SIMULASI JARINGAN DENGAN MININET

Disusun Untuk Memenuhi Tugas Besar Jaringan Komputer

Logo, company name

Description automatically generated

Disusun oleh :

**Hilman Taris Muttaqin (1301204208 – IF4408)**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY**

**2021/2022**

# **Daftar Isi**

[**Daftar Isi**](#_heading=h.pi5qaqxbh4e0) **2**

[**CLO 1**](#_heading=h.xju0mlvxev4) **3**

[Desain subnet masing-masing network](#_heading=h.4o111zaqe4fs) 3

[Assign IP sesuai subnet](#_heading=h.f0nb64b0dne7) 4

[Uji konektivitas dengan ping antara 2 host yang berada dalam 1 network](#_heading=h.5sfjro66sxug) 11

[**CLO 2**](#_heading=h.cvcleb2a544i) **16**

[Membuat tabel routing di semua host, dibuktikan dengan ping antar host](#_heading=h.h7whc8hhkcks) 16

[Menganalisis routing yang digunakan menggunakan traceroute](#_heading=h.kbzuf5x24vxf) 19

[**CLO 3**](#_heading=h.qb36hjnqwnzq) **26**

[Generate traffic menggunakan iPerf](#_heading=h.xvjd95hp0zem) 26

[Capture traffic menggunakan custom script atau wireshark untuk diinspeksi, dibuktikan dengan traffic di wireshark/tcpdump](#_heading=h.8kge6s89sta4) 28

[**CLO 4**](#_heading=h.n8rhpp7s3ep9) **29**

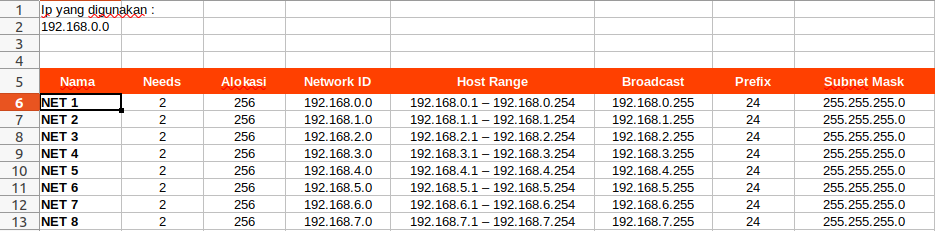
[Set ukuran buffer pada router : 20, 40, 60 dan 100 dan Capture pengaruh ukuran buffer terhadap delay](#_heading=h.k4nc4o6ofmej) 29

# **CLO 1**

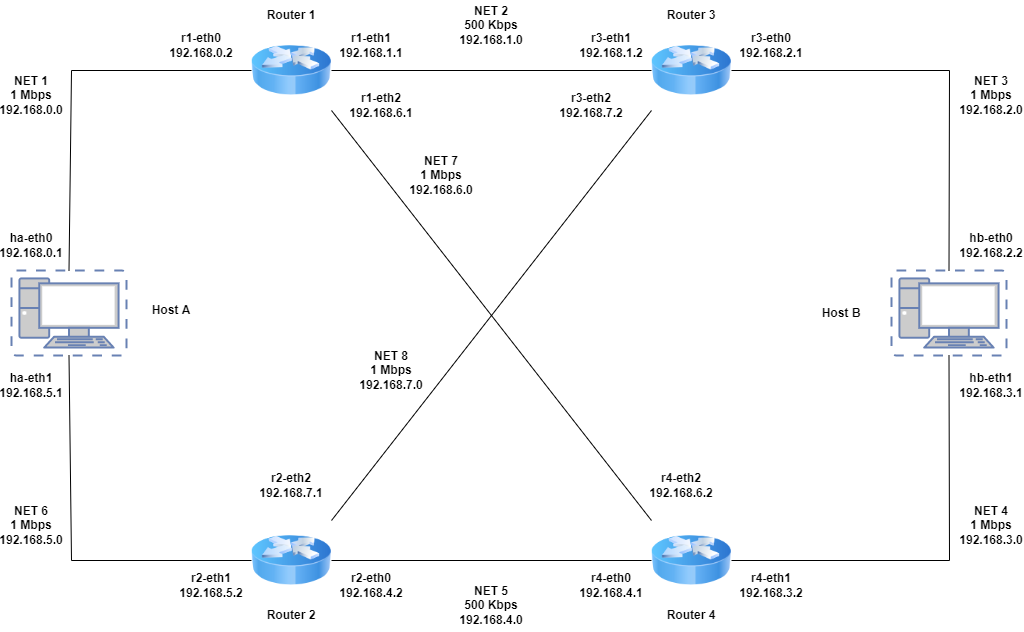
Link github project:  
<https://github.com/Hilmantm/tubes-mininet-jarkom>

## **Desain subnet masing-masing network**

IP yang digunakan adalah 192.168.0.0. Dibawah ini adalah desain subnet yang akan digunakan pada topologi:

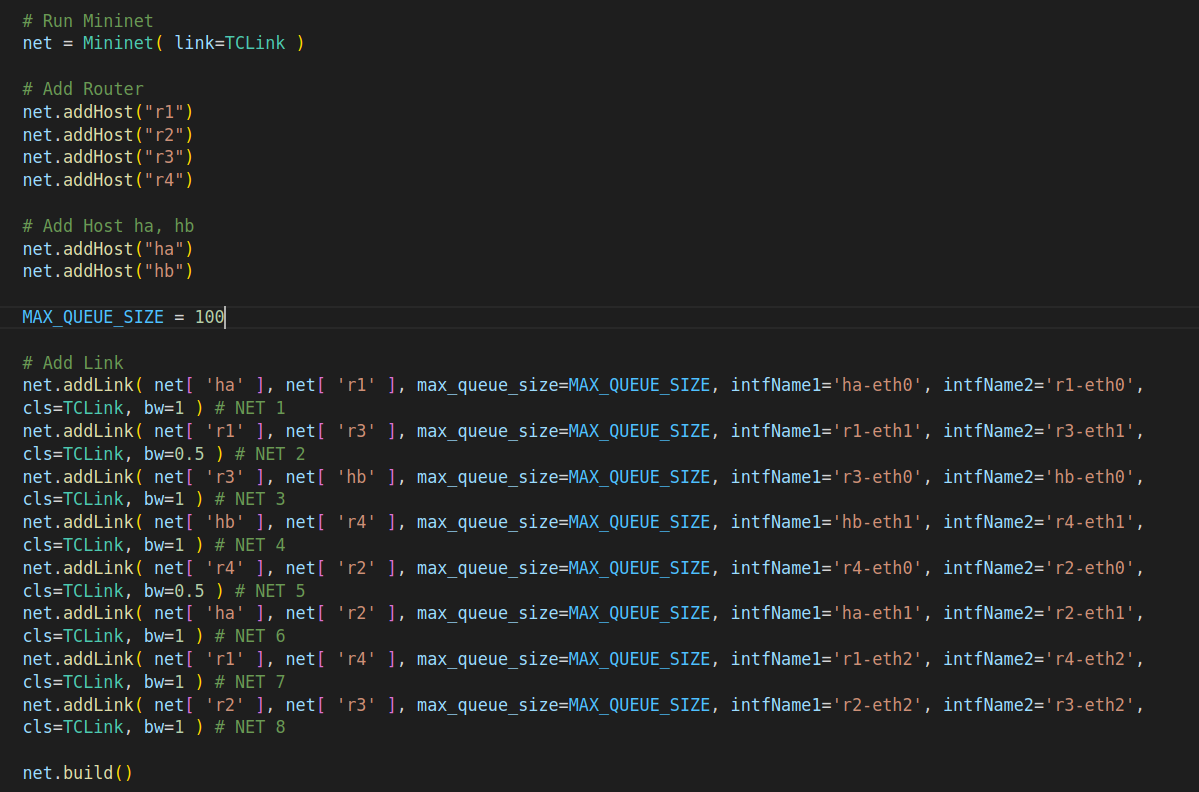


Apabila diimplementasikan pada topologi yang sudah diberikan, maka akan menghasilkan diagram seperti dibawah ini.

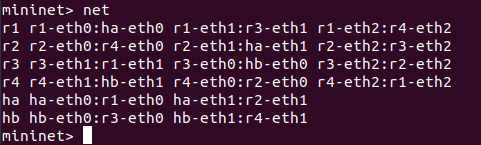


## **Assign IP sesuai subnet**

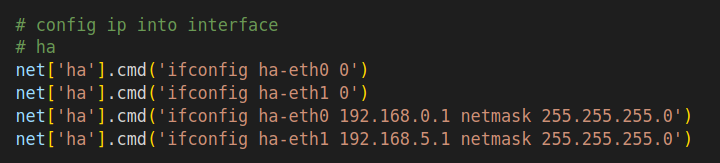
Untuk memulai membangun topologi jaringan tersebut, maka hal pertama yang harus dilakukan adalah menginisialisasi komponen yang ada pada topologi tersebut. Di tugas ini, komponen yang dibutuhkan adalah host (PC) dan router. Serta masing-masing host harus terhubung satu sama lain.

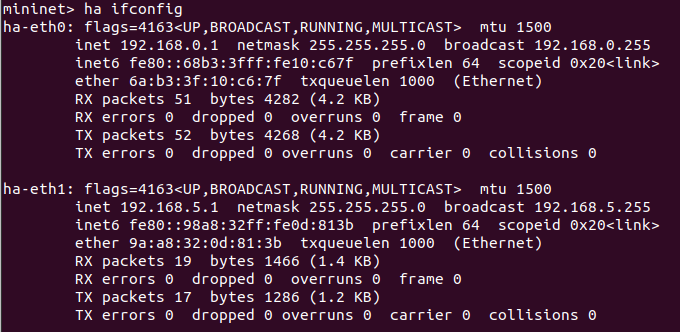


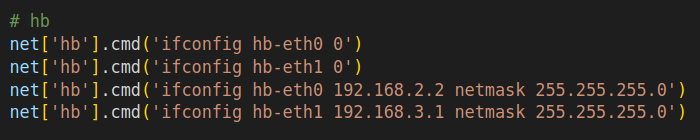
Untuk memastikan apakah interface yang dikoneksikan sudah sesuai dengan topologi, dapat dicek menggunakan command:  
**mininet > net**

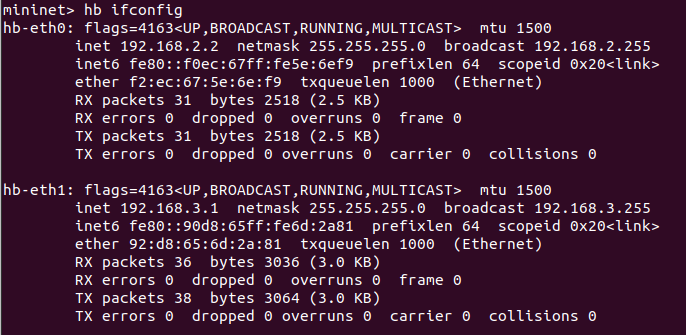


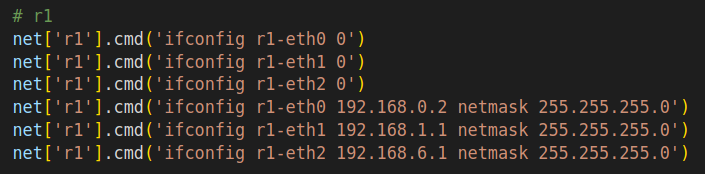
Setelah diinisialisasi dan dikoneksikan, maka masing-masing host harus dilakukan penyetingan IP sesuai dengan subnet yang sudah dibuat.

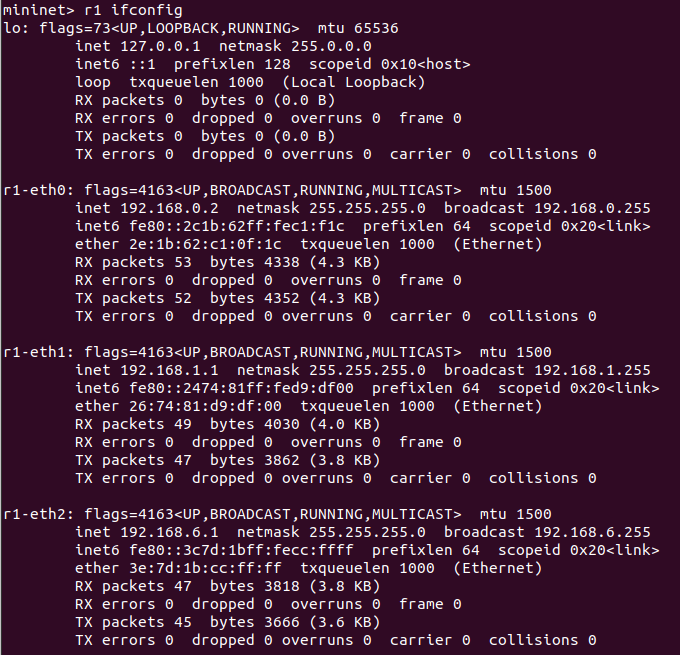
Setting IP: **HOST A  
**

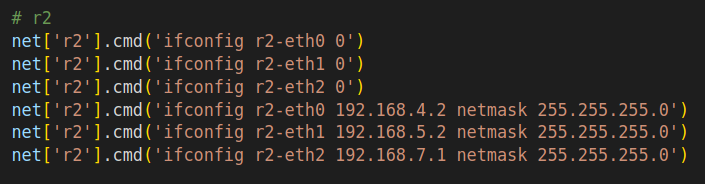
Pastikan kembali bahwa config IP pada interface di **host A** sudah sesuai  


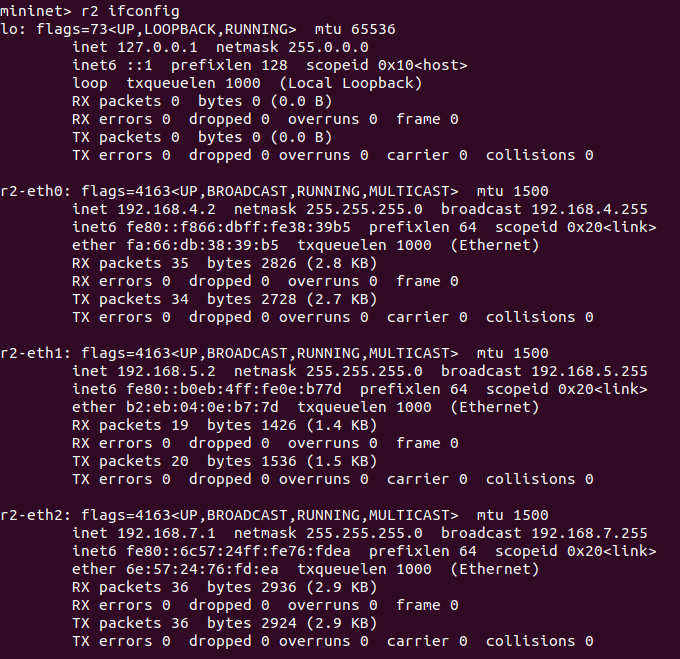
Setting IP: **HOST B**

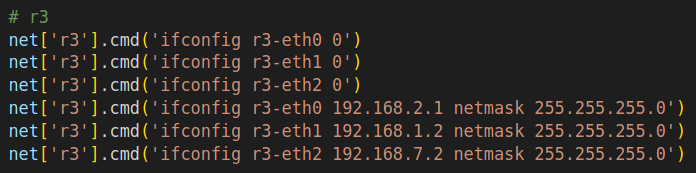
Pastikan kembali bahwa config IP pada interface **host B** sudah sesuai  


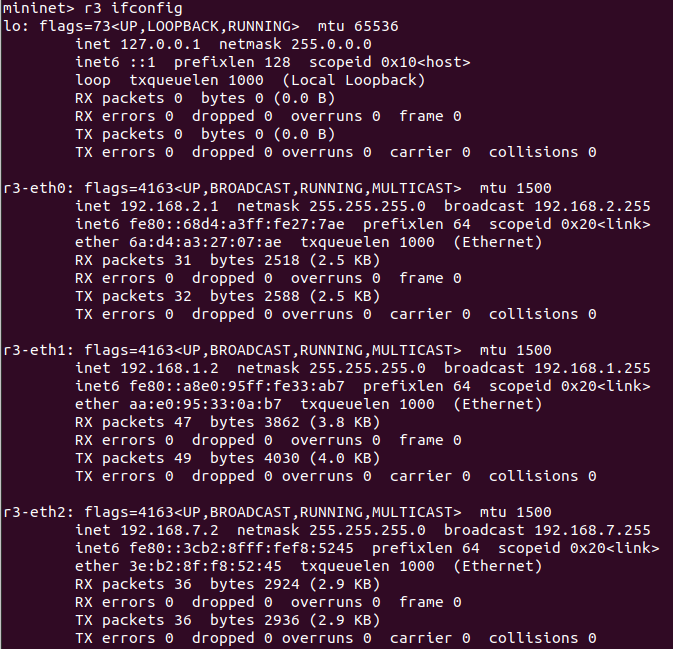
Setting IP: **Router 1**  


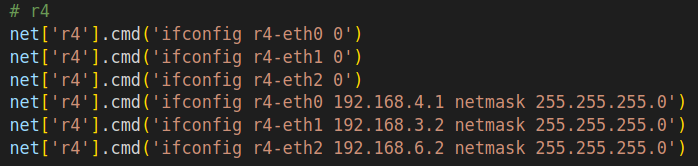
Pastikan kembali bahwa config IP pada interface **router 1** sudah sesuai  


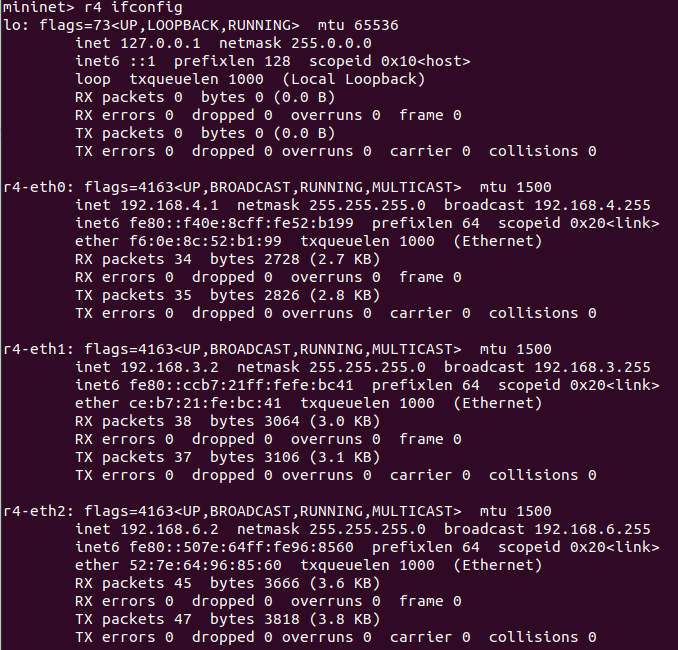
Setting IP: **Router 2**  


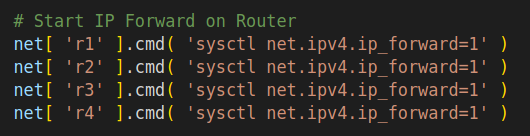
Pastikan kembali bahwa config IP pada interface **router 2** sudah sesuai  


Setting IP: **Router 3**  


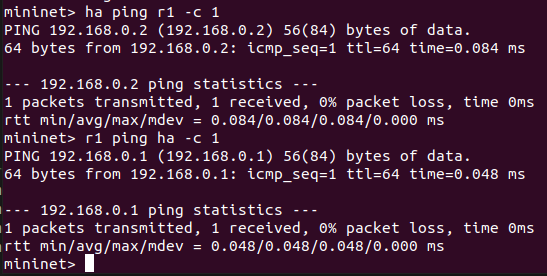
Pastikan kembali bahwa config IP pada interface **router 3** sudah sesuai  


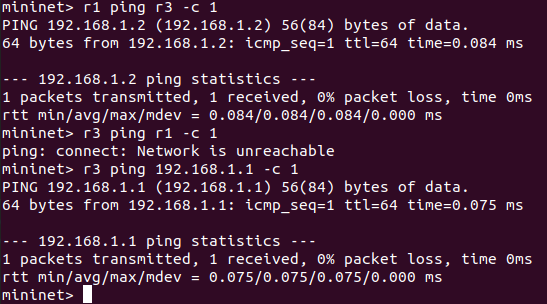
Setting IP: **Router 4**  


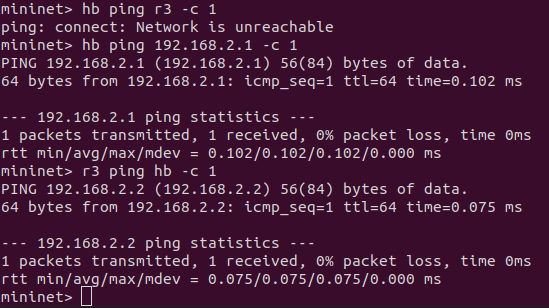
Pastikan kembali bahwa config IP pada interface **router 4** sudah sesuai  


Setelah IP di setting, maka router yang terhubung harus diaktifkan fitur ip\_forward  


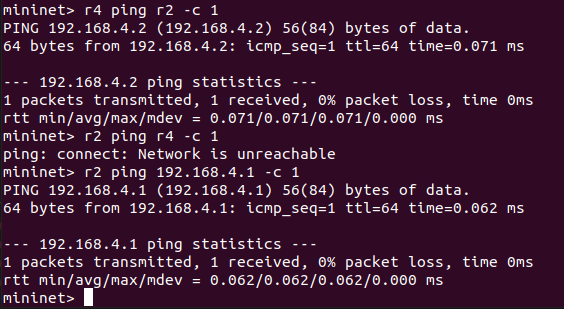
## **Uji konektivitas dengan ping antara 2 host yang berada dalam 1 network**

Ping Network 1:  


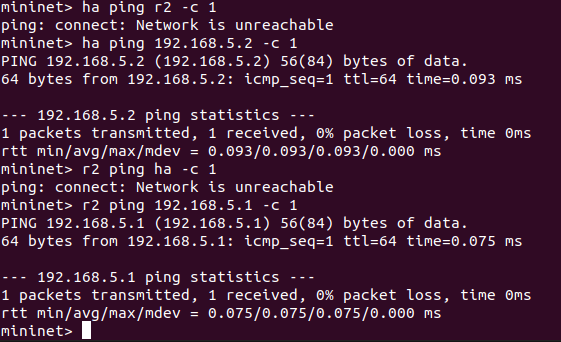
Ping Network 2:  


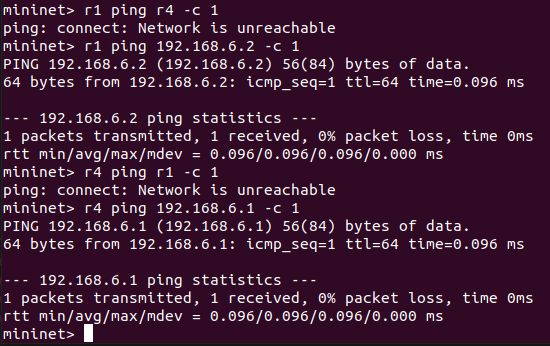
Ping Network 3:  


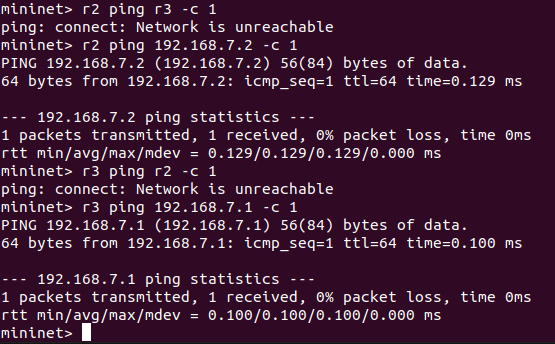
Ping Network 4:  


Ping Network 5:  


Ping Network 6:



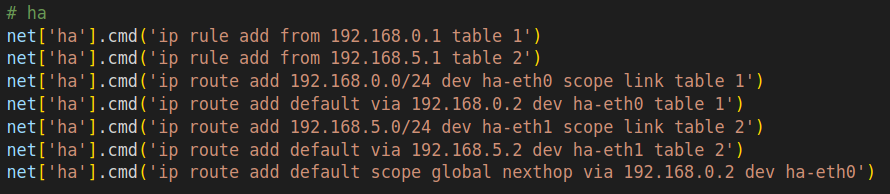
Ping Network 7:  


Ping Network 8:  


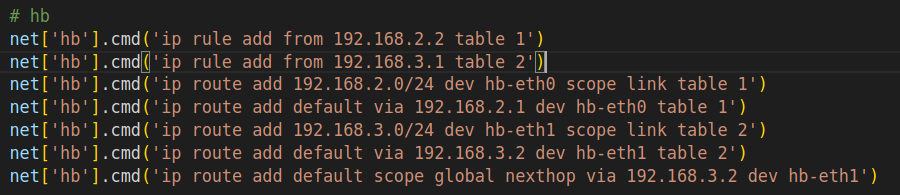
# **CLO 2**

## **Membuat tabel routing di semua host, dibuktikan dengan ping antar host**

Setelah berhasil melakukan ping antar host di masing-masing network, maka langkah selanjutnya adalah membuat routing. Untuk host, maka dibutuhkan tabel routing untuk mengetahui informasi jalur mana saja yang dapat digunakan untuk pengiriman data pada jaringan.

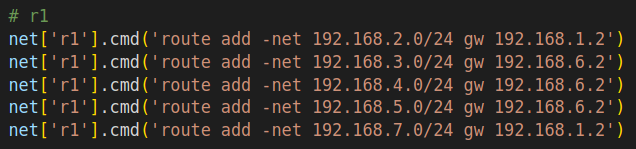
Tabel routing: **Host A**  


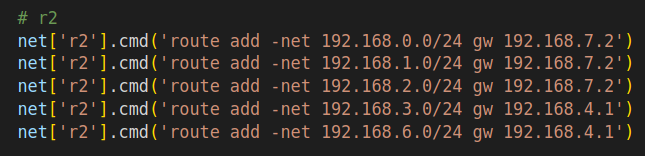
Pada host A, jalur yang bisa digunakan untuk pengiriman data adalah jalur pada **NET 1 (192.168.0.1 - table 1)** dan **NET 6 (192.168.5.1 - table 2)**. Untuk host A, default scope global nexthop nya melalui **NET 1**.

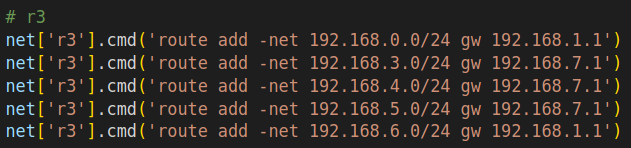
Tabel routing: **Host B**  


Pada host B, jalur yang bisa digunakan untuk pengiriman data adalah jalur pada **NET 3 (192.168.2.2 - table 1)** dan **NET 4 (192.168.3.1 - table 2)**. Untuk host A, default scope global nexthop nya melalui **NET 4**.

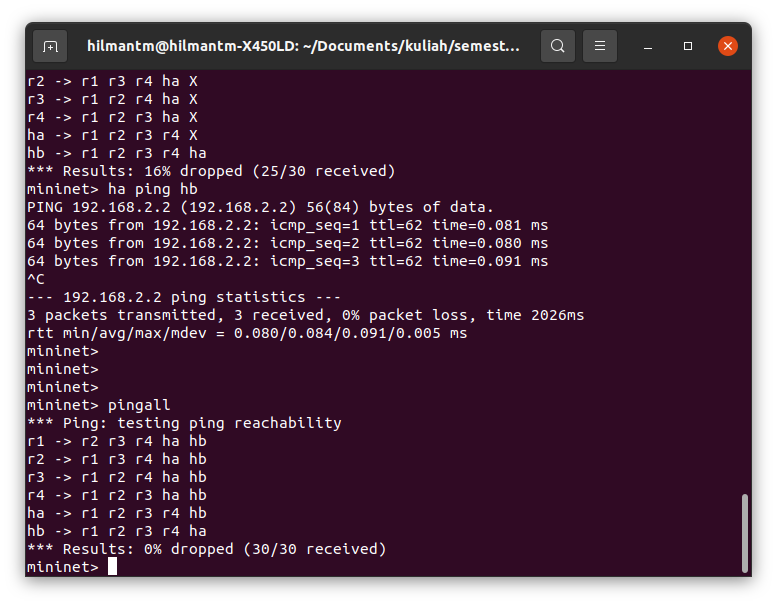
Selain tabel routing pada host, perlu dilakukan gateway pada router yang ada.

Gateway: **Router 1**  


Gateway: **Router 2**  


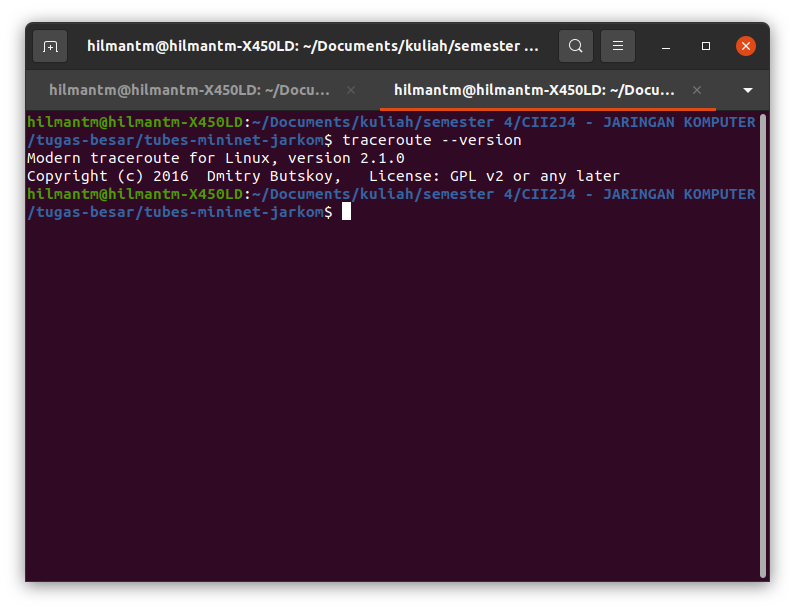
Gateway: **Router 3**  


Gateway: **Router 4**  

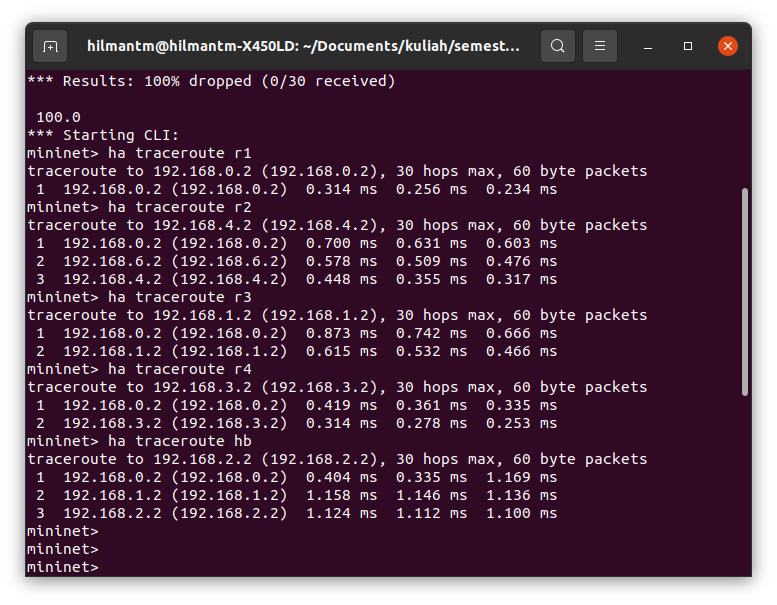

Untuk membuktikan bahwa semua sudah terkoneksi, pastikan ping satu persatu hingga ketika pingall tidak ada lagi “X” atau host yang tidak terkoneksi  


## 

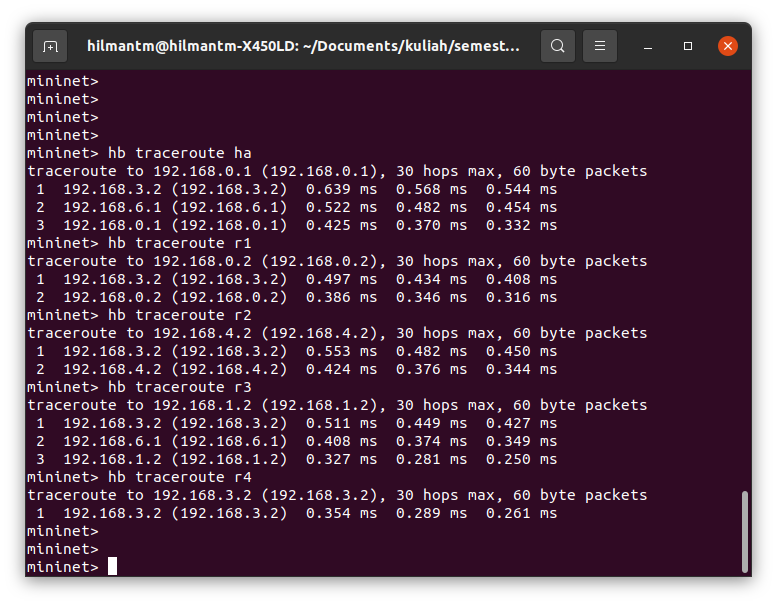
## **Menganalisis routing yang digunakan menggunakan traceroute**

Traceroute merupakan library yang harus di install pada linux. Untuk cek apakah traceroute sudah terinstall atau belum, ketikkan command:  
**traceroute --version  
**

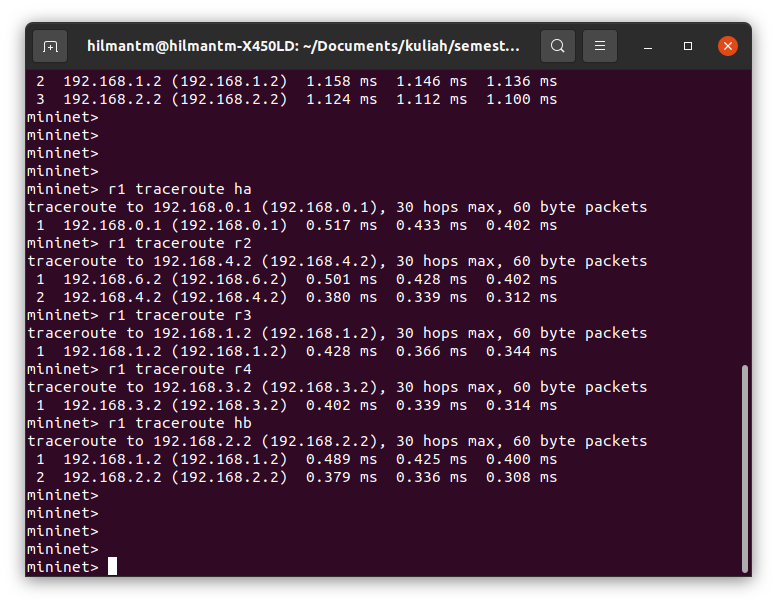
Apabila belum terinstall pada linux anda, maka install menggunakan command:  
**sudo apt-get update  
sudo apt-get install traceroute**

Traceroute: **Host A**  


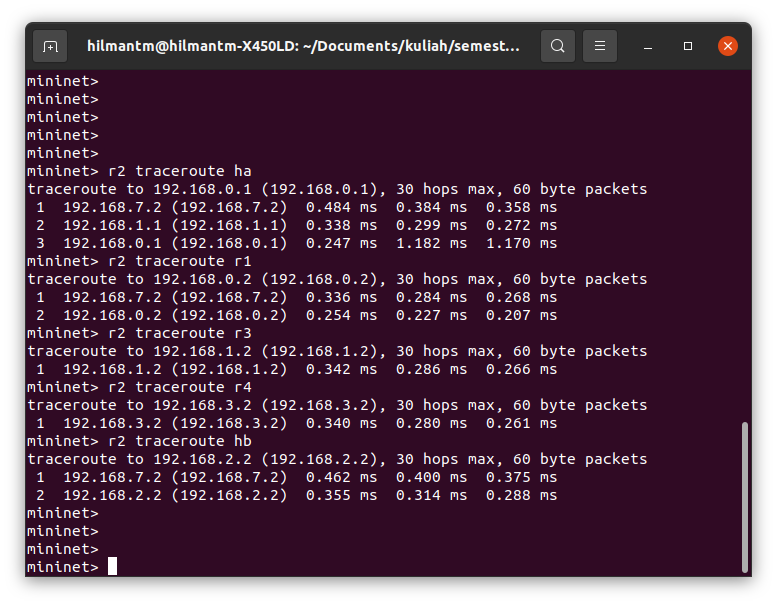
* **Route untuk mencapai ha → r2**  
  ha → r1 → r4 → r2
* **Route untuk mencapai ha → r3**  
  ha → r1 → r3
* **Route untuk mencapai ha → r4**  
  ha → r1 → r4
* **Route untuk mencapai ha → hb**  
  ha → r1 → r3 → hb

Traceroute: **Host B**  


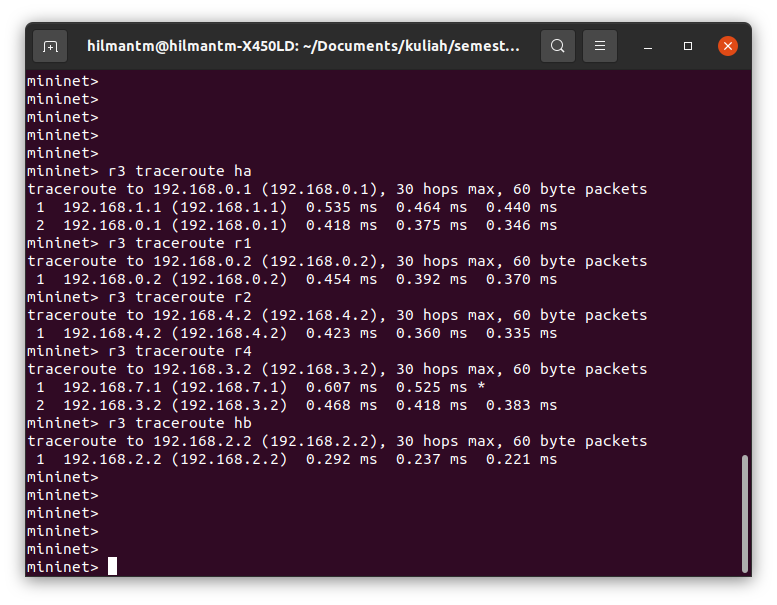
* **Route untuk mencapai hb → ha**hb → r4 → r1 → ha
* **Route untuk mencapai hb → r1**hb → r4 → r1
* **Route untuk mencapai hb → r2**hb → r4 → r2
* **Route untuk mencapai hb → r3**hb → r4 → r1 → r3
* **Route untuk mencapai hb → r4**hb → r4

Traceroute: **Router 1**  


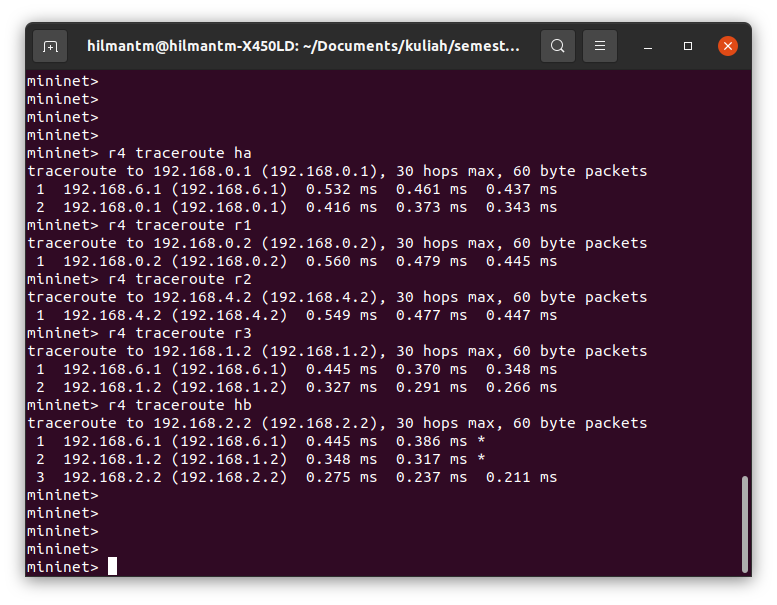
* **Route untuk mencapai r1 → ha**r1 → ha
* **Route untuk mencapai r1 → r2**r1 → r4 → r2
* **Route untuk mencapai r1 → r3**r1 → r3
* **Route untuk mencapai r1 → r4**r1 → r4
* **Route untuk mencapai r1 → hb**r1 → r3 → hb

Traceroute: **Router 2**  


* **Route untuk mencapai r2 → ha**r2 → r3 → r1 → ha
* **Route untuk mencapai r2 → r1**r2 → r3 → r1
* **Route untuk mencapai r2 → r3**r2 → r3
* **Route untuk mencapai r2 → r4**r2 → r4
* **Route untuk mencapai r2 → hb**r2 → r3 → hb

Traceroute: **Router 3**  


* **Route untuk mencapai r3 → ha**r3 → r1 → ha
* **Route untuk mencapai r3 → r1**r3 → r1
* **Route untuk mencapai r3 → r2**r3 → r2
* **Route untuk mencapai r3 → r4**r3 → r2 → r4
* **Route untuk mencapai r3 → hb**r3 → hb

Traceroute: **Router 4**  


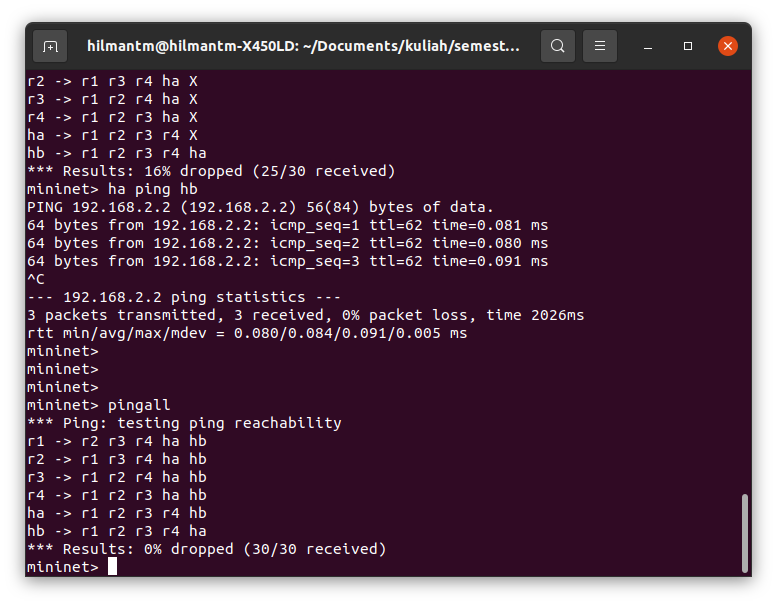
* **Route untuk mencapai r4 → ha**r4 → r1 → ha
* **Route untuk mencapai r4 → r1**r4 → r1
* **Route untuk mencapai r4 → r2**r4 → r2
* **Route untuk mencapai r4 → r3**r4 → r1 → r3
* **Route untuk mencapai r4 → hb**r4 → r1 → r3 → hb

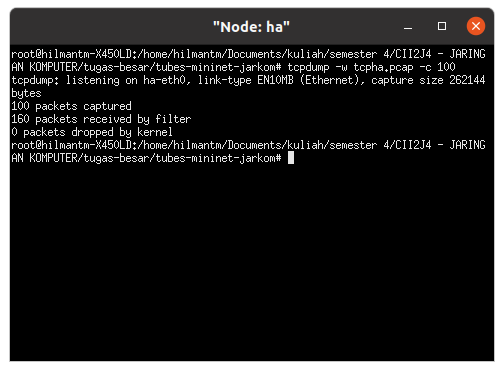
# **CLO 3**

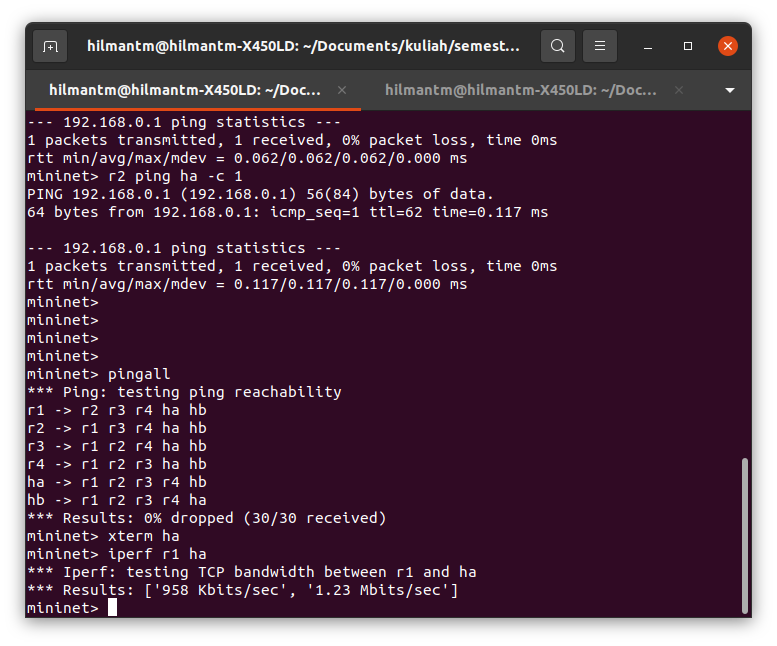
## **Generate traffic menggunakan iPerf**

**Iperf** merupakan open source tools yang digunakan untuk mengukur kecepatan, kecepatan, throughput dan kualitas link jaringan. Tools ini menggunakan protokol TCP dan UDP. TCP digunakan untuk mengukur kecepatan dan throughput link, UDP digunakan untuk mengukur jitter (variasi dari paket ke paket) dan Packet Loss.

Untuk generate traffic menggunakan iPerf, pastikan topologi yang sudah dibangun sudah terkoneksi sempurna (jika pingall, maka tidak ada lagi “X” pada result nya).

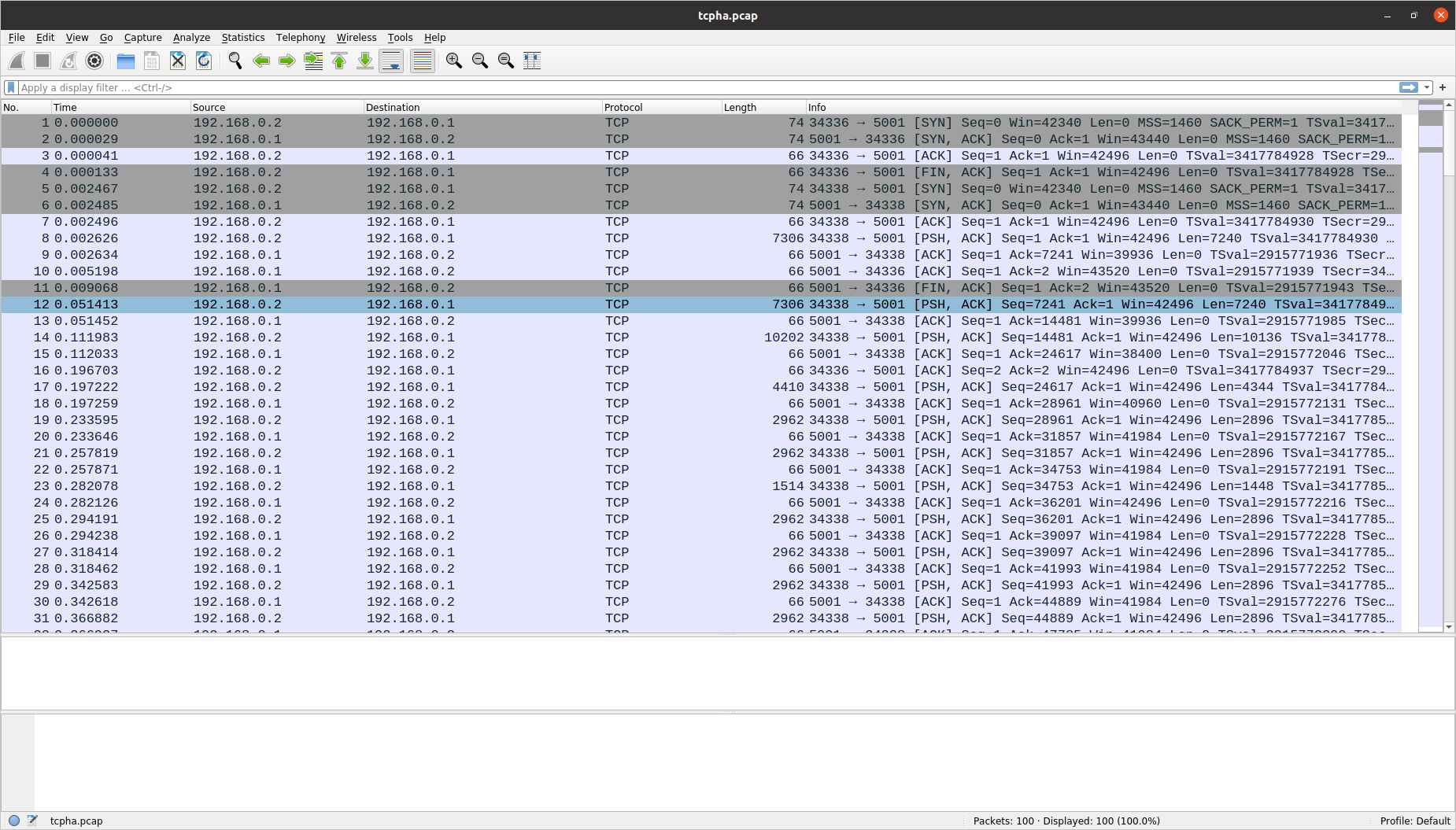


Setelah itu, gunakan **xterm** untuk membuka terminal pada salah satu host untuk di test. Gunakan tcpdump untuk mengcapture pengiriman data menggunakan protokol TCP dan menulisnya kedalam file .pcap agar bisa dibaca melalui aplikasi wireshark.   


Untuk simulasi pengetesan, iperf akan dilakukan dari **router 1** ke **host A**.  


## **Capture traffic menggunakan custom script atau wireshark untuk diinspeksi, dibuktikan dengan traffic di wireshark/tcpdump**

Setelah traffic di capture kedalam file .pcap, maka buka file tersebut menggunakan wireshark untuk memvalidasi apakah benar protokol yang digunakan adalah TCP.



Dari hasil diatas, benar bahwa protokol pengetesan iperf menggunakan TCP.

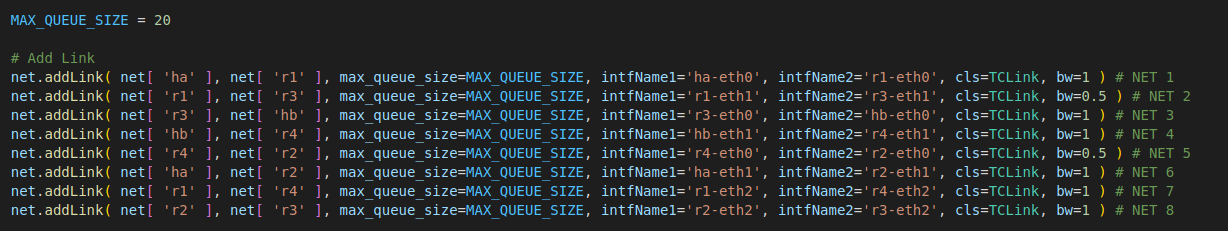
**Lantas, apa perbedaan TCP dan UDP?**

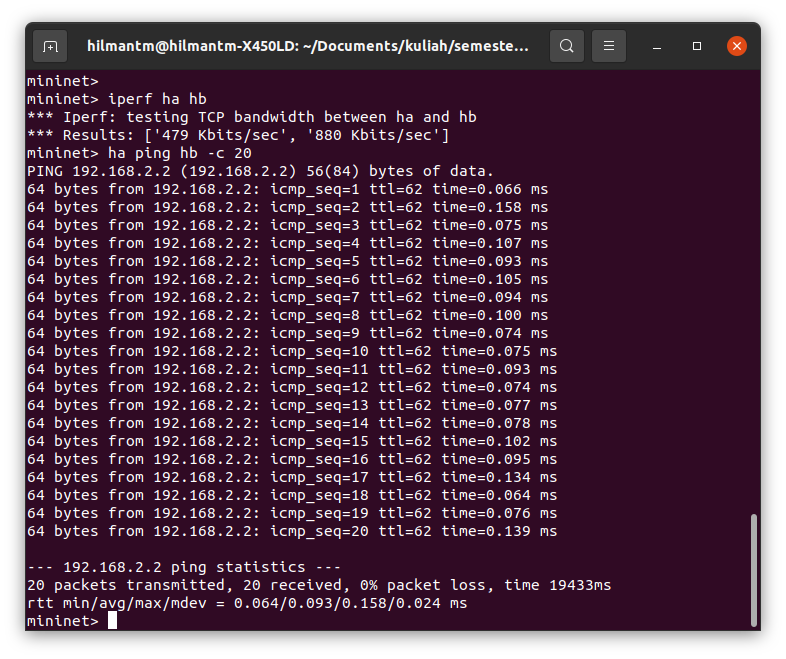
TCP memiliki kelebihan yaitu dapat memastikan keakuratan data yang sampai ke penerima. Sedangkan UDP memiliki keunggulan dalam hal kecepatan transfer data. Penerapan kedua protokol tersebut dapat disesuaikan dengan kebutuhan. UDP merupakan jenis protokol yang memiliki karakteristik connectionless atau tidak berbasis koneksi. Sedangkan TCP adalah kebalikannya, yaitu berbasis koneksi.

# **CLO 4**

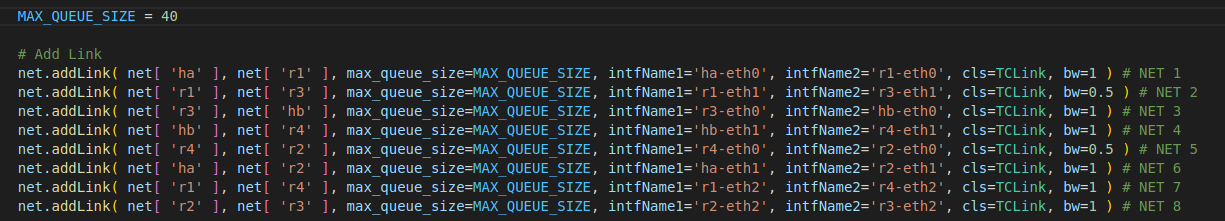
## **Set ukuran buffer pada router : 20, 40, 60 dan 100 dan Capture pengaruh ukuran buffer terhadap delay**

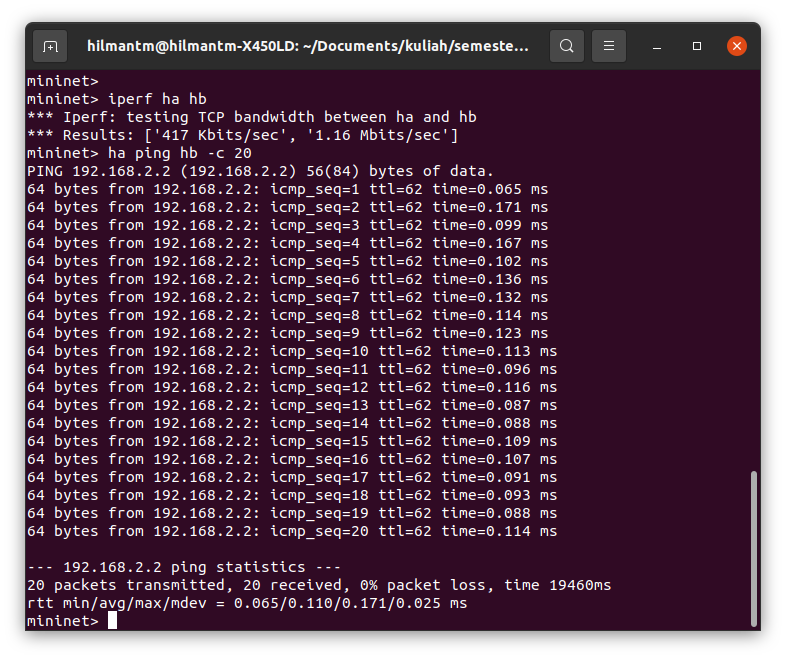
Buffer size: **20**



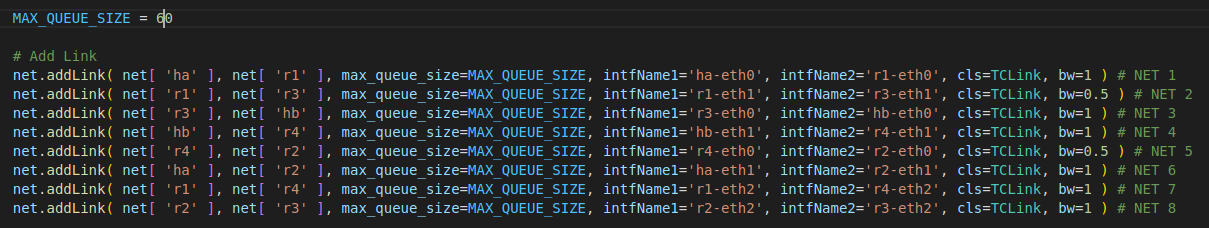


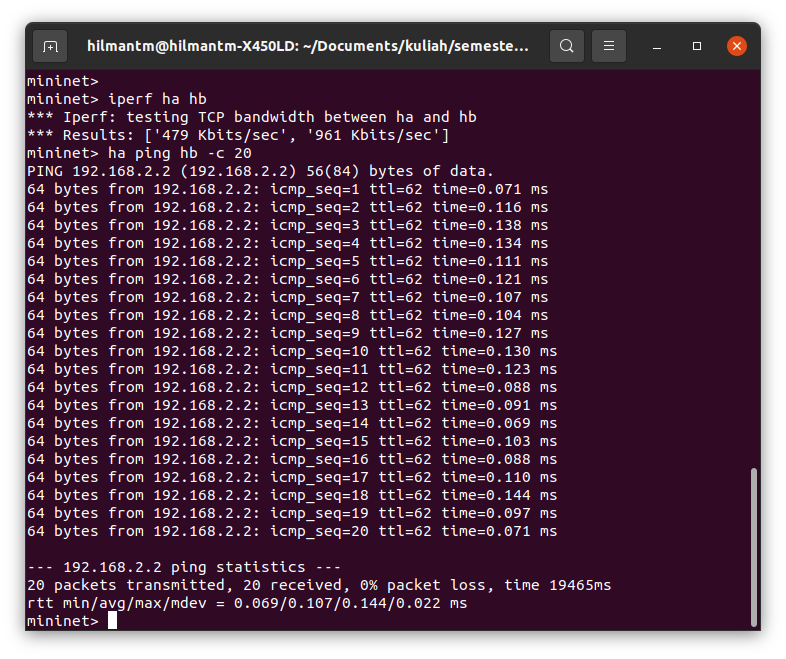
Buffer size: **40**



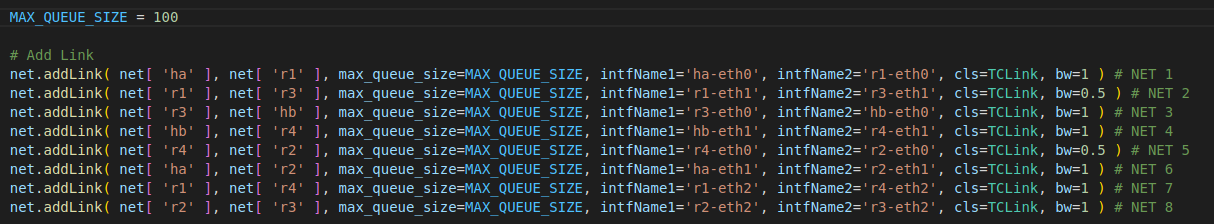


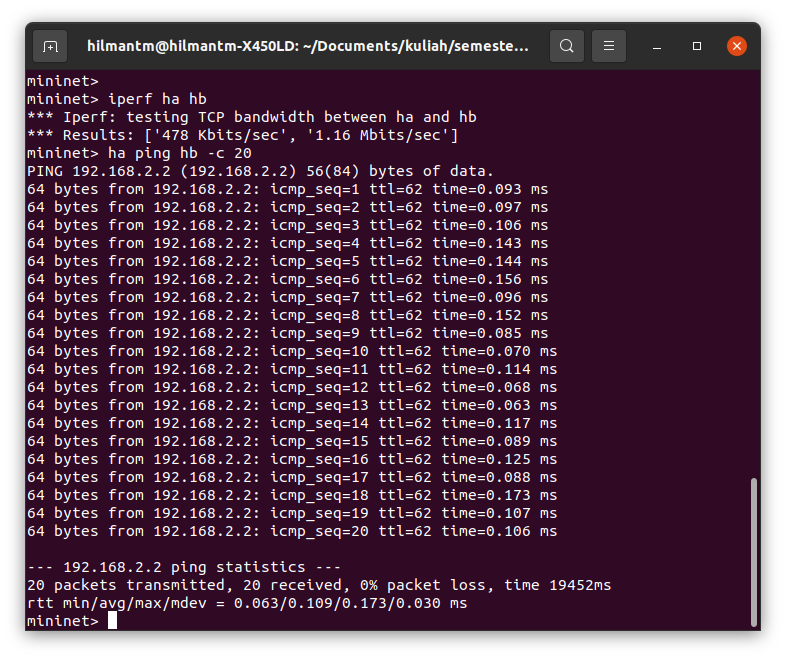
Buffer size: **60**





Buffer size: **100**





**Kesimpulan:**

Apabila dilihat dari hasil test iPerf, tidak ada ketentuan yang pasti apa pengaruh buffer terhadap iPerf test. Contohnya pada buffer 60 dan 100, waktu result turun.

Apabila dilihat dari hasil test ping, maka ada sedikit perbedaan pada time nya. Semakin besar buffer semakin tinggi time yang diperlukan walaupun tidak naik signifikan.