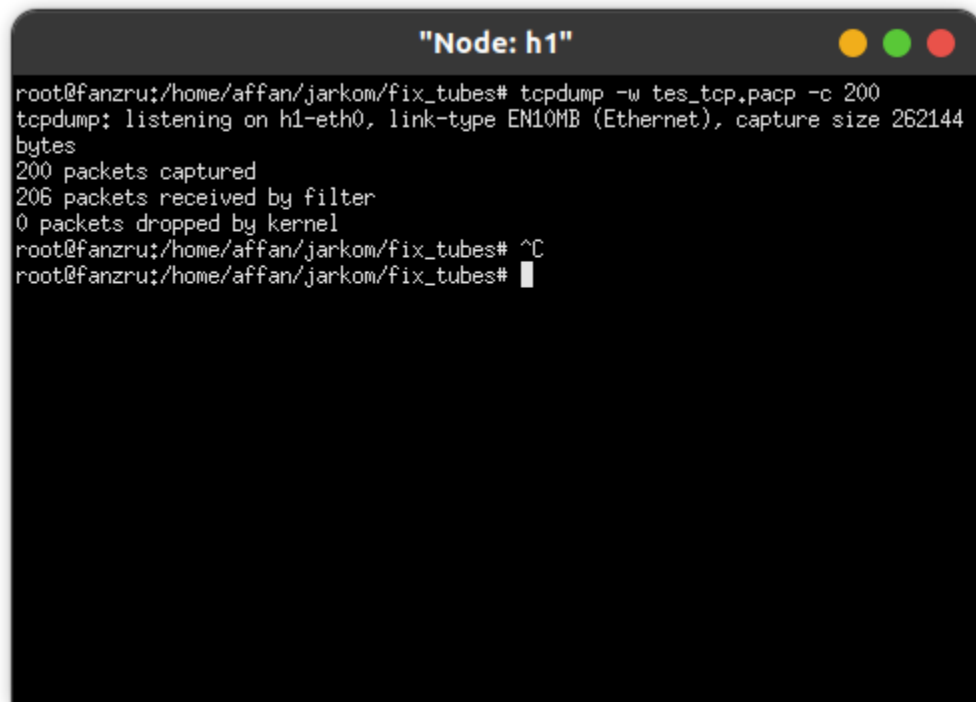
4. CLO 3

3.1 Penjelasan generate traffic TCP

Protokol Kendali Transmisi atau Transmission Control Protocol (TCP) adalah suatu protokol yang berada di lapisan transport (baik itu dalam tujuh lapis model referensi OSI atau model DARPA) yang berorientasi sambungan (connection-oriented) dan dapat diandalkan (reliable). . network traffic itu mengacu sekumpulan data yang bergerak di jaringan pada titik waktu tertentu. Lalu lintas jaringan merupakan komponen utama untuk pengukuran lalu kontrol lalu lintas, lintas jaringan, dan simulasi [5]. Adapun hasil atau cara untuk Inspeksi trafik pakai wireshark, dibuktikan dengan trafik TCP di wireshark/tcpdump adalah sebagai berikut :



```
affan@fanzru: ~/jarkom/fix_tubes
mininet> xterm h1
mininet> iperf h1 h2
*** Iperf: testing TCP bandwidth between h1 and h2
*** Results: ['479 Kbits/sec', '1.00 Mbits/sec']
mininet> 
```



```
"Node: h1"
root@fanzru:/home/affan/jarkom/fix_tubes# tcpdump -w tes_tcp.pcap -c 200
tcpdump: listening on h1-eth0, link-type EN10MB (Ethernet), capture size 262144
bytes
200 packets captured
206 packets received by filter
0 packets dropped by kernel
root@fanzru:/home/affan/jarkom/fix_tubes# ^C
root@fanzru:/home/affan/jarkom/fix_tubes# 
```

Hasil dari TCP Inspeksi trafik pakai tcpdump :

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.2	192.168.2.2	TCP	74	37172 → 5001 [SYN] Seq=0 Win=4234
2	0.000097	192.168.2.2	192.168.0.2	TCP	74	5001 → 37172 [SYN, ACK] Seq=0 Ack=
3	0.000121	192.168.0.2	192.168.2.2	TCP	66	37172 → 5001 [ACK] Seq=1 Ack=1 Wi
4	0.000217	192.168.0.2	192.168.2.2	TCP	66	37172 → 5001 [FIN, ACK] Seq=1 Ack=
5	0.004079	192.168.0.2	192.168.2.2	TCP	74	37174 → 5001 [SYN] Seq=0 Win=4234
6	0.004124	192.168.2.2	192.168.0.2	TCP	66	5001 → 37172 [ACK] Seq=1 Ack=2 Wi
7	0.004151	192.168.2.2	192.168.0.2	TCP	74	5001 → 37174 [SYN, ACK] Seq=8 Ack=
8	0.004165	192.168.0.2	192.168.2.2	TCP	66	37174 → 5001 [ACK] Seq=1 Ack=1 Wi
9	0.004367	192.168.0.2	192.168.2.2	TCP	7306	37174 → 5001 [PSH, ACK] Seq=1 Ack=
10	0.004397	192.168.2.2	192.168.0.2	TCP	66	5001 → 37174 [ACK] Seq=1 Ack=7241

Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
 Ethernet II, Src: 46:5a:5e:38:5d:cd (46:5a:5e:38:5d:cd), Dst: b2:09:7b:24:23:7f (b2:09:7b:24:23:7f)
 Internet Protocol Version 4, Src: 192.168.0.2, Dst: 192.168.2.2
 Transmission Control Protocol, Src Port: 37172, Dst Port: 5001, Seq: 0, Len: 0

0000 b2 09 7b 24 23 7f 46 5a 5e 38 5d cd 08 00 45 10 ..{S#FZ^8]...E-
 0010 00 3c 6f bd 40 00 40 06 47 9a c0 a8 00 02 c0 a8 <0@0 G.....
 0020 02 02 91 34 13 89 98 32 0f e3 00 00 00 00 a0 02 ...4...2.....
 0030 a5 64 83 83 00 00 02 04 05 b4 04 02 08 0a 85 b1 -d.....
 0040 a7 21 00 00 00 00 01 03 03 09 -!.....

3.2 Uji konektivitas

```

affan@fanzru: ~/jarkom/fix_tubes
mininet> pingall
*** Ping: testing ping reachability
r1 -> r2 r3 r4 h1 h2
r2 -> r1 r3 r4 h1 h2
r3 -> r1 r2 r4 h1 h2
r4 -> r1 r2 r3 h1 h2
h1 -> r1 r2 r3 r4 h2
h2 -> r1 r2 r3 r4 h1
*** Results: 0% dropped (30/30 received)
mininet>

```

5. CLO 4

Algoritma queue adalah sebuah pengatur pada node dalam jaringan komunikasi packet switching. Network Scheduler mengatur sequence dari paket-paket jaringan dalam antrian dikirim dan diterima dari interface jaringan. Logika dari Network Scheduler memutuskan untuk memilih paket yang mana untuk diteruskan terlebih dahulu. Sistem mungkin memiliki satu atau lebih antrian yang mungkin menyimpan paket dalam satu alur, klasifikasi, atau prioritas.

HTB atau Hierarchical Token Bucket merupakan pengganti antrian yang lebih bagus dari CBQ di Linux ini berguna untuk membatasi kecepatan unduh atau unggah klien sehingga klien terbatas tidak dapat memenuhi total bandwidth. HTB mengimplementasikan mekanisme queuing classful untuk sistem kontrol lalu lintas linux, dan menyediakan rate dan ceil untuk memungkinkan pengguna untuk mengontrol bandwidth absolut ke kelas tertentu dari lalu lintas serta menunjukkan rasio distribusi bandwidth ketika bandwidth ekstra menjadi tersedia.

4.1 Ketepatan konfigurasi besar buffer

Contoh penerapan buffer :

```
net.addLink(h1,r1,max_queue_size=buffer,use_htb=True,intfName1='h1-eth0',intfName2='r1-eth0',cls = TCLink,bw=1)

net.addLink(h1,r2,max_queue_size=buffer,use_htb=True,intfName1='h1-eth1',intfName2='r2-eth0',cls = TCLink,bw=1)

net.addLink(h2,r3,max_queue_size=buffer,use_htb=True,intfName1='h2-eth0',intfName2='r3-eth0',cls = TCLink,bw=1)

net.addLink(h2,r4,max_queue_size=buffer,use_htb=True,intfName1='h2-eth1',intfName2='r4-eth0',cls = TCLink,bw=1)

net.addLink(r1,r3,max_queue_size=buffer,use_htb=True,intfName1='r1-eth1',intfName2='r3-eth1',cls = TCLink,bw=0.5)

net.addLink(r1,r4,max_queue_size=buffer,use_htb=True,intfName1='r1-eth2',intfName2='r4-eth1',cls = TCLink,bw=1)

net.addLink(r2,r3,max_queue_size=buffer,use_htb=True,intfName1='r2-eth1',intfName2='r3-eth2',cls = TCLink,bw=1)

net.addLink(r2,r4,max_queue_size=buffer,use_htb=True,intfName1='r2-eth2',intfName2='r4-eth2',cls = TCLink,bw=0.5)
```


4.2 Penjelasan pengaruh ukuran buffer

Dalam penerapannya seperti saat saya di dunia nyata (implementasi saya saat membuat topologi RT/RW net) buffer digunakan sebagai pengatur limit bandwidth, Buffer sendiri adalah area memori yang menyimpan data ketika mereka sedang dipindahkan antara dua device atau antara device dan aplikasi.

Dimana pada grafik itu menyatakan bahwa semakin besar buffer yang digunakan memakan waktu yang lebih banyak untuk mengirimkan paket. Maka dari itu buffer digunakan untuk mengontrol bandwidth dari pengiriman paket. Semakin besar bandwidth maka diperlukanlah analisa dalam menentukan prioritas packet. Lalu dapat dilihat hasil running di atas bandwidth 20, 40, 60 dan seratus semakin besar buffernya semakin besar juga ping dan delay pengiriman paket bandwidth.

- Buffer 20

```
affan@fanzru: ~/jarkom/fix_tubes
Client connecting to 192.168.2.2, TCP port 5001
Binding to local address 192.162.0.2
TCP window size: 85.3 KByte (default)
-----
[ 3] local 192.168.0.2 port 42446 connected with 192.168.2.2 port 5001
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=62 time=890 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=62 time=860 ms
64 bytes from 192.168.2.2: icmp_seq=6 ttl=62 time=512 ms
64 bytes from 192.168.2.2: icmp_seq=7 ttl=62 time=605 ms
64 bytes from 192.168.2.2: icmp_seq=8 ttl=62 time=599 ms
64 bytes from 192.168.2.2: icmp_seq=9 ttl=62 time=619 ms
64 bytes from 192.168.2.2: icmp_seq=10 ttl=62 time=662 ms
64 bytes from 192.168.2.2: icmp_seq=13 ttl=62 time=772 ms
64 bytes from 192.168.2.2: icmp_seq=14 ttl=62 time=599 ms
64 bytes from 192.168.2.2: icmp_seq=15 ttl=62 time=619 ms
^C
--- 192.168.2.2 ping statistics ---
15 packets transmitted, 10 received, 33,333% packet loss, time 14096ms
rtt min/avg/max/mdev = 512.065/673.623/889.605/118.023 ms
mininet>
```

```
affan@fanzru: ~/jarkom/fix_tubes
mininet> h1 iperf -i 1 -c 192.168.2.2
[ ID] Interval          Transfer      Bandwidth
[ 3] 0.0-19.9 sec    1.50 MBytes   631 Kbits/sec
-----
Client connecting to 192.168.2.2, TCP port 5001
TCP window size: 128 KByte (default)
-----
[ 3] local 192.168.0.2 port 42454 connected with 192.168.2.2 port 5001
[ ID] Interval          Transfer      Bandwidth
[ 3] 0.0- 1.0 sec     438 KBytes    3.59 Mbits/sec
[ 3] 1.0- 2.0 sec     123 KBytes    1.01 Mbits/sec
[ 3] 2.0- 3.0 sec     89.1 KBytes    730 Kbits/sec
[ 3] 3.0- 4.0 sec       0.00 Bytes     0.00 bits/sec
[ 3] 4.0- 5.0 sec     63.6 KBytes    521 Kbits/sec
[ 3] 5.0- 6.0 sec     127 KBytes    1.04 Mbits/sec
[ 3] 6.0- 7.0 sec     63.6 KBytes    521 Kbits/sec
[ 3] 7.0- 8.0 sec     63.6 KBytes    521 Kbits/sec
[ 3] 8.0- 9.0 sec     63.6 KBytes    521 Kbits/sec
[ 3] 9.0-10.0 sec       0.00 Bytes     0.00 bits/sec
[ 3] 0.0-10.1 sec    1.01 MBytes    839 Kbits/sec
mininet>
```

- Buffer 40

```
affan@fanzru: ~/jarkom/fix_tubes
18 packets transmitted, 12 received, 33,3333% packet loss, time 17316ms
rtt min/avg/max/mdev = 0.095/301.036/1240.233/442.403 ms, pipe 2
mininet> h1 iperf -i 1 -c 192.168.2.2
-----
Client connecting to 192.168.2.2, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 3] local 192.168.0.2 port 42466 connected with 192.168.2.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0- 1.0 sec   481 KBytes  3.94 Mbits/sec
[ 3] 1.0- 2.0 sec  84.8 KBytes  695 Kbits/sec
[ 3] 2.0- 3.0 sec  42.4 KBytes  348 Kbits/sec
[ 3] 3.0- 4.0 sec  127 KBytes  1.04 Mbits/sec
[ 3] 4.0- 5.0 sec  63.6 KBytes  521 Kbits/sec
[ 3] 5.0- 6.0 sec  63.6 KBytes  521 Kbits/sec
[ 3] 6.0- 7.0 sec  127 KBytes  1.04 Mbits/sec
[ 3] 7.0- 8.0 sec  255 KBytes  2.09 Mbits/sec
[ 3] 8.0- 9.0 sec  63.6 KBytes  521 Kbits/sec
[ 3] 9.0-10.0 sec   0.00 Bytes  0.00 bits/sec
[ 3] 0.0-10.2 sec  1.28 MBytes  1.05 Mbits/sec
mininet>
```

```
affan@fanzru: ~/jarkom/fix_tubes
mininet> h1 ping h2
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0- 7.9 sec   1.00 MBytes  1.06 Mbits/sec
64 bytes from 192.168.2.2: icmp_seq=5 ttl=62 time=1240 ms
64 bytes from 192.168.2.2: icmp_seq=8 ttl=62 time=867 ms
64 bytes from 192.168.2.2: icmp_seq=9 ttl=62 time=837 ms
64 bytes from 192.168.2.2: icmp_seq=10 ttl=62 time=667 ms
64 bytes from 192.168.2.2: icmp_seq=11 ttl=62 time=0.118 ms
64 bytes from 192.168.2.2: icmp_seq=12 ttl=62 time=0.096 ms
64 bytes from 192.168.2.2: icmp_seq=13 ttl=62 time=0.113 ms
64 bytes from 192.168.2.2: icmp_seq=14 ttl=62 time=0.095 ms
64 bytes from 192.168.2.2: icmp_seq=15 ttl=62 time=0.095 ms
64 bytes from 192.168.2.2: icmp_seq=16 ttl=62 time=0.112 ms
64 bytes from 192.168.2.2: icmp_seq=17 ttl=62 time=0.099 ms
64 bytes from 192.168.2.2: icmp_seq=18 ttl=62 time=0.102 ms
^C
--- 192.168.2.2 ping statistics ---
18 packets transmitted, 12 received, 33,3333% packet loss, time 17316ms
rtt min/avg/max/mdev = 0.095/301.036/1240.233/442.403 ms, pipe 2
mininet>
```

- Buffer 60

```
affan@fanzru: ~/jarkom/fix_tubes

mininet> h1 ping h2
bind failed: Cannot assign requested address
-----
Client connecting to 192.168.2.2, TCP port 5001
Binding to local address 192.162.0.2
TCP window size: 85.3 KByte (default)
-----
[ 3] local 192.168.0.2 port 42474 connected with 192.168.2.2 port 5001
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=62 time=1180 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=62 time=1511 ms
64 bytes from 192.168.2.2: icmp_seq=3 ttl=62 time=1529 ms
64 bytes from 192.168.2.2: icmp_seq=4 ttl=62 time=1621 ms
64 bytes from 192.168.2.2: icmp_seq=5 ttl=62 time=1763 ms
64 bytes from 192.168.2.2: icmp_seq=6 ttl=62 time=2122 ms
64 bytes from 192.168.2.2: icmp_seq=7 ttl=62 time=2602 ms
^C
--- 192.168.2.2 ping statistics ---
9 packets transmitted, 7 received, 22,222% packet loss, time 8020ms
rtt min/avg/max/mdev = 1180.208/1761.193/2601.782/432.713 ms, pipe 3
mininet>
```

```
affan@fanzru: ~/jarkom/fix_tubes

mininet> h1 iperf -i 1 -c 192.168.2.2
[ ID] Interval      Transfer      Bandwidth
[ 3]  0.0-16.3 sec  1.75 MBytes   903 Kbits/sec
-----
Client connecting to 192.168.2.2, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 3] local 192.168.0.2 port 42478 connected with 192.168.2.2 port 5001
[ ID] Interval      Transfer      Bandwidth
[ 3]  0.0- 1.0 sec   96.2 KBytes    788 Kbits/sec
[ 3]  1.0- 2.0 sec    0.00 Bytes     0.00 bits/sec
[ 3]  2.0- 3.0 sec   84.8 KBytes    695 Kbits/sec
[ 3]  3.0- 4.0 sec    0.00 Bytes     0.00 bits/sec
[ 3]  4.0- 5.0 sec   11.3 KBytes    92.7 Kbits/sec
[ 3]  5.0- 6.0 sec    0.00 Bytes     0.00 bits/sec
[ 3]  6.0- 7.0 sec    2.83 KBytes    23.2 Kbits/sec
[ 3]  7.0- 8.0 sec    0.00 Bytes     0.00 bits/sec
[ 3]  8.0- 9.0 sec    0.00 Bytes     0.00 bits/sec
[ 3]  9.0-10.0 sec    0.00 Bytes     0.00 bits/sec
[ 3]  0.0-10.2 sec   195 KBytes    156 Kbits/sec
mininet>
```

- **Buffer 100**

```
affan@fanzru: ~/jarkom/fix_tubes
Binding to local address 192.162.0.2
TCP window size: 85.3 KByte (default)
-----
[ 3] local 192.168.0.2 port 42488 connected with 192.168.2.2 port 5001
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=62 time=1547 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=62 time=1533 ms
64 bytes from 192.168.2.2: icmp_seq=3 ttl=62 time=1627 ms
64 bytes from 192.168.2.2: icmp_seq=4 ttl=62 time=1865 ms
64 bytes from 192.168.2.2: icmp_seq=5 ttl=62 time=2198 ms
64 bytes from 192.168.2.2: icmp_seq=6 ttl=62 time=2678 ms
64 bytes from 192.168.2.2: icmp_seq=7 ttl=62 time=3201 ms
^C
--- 192.168.2.2 ping statistics ---
11 packets transmitted, 7 received, 36,3636% packet loss, time 10041ms
rtt min/avg/max/mdev = 1533.300/2092.796/3201.414/593.093 ms, pipe 4
mininet> h1 iperf -i 1 -c 192.168.2.2
```

```
affan@fanzru: ~/jarkom/fix_tubes
Client connecting to 192.168.2.2, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 3] local 192.168.0.2 port 42496 connected with 192.168.2.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0- 1.0 sec   93.3 KBytes 765 Kbits/sec
[ 3] 1.0- 2.0 sec   21.2 KBytes 174 Kbits/sec
[ 3] 2.0- 3.0 sec   63.6 KBytes 521 Kbits/sec
[ 3] 3.0- 4.0 sec    0.00 Bytes 0.00 bits/sec
[ 3] 4.0- 5.0 sec   123 KBytes 1.01 Mbits/sec
[ 3] 5.0- 6.0 sec   38.2 KBytes 313 Kbits/sec
[ 3] 6.0- 7.0 sec    0.00 Bytes 0.00 bits/sec
[ 3] 7.0- 8.0 sec   215 KBytes 1.76 Mbits/sec
[ 3] 8.0- 9.0 sec   127 KBytes 1.04 Mbits/sec
[ 3] 9.0-10.0 sec   255 KBytes 2.09 Mbits/sec
[ 3] 0.0-10.2 sec   936 KBytes 748 Kbits/sec
mininet> █
```

6. DOKUMENTASI TUGAS BESAR

Kode Program terbagi menjadi 3 Prosedur def clo1 untuk point clo 1, def clo23 untuk point clo 2 dan clo 3, def clo4 untuk clo 4, kodenya sebagai berikut :

```
from mininet.net import Mininet
```

```

from mininet.link import TCLink
from mininet.log import setLogLevel
from mininet.cli import CLI

def clo1():
    net = Mininet()
    #definisi host
    r1 = net.addHost("r1")
    r2 = net.addHost("r2")
    r3 = net.addHost("r3")
    r4 = net.addHost("r4")
    h1 = net.addHost("h1")
    h2 = net.addHost("h2")
    #penyambungan link atau subnetting
    net.addLink(h1,r1,intfName1='h1-eth0',intfName2='r1-eth0',cls = TCLink,bw=1)
    net.addLink(h1,r2,intfName1='h1-eth1',intfName2='r2-eth0',cls = TCLink,bw=1)
    net.addLink(h2,r3,intfName1='h2-eth0',intfName2='r3-eth0',cls = TCLink,bw=1)
    net.addLink(h2,r4,intfName1='h2-eth1',intfName2='r4-eth0',cls = TCLink,bw=1)
    net.addLink(r1,r3,intfName1='r1-eth1',intfName2='r3-eth1',cls = TCLink,bw=0.5)
    net.addLink(r1,r4,intfName1='r1-eth2',intfName2='r4-eth1',cls = TCLink,bw=1)
    net.addLink(r2,r3,intfName1='r2-eth1',intfName2='r3-eth2',cls = TCLink,bw=1)
    net.addLink(r2,r4,intfName1='r2-eth2',intfName2='r4-eth2',cls = TCLink,bw=0.5)
    net.start()

    # Konfigurasi IP Address di host 1 dan 2
    h1.cmd("ifconfig h1-eth0 192.168.0.2/24 netmask 255.255.255.0")
    h1.cmd("ifconfig h1-eth1 192.168.1.2/24 netmask 255.255.255.0")
    h2.cmd("ifconfig h2-eth0 192.168.2.2/24 netmask 255.255.255.0")
    h2.cmd("ifconfig h2-eth1 192.168.3.2/24 netmask 255.255.255.0")

    # konfigurasi Router
    r1.cmd("ifconfig r1-eth0 192.168.0.1/24 netmask 255.255.255.0")
    r1.cmd("ifconfig r1-eth1 192.168.4.1/24 netmask 255.255.255.0")
    r1.cmd("ifconfig r1-eth2 192.168.5.2/24 netmask 255.255.255.0")
    r1.cmd("sysctl net.ipv4.ip_forward=1")

    r2.cmd("ifconfig r2-eth0 192.168.1.1/24 netmask 255.255.255.0")
    r2.cmd("ifconfig r2-eth1 192.168.6.2/24 netmask 255.255.255.0")
    r2.cmd("ifconfig r2-eth2 192.168.7.1/24 netmask 255.255.255.0")
    r2.cmd("sysctl net.ipv4.ip_forward=1")

    r3.cmd("ifconfig r3-eth0 192.168.2.1/24 netmask 255.255.255.0")
    r3.cmd("ifconfig r3-eth1 192.168.4.2/24 netmask 255.255.255.0")
    r3.cmd("ifconfig r3-eth2 192.168.6.1/24 netmask 255.255.255.0")
    r3.cmd("sysctl net.ipv4.ip_forward=1")

    #routing

    r4.cmd("ifconfig r4-eth0 192.168.3.1/24 netmask 255.255.255.0")
    r4.cmd("ifconfig r4-eth1 192.168.5.1/24 netmask 255.255.255.0")
    r4.cmd("ifconfig r4-eth2 192.168.7.2/24 netmask 255.255.255.0")
    r4.cmd("sysctl net.ipv4.ip_forward=1")

    print("TES PING H1 - R1")
    h1.cmdPrint("ping -c 3 192.168.0.1")
    print("\n")
    print("TES PING R1 - H1")
    r1.cmdPrint("ping -c 3 192.168.0.2")
    print("\n")

```

```

print("TES PING H1 - R2")
h1.cmdPrint("ping -c 3 192.168.1.1")
print("\n")
print("TES PING R2 - H1")
r2.cmdPrint("ping -c 3 192.168.1.2")
print("\n")

print("TES PING H2 - R3")
h2.cmdPrint("ping -c 3 192.168.2.1")
print("\n")
print("TES PING R3 - H2")
r3.cmdPrint("ping -c 3 192.168.2.2")
print("\n")

print("TES PING H2 - R4")
h2.cmdPrint("ping -c 3 192.168.3.1")
print("\n")
print("TES PING R4 - H2")
r4.cmdPrint("ping -c 3 192.168.3.2")
print("\n")

print("TES PING R1 - R3")
r1.cmdPrint("ping -c 3 192.168.5.1")
print("\n")
print("TES PING R3 - R1")
r3.cmdPrint("ping -c 3 192.168.5.2")
print("\n")

print("TES PING R1 - R4")
r1.cmdPrint("ping -c 3 192.168.4.2")
print("\n")
print("TES PING R4 - R1")
r4.cmdPrint("ping -c 3 192.168.4.1")
print("\n")

print("TES PING R2 - R3")
r2.cmdPrint("ping -c 3 192.168.6.1")
print("\n")
print("TES PING R3 - R2")
r3.cmdPrint("ping -c 3 192.168.6.2")
print("\n")

print("TES PING R2 - R4")
r2.cmdPrint("ping -c 3 192.168.7.1")
print("\n")
print("TES PING R4 - R2")
r4.cmdPrint("ping -c 3 192.168.7.2")
print("\n")

CLI(net)
net.stop()

def clo2_3():

    net = Mininet()
    #definisi Host
    r1 = net.addHost("r1")

```

```

r2 = net.addHost("r2")
r3 = net.addHost("r3")
r4 = net.addHost("r4")
h1 = net.addHost("h1")
h2 = net.addHost("h2")
#penyambungan link atau subnetting
net.addLink(h1,r1,intfName1='h1-eth0',intfName2='r1-eth0',cls = TCLink,bw=1)
net.addLink(h1,r2,intfName1='h1-eth1',intfName2='r2-eth0',cls = TCLink,bw=1)
net.addLink(h2,r3,intfName1='h2-eth0',intfName2='r3-eth0',cls = TCLink,bw=1)
net.addLink(h2,r4,intfName1='h2-eth1',intfName2='r4-eth0',cls = TCLink,bw=1)
net.addLink(r1,r3,intfName1='r1-eth1',intfName2='r3-eth1',cls = TCLink,bw=0.5)
net.addLink(r1,r4,intfName1='r1-eth2',intfName2='r4-eth1',cls = TCLink,bw=1)
net.addLink(r2,r3,intfName1='r2-eth1',intfName2='r3-eth2',cls = TCLink,bw=1)
net.addLink(r2,r4,intfName1='r2-eth2',intfName2='r4-eth2',cls = TCLink,bw=0.5)
net.start()

# Konfigurasi IP Address di host 1 dan 2
h1.cmd("ifconfig h1-eth0 192.168.0.2/24 netmask 255.255.255.0")
h1.cmd("ifconfig h1-eth1 192.168.1.2/24 netmask 255.255.255.0")
h2.cmd("ifconfig h2-eth0 192.168.2.2/24 netmask 255.255.255.0")
h2.cmd("ifconfig h2-eth1 192.168.3.2/24 netmask 255.255.255.0")

#routing h1 dan h2

h1.cmd("ip rule add from 192.168.0.2 table 1")
h1.cmd("ip rule add from 192.168.1.2 table 2")
h1.cmd("ip route add 192.168.0.0/24 dev h1-eth0 link table 1")
h1.cmd("ip route add default via 192.168.0.1 dev h1-eth0 table 1")
h1.cmd("ip route add 192.168.1.0/24 dev h1-eth0 link table 2")
h1.cmd("ip route add default via 192.168.1.1 dev h1-eth1 table 2")
h1.cmd("ip route add default scope global nexthop via 192.168.0.1 dev h1-eth0")
h1.cmd("ip route add default scope global nexthop via 192.168.1.1 dev h1-eth1")
#h1.cmd("route add default gw 192.168.0.1 h1-eth0")
#h1.cmd("route add default gw 192.168.1.1 h1-eth1")

h2.cmd("ip rule add from 192.168.2.2 table 3")
h2.cmd("ip rule add from 192.168.3.2 table 4")
h2.cmd("ip route add 192.168.2.0/24 dev h2-eth0 link table 3")
h2.cmd("ip route add default via 192.168.2.1 dev h2-eth0 table 3")
h2.cmd("ip route add 192.168.3.0/24 dev h2-eth0 link table 4")
h2.cmd("ip route add default via 192.168.3.1 dev h2-eth1 table 4")
h2.cmd("ip route add default scope global nexthop via 192.168.2.1 dev h2-eth0")
h2.cmd("ip route add default scope global nexthop via 192.168.3.1 dev h2-eth1")
#h2.cmd("route add default gw 192.168.2.1 h2-eth0")
#h2.cmd("route add default gw 192.168.3.1 h2-eth1")

#Konfigurasi Router dan Routing Static
r1.cmd("ifconfig r1-eth0 192.168.0.1/24 netmask 255.255.255.0")
r1.cmd("ifconfig r1-eth1 192.168.4.1/24 netmask 255.255.255.0")
r1.cmd("ifconfig r1-eth2 192.168.5.2/24 netmask 255.255.255.0")
r1.cmd("sysctl net.ipv4.ip_forward=1")

r1.cmd("route add -net 192.168.1.0/24 gw 192.168.5.1")
r1.cmd("route add -net 192.168.1.0/24 gw 192.168.4.2")
r1.cmd("route add -net 192.168.2.0/24 gw 192.168.4.2")
r1.cmd("route add -net 192.168.3.0/24 gw 192.168.5.1")
r1.cmd("route add -net 192.168.6.0/24 gw 192.168.4.2")
r1.cmd("route add -net 192.168.7.0/24 gw 192.168.5.1")

```



```

r2.cmd("ifconfig r2-eth0 192.168.1.1/24 netmask 255.255.255.0")
r2.cmd("ifconfig r2-eth1 192.168.6.2/24 netmask 255.255.255.0")
r2.cmd("ifconfig r2-eth2 192.168.7.1/24 netmask 255.255.255.0")
r2.cmd("sysctl net.ipv4.ip_forward=1")

r2.cmd("route add -net 192.168.0.0/24 gw 192.168.6.1")
r2.cmd("route add -net 192.168.0.0/24 gw 192.168.7.2")
r2.cmd("route add -net 192.168.2.0/24 gw 192.168.6.1")
r2.cmd("route add -net 192.168.3.0/24 gw 192.168.7.2")
r2.cmd("route add -net 192.168.4.0/24 gw 192.168.6.1")
r2.cmd("route add -net 192.168.5.0/24 gw 192.168.7.2")

r3.cmd("ifconfig r3-eth0 192.168.2.1/24 netmask 255.255.255.0")
r3.cmd("ifconfig r3-eth1 192.168.4.2/24 netmask 255.255.255.0")
r3.cmd("ifconfig r3-eth2 192.168.6.1/24 netmask 255.255.255.0")
r3.cmd("sysctl net.ipv4.ip_forward=1")

r3.cmd("route add -net 192.168.0.0/24 gw 192.168.4.1")
r3.cmd("route add -net 192.168.1.0/24 gw 192.168.6.2")
r3.cmd("route add -net 192.168.3.0/24 gw 192.168.6.2")
r3.cmd("route add -net 192.168.3.0/24 gw 192.168.4.1")
r3.cmd("route add -net 192.168.7.0/24 gw 192.168.6.2")
r3.cmd("route add -net 192.168.5.0/24 gw 192.168.4.1")

r4.cmd("ifconfig r4-eth0 192.168.3.1/24 netmask 255.255.255.0")
r4.cmd("ifconfig r4-eth1 192.168.5.1/24 netmask 255.255.255.0")
r4.cmd("ifconfig r4-eth2 192.168.7.2/24 netmask 255.255.255.0")
r4.cmd("sysctl net.ipv4.ip_forward=1")

r4.cmd("route add -net 192.168.0.0/24 gw 192.168.5.2")
r4.cmd("route add -net 192.168.1.0/24 gw 192.168.7.1")
r4.cmd("route add -net 192.168.2.0/24 gw 192.168.7.1")
r4.cmd("route add -net 192.168.2.0/24 gw 192.168.5.2")
r4.cmd("route add -net 192.168.6.0/24 gw 192.168.7.1")
r4.cmd("route add -net 192.168.4.0/24 gw 192.168.5.2")

print("TES PING R1 - R2")
r1.cmdPrint("ping -c 3 192.168.6.2")
print("\n")
print("TES PING R3 - R4")
r3.cmdPrint("ping -c 3 192.168.4.2")
print("\n")

print("TES PING H1 - H2 interface Atas")
h1.cmdPrint("ping -c 3 192.168.2.2")
print("\n")
print("TES PING H1- H2 interface Bawah")
h1.cmdPrint("ping -c 3 192.168.3.2")
print("\n")
print("TES PING H2 - H1 interface Atas")
h2.cmdPrint("ping -c 3 192.168.0.2")
print("\n")
print("TES PING H2- H1 interface Bawah")
h2.cmdPrint("ping -c 3 192.168.1.2")
print("\n")

```

```

CLI(net)
net.stop()
def clo4(buffer):

    net = Mininet()
    #definisi host
    r1 = net.addHost("r1")
    r2 = net.addHost("r2")
    r3 = net.addHost("r3")
    r4 = net.addHost("r4")
    h1 = net.addHost("h1")
    h2 = net.addHost("h2")

    buffer= int(buffer)
    #penyambungan link dan set buffer dan menggunakan queue HTB dengan static routing

net.addLink(h1,r1,max_queue_size=buffer,use_htb=True,intfName1='h1-eth0',intfName2='r1-eth0',cls = TCLink,bw=1)

net.addLink(h1,r2,max_queue_size=buffer,use_htb=True,intfName1='h1-eth1',intfName2='r2-eth0',cls = TCLink,bw=1)

net.addLink(h2,r3,max_queue_size=buffer,use_htb=True,intfName1='h2-eth0',intfName2='r3-eth0',cls = TCLink,bw=1)

net.addLink(h2,r4,max_queue_size=buffer,use_htb=True,intfName1='h2-eth1',intfName2='r4-eth0',cls = TCLink,bw=1)

net.addLink(r1,r3,max_queue_size=buffer,use_htb=True,intfName1='r1-eth1',intfName2='r3-eth1',cls = TCLink,bw=0.5)

net.addLink(r1,r4,max_queue_size=buffer,use_htb=True,intfName1='r1-eth2',intfName2='r4-eth1',cls = TCLink,bw=1)

net.addLink(r2,r3,max_queue_size=buffer,use_htb=True,intfName1='r2-eth1',intfName2='r3-eth2',cls = TCLink,bw=1)

net.addLink(r2,r4,max_queue_size=buffer,use_htb=True,intfName1='r2-eth2',intfName2='r4-eth2',cls = TCLink,bw=0.5)
    net.start()


    # Konfigurasi IP Address di host 1 dan 2
    h1.cmd("ifconfig h1-eth0 192.168.0.2/24 netmask 255.255.255.0")
    h1.cmd("ifconfig h1-eth1 192.168.1.2/24 netmask 255.255.255.0")
    h2.cmd("ifconfig h2-eth0 192.168.2.2/24 netmask 255.255.255.0")
    h2.cmd("ifconfig h2-eth1 192.168.3.2/24 netmask 255.255.255.0")


    #routing h1 h2

    h1.cmd("ip rule add from 192.168.0.2 table 1")
    h1.cmd("ip rule add from 192.168.1.2 table 2")
    h1.cmd("ip route add 192.168.0.0/24 dev h1-eth0 link table 1")
    h1.cmd("ip route add default via 192.168.0.1 dev h1-eth0 table 1")
    h1.cmd("ip route add 192.168.1.0/24 dev h1-eth0 link table 2")
    h1.cmd("ip route add default via 192.168.1.1 dev h1-eth1 table 2")
    h1.cmd("ip route add default scope global nexthop via 192.168.0.1 dev h1-eth0")

```

```
h1.cmd("ip route add default scope global nexthop via 192.168.1.1 dev h1-eth1")
#h1.cmd("route add -net default gw 192.168.0.1")
#h1.cmd("route add -net default gw 192.168.1.1")
```

```
h2.cmd("ip rule add from 192.168.2.2 table 3")
h2.cmd("ip rule add from 192.168.3.2 table 4")
h2.cmd("ip route add 192.168.2.0/24 dev h2-eth0 link table 3")
h2.cmd("ip route add default via 192.168.2.1 dev h2-eth0 table 3")
h2.cmd("ip route add 192.168.3.0/24 dev h2-eth0 link table 4")
h2.cmd("ip route add default via 192.168.3.1 dev h2-eth1 table 4")
h2.cmd("ip route add default scope global nexthop via 192.168.2.1 dev h2-eth0")
h2.cmd("ip route add default scope global nexthop via 192.168.3.1 dev h2-eth1")
#h2.cmd("route add default gw 192.168.2.1 h2-eth0")
#h2.cmd("route add default gw 192.168.3.1 h2-eth1")
```

#Konfigurasi Router dan Routing

```
r1.cmd("ifconfig r1-eth0 192.168.0.1/24 netmask 255.255.255.0")
r1.cmd("ifconfig r1-eth1 192.168.4.1/24 netmask 255.255.255.0")
r1.cmd("ifconfig r1-eth2 192.168.5.2/24 netmask 255.255.255.0")
r1.cmd("sysctl net.ipv4.ip_forward=1")
```

```
r1.cmd("route add -net 192.168.1.0/24 gw 192.168.5.1")
r1.cmd("route add -net 192.168.1.0/24 gw 192.168.4.2")
r1.cmd("route add -net 192.168.2.0/24 gw 192.168.4.2")
r1.cmd("route add -net 192.168.3.0/24 gw 192.168.5.1")
r1.cmd("route add -net 192.168.6.0/24 gw 192.168.4.2")
r1.cmd("route add -net 192.168.7.0/24 gw 192.168.5.1")
```

```
r2.cmd("ifconfig r2-eth0 192.168.1.1/24 netmask 255.255.255.0")
r2.cmd("ifconfig r2-eth1 192.168.6.2/24 netmask 255.255.255.0")
r2.cmd("ifconfig r2-eth2 192.168.7.1/24 netmask 255.255.255.0")
r2.cmd("sysctl net.ipv4.ip_forward=1")
```

```
r2.cmd("route add -net 192.168.0.0/24 gw 192.168.6.1")
r2.cmd("route add -net 192.168.0.0/24 gw 192.168.7.2")
r2.cmd("route add -net 192.168.2.0/24 gw 192.168.6.1")
r2.cmd("route add -net 192.168.3.0/24 gw 192.168.7.2")
r2.cmd("route add -net 192.168.4.0/24 gw 192.168.6.1")
r2.cmd("route add -net 192.168.5.0/24 gw 192.168.7.2")
```

```
r3.cmd("ifconfig r3-eth0 192.168.2.1/24 netmask 255.255.255.0")
r3.cmd("ifconfig r3-eth1 192.168.4.2/24 netmask 255.255.255.0")
r3.cmd("ifconfig r3-eth2 192.168.6.1/24 netmask 255.255.255.0")
r3.cmd("sysctl net.ipv4.ip_forward=1")
```

```
r3.cmd("route add -net 192.168.0.0/24 gw 192.168.4.1")
r3.cmd("route add -net 192.168.1.0/24 gw 192.168.6.2")
r3.cmd("route add -net 192.168.3.0/24 gw 192.168.6.2")
r3.cmd("route add -net 192.168.3.0/24 gw 192.168.4.1")
r3.cmd("route add -net 192.168.7.0/24 gw 192.168.6.2")
r3.cmd("route add -net 192.168.5.0/24 gw 192.168.4.1")
```

```
r4.cmd("ifconfig r4-eth0 192.168.3.1/24 netmask 255.255.255.0")
r4.cmd("ifconfig r4-eth1 192.168.5.1/24 netmask 255.255.255.0")
r4.cmd("ifconfig r4-eth2 192.168.7.2/24 netmask 255.255.255.0")
r4.cmd("sysctl net.ipv4.ip_forward=1")
```

```

r4.cmd("route add -net 192.168.0.0/24 gw 192.168.5.2")
r4.cmd("route add -net 192.168.1.0/24 gw 192.168.7.1")
r4.cmd("route add -net 192.168.2.0/24 gw 192.168.7.1")
r4.cmd("route add -net 192.168.2.0/24 gw 192.168.5.2")
r4.cmd("route add -net 192.168.6.0/24 gw 192.168.7.1")
r4.cmd("route add -net 192.168.4.0/24 gw 192.168.5.2")

#membuat trafic iperf server client dalam background
h2.cmdPrint("iperf -s &")
h1.cmdPrint("iperf -t 60 -B 192.162.0.2 -c 192.168.2.2 &")
#h1.cmdPrint("iperf -t 60 -B 192.162.1.2 -c 192.168.2.2 &")

CLI(net)
net.stop()

a = 0
while(a != '1' and a != '2' and a != '3' ):
    print('Selamat Datang Di Tubes Jarkom')
    print('1. Masuk ke CLO 1')
    print('2. Masuk ke CLO 2 dan 3')
    print('3. Masuk ke CLO 4')
    a = input('Pilih : ')

if(a=='1'):
    setLogLevel('info')
    clo1()
elif(a=='2'):
    setLogLevel('info')
    clo2_3()
elif(a=='3'):
    buffer = -1
    while( buffer != '20' and buffer != '40' and buffer != '60' and buffer != '100'):
        buffer = input('Masukan Nilai Buffer (20/40/60/100) : ')
    setLogLevel('info')
    clo4(buffer)

```

7. REFERENSI

- [1] Kontributor dari proyek Wikimedia (2012). Subnetwork. [online] Wikipedia.org. Available at: <https://id.wikipedia.org/wiki/Subnetwork> [Accessed 13 Jun. 2021].
- [2] <https://www.facebook.com/PROFIL-FACEBOOK> (2016). Pengertian Routing, Static Routing, dan Dynamic Routing. [online] Diary Config. Available at: <https://www.diaryconfig.com/2016/01/pengertian-routing.html> [Accessed 13 Jun. 2021].
- [3] Kontributor dari proyek Wikimedia (2012). Subnetwork. [online] Wikipedia.org. Available at: <https://id.wikipedia.org/wiki/Subnetwork> [Accessed 13 Jun. 2021].
- [4] Kontributor dari proyek Wikimedia (2004). Protokol Kendali Transmisi. [online] Wikipedia.org. Available at: https://id.wikipedia.org/wiki/Protokol_Kendali_Transmisi [Accessed 13 Jun. 2021].
- [5] Netmonk.id. (2019). ✓ Tips Monitoring Lalu Lintas Jaringan (Network Traffic). [online] Available at: <https://netmonk.id/tips-monitoring-network-traffic> [Accessed 13 Jun. 2021].
- [6] Kaunaini, Nuril, 2021. “Tubes Jarkom 2021” [Video]. Youtube. <https://www.youtube.com/playlist?app=desktop&list=PLt8tRre7bmMADQbZINxeXYtYUmVK7eYc4> diakses 11 juni 2021 Davisko, 2011.