

Analisis Performa Metode HTB - Traffic Shaping Layer 7 Pada Manajemen Bandwidth

Analysis of HTB Method Performance - Layer 7 Traffic Shaping in Bandwidth Management

Jatmika^{1*}, Farelia Devi Julia Putri¹, Yo'el Pieter Sumihar¹, Haeni Budiati¹

¹*Fiskom, Informatika, Universitas Kristen Immanuel Yogyakarta, Yogyakarta 55571, Indonesia*

(*Email Korespondensi: jatmika@ukrimuniversity.ac.id)

Abstrak: Internet sangat dibutuhkan untuk mendapatkan informasi. Internet merupakan komponen penting dalam bidang pendidikan, perkantoran, bisnis, perusahaan, dan lain sebagainya. Perkembangan Internet semakin pesat, tetapi jika tidak dimanajemen dengan baik bandwidth-nya akan mempengaruhi kinerja jaringan. Pada jaringan Internet saat ini sering terjadi masalah, dimana pada layanan tertentu bisa mengkonsumsi bandwidth dalam jumlah besar mengakibatkan layanan lain tidak bisa mendapat bandwidth sesuai kebutuhan. Penelitian ini melakukan pembagian bandwidth ke dalam beberapa kelas, melakukan pembatasan traffic pada tiap level, membuat alokasi aktivitas upload dan download. Manajemen bandwidth dilakukan dengan menggunakan metode HTB (Hierarchical Token Bucket). Pembagian bandwidth dilakukan secara hirarki yang dibagi ke dalam beberapa kelas. Performa jaringan lebih terjamin jika menambahkan monitoring traffic dengan menggunakan metode Traffic Shaping. Berdasarkan hasil penelitian, didapatkan performa jaringan menggunakan standart TIPHON untuk pengukuran parameter performa jaringan menggunakan metode HTB (Hierarchical Token Bucket) dan Traffic Shaping memiliki rata-rata 3,5 dan termasuk dalam kategori "Baik", sedangkan pengujian tanpa menggunakan metode HTB (Hierarchical Token Bucket) dan Traffic Shaping memiliki rata-rata 2,75 dan termasuk kategori "Kurang Baik".

Kata Kunci: Internet, Bandwidth, HTB, Traffic Shaping, TIPHON

Abstract: Internet is needed to get information. The internet is an important component in the fields of education, offices, businesses, companies, and so on. The development of the Internet is growing rapidly, but if it is not managed properly its bandwidth will affect network performance. In today's Internet network problems often occur, where certain services can consume large amounts of bandwidth resulting in other services not being able to get the bandwidth as needed. This research divides bandwidth into several classes, limits traffic at each level, allocates upload and download activities. Bandwidth management is done using the HTB (Hierarchical Token Bucket) method. The distribution of bandwidth is done in a hierarchical manner which is divided into several classes. Network performance is more guaranteed if you add traffic monitoring using the Traffic Shaping method. Based on the results of the study, it was found that network performance using the TIPHON standard for measuring network performance parameters using the HTB (Hierarchical Token Bucket) method and Traffic Shaping had an average of 3.5 and included in the "Good" category, while the test without using the HTB (Hierarchical Token) method Bucket) and Traffic Shaping have an average of 2.75 and are in the "Not Good" category.

Keywords: Internet, Bandwidth, HTB, Traffic Shaping, TIPHON

Naskah diterima 16 Oktober 2023; direvisi 8 Desember 2023; dipublikasi 30 Desember 2023.
JUISI is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.



1. Pendahuluan

Internet merupakan sebuah kebutuhan penting bagi manusia saat ini. Internet juga sangat dibutuhkan untuk mendapatkan atau mencari sebuah informasi. Internet juga merupakan komponen penting dalam bidang-bidang tertentu seperti pemerintahan, pendidikan, pariwisata dan terutama perusahaan dengan *human resource* yang cukup besar. Pada jaringan Internet saat ini sering timbul masalah dimana pada layanan tertentu bisa mengkonsumsi bandwidth dalam jumlah besar yang menyebabkan layanan lain tidak bisa mendapatkan bandwidth sesuai yang dibutuhkan oleh karena itu, perlu dilakukan manajemen *bandwidth* serta optimalisasi *traffic* sehingga *bandwidth* lebih optimal dan performa jaringan lebih terjamin. Manajemen bandwidth berfungsi untuk mengatur bandwidth dalam jaringan komputer sehingga masing-masing komputer yang terhubung dalam sebuah jaringan memiliki akses Internet yang tertata dengan baik.

Manajemen *bandwidth* bisa diimplementasikan menggunakan beberapa metode, salah satunya adalah metode HTB (*Hierarchical Token Bucket*). Metode ini dapat mengontrol penggunaan Internet yang dilakukan oleh setiap *client*. Teknik antrian metode ini adalah memberikan fasilitas pembatasan *traffic*. Manajemen *bandwidth* perlu dilakukan untuk ukuran media pengiriman data, membagi kecepatan transfer data, dan mengatur besar kecilnya data yang ditransfer yang bertujuan untuk mengurangi *traffic* yang tinggi dalam jangka panjang.

Penelitian sebelumnya yang dilakukan oleh Wijaya, dkk (2012) dengan judul Manajemen *Bandwidth* dengan metode HTB (*Hierarchical Token Bucket*) pada Sekolah Menengah Negeri 5 Semarang bertujuan untuk menerapkan control penggunaan Internet, menerapkan manajemen *bandwidth* dan menstabilkan koneksi Internet. Kelebihan dari penelitian ini antara lain implementasi metode HTB dapat mengontrol penggunaan Internet yang dilakukan oleh setiap *client* dengan baik, komputer – komputer *client* dapat terkoneksi ke Internet dengan pembagian bandwidth yang telah disediakan dan koneksi Internet pada masing-masing komputer *client* menjadi stabil karena adanya pembatas *bandwidth* yang diberikan.

Penelitian yang dilakukan oleh Syaifullah (2017) judul Metode *Traffic Shaping* pada Layer 7 Protocol Untuk Mengoptimalkan Kinerja Jaringan Komputer Menggunakan Mikrotik. Tujuan penelitian yang dibuat untuk mengontrol jumlah volume *traffic* data yang dikirim ke dalam jaringan sesuai alokasi *bandwidth* yang ditentukan. Kelebihan dari penelitian ini adalah penggunaan mekanisme layet 7 protocol pada penerapan *traffic Shaping* dapat meningkatkan kinerja jaringan Internet.

Penelitian yang akan dibangun menggunakan judul Analisis Performa Metode HTB *Traffic Shaping* Layer 7 pada Manajemen *Bandwidth*. Penelitian ini merupakan gabungan dari penelitian yang dilakukan oleh Wijaya, dkk (2012) dan Syaifullah (2017). Alasan memilih judul ini karena pada studi kasus yang digunakan yaitu RS Baptis Kediri memiliki bagian devisi-devisi yang berbeda serta membutuhkan *bandwidth* yang berbeda. Metode HTB yang memiliki pola hirarki dapat menyelesaikan masalah tersebut ditambah dengan metode *Traffic Shaping* yang mampu membuat kinerja jaringan komputer lebih optimal.

2. Metode Penelitian

2.1 Analisa Masalah Sistem Jaringan

Sesuai hasil pengamatan yang telah dilakukan, didapatkan hasil analisis sebagai berikut. Pertama, sistem jaringan yang telah dianalisis terdapat infrastruktur jaringan lokal atau biasa disebut dengan LAN (*Local Area Network*). Menggunakan ISP Biznet, satu router Mikrotik yang berfungsi sebagai pusat jaringan kemudian dipecah menjadi beberapa jalur menggunakan *switch*. Kedua, sistem jaringan ini setiap hari digunakan oleh cukup banyak *client* dan membutuhkan sumberdaya yang cukup besar, selain itu sistem jaringan yang sudah terimplementasi belum bisa membagi *bandwidth* secara berkelompok atau hirarki yang dapat membagi bandwidth berdasarkan kebutuhan *client*, dan akhirnya semua *client* mendapat *bandwidth* yang tidak stabil.

2.2 Analisa Kebutuhan Perangkat Pendukung Perancangan Jaringan

2.2.1 Analisis Kebutuhan Perangkat Keras

Satu buah laptop dengan spesifikasi sebagai berikut ini. Processor minimal AMD A4, RAM 4GB DDR4, *Storage* SSD 120GB HDD 500GB, dan *Internet Broadband Connection*. GNS3, digunakan untuk membuat desain topologi sistem jaringan dan menghubungkan ke *Virtualbox*. *Virtualbox*, digunakan untuk membuat simulasi Mikrotik *RouterOS* dengan spesifikasi sebagai berikut: Processor minimal AMD A4; RAM 64MB; *Storage* 2GB; Serta *Windows 7* yang diinstall pada *Virtualbox*, dengan spesifikasi sebagai berikut: Processor minimal AMD A4; RAM 1024MB; *Storage* 10GB.

2.2.2 Analisis Kebutuhan Perangkat Lunak

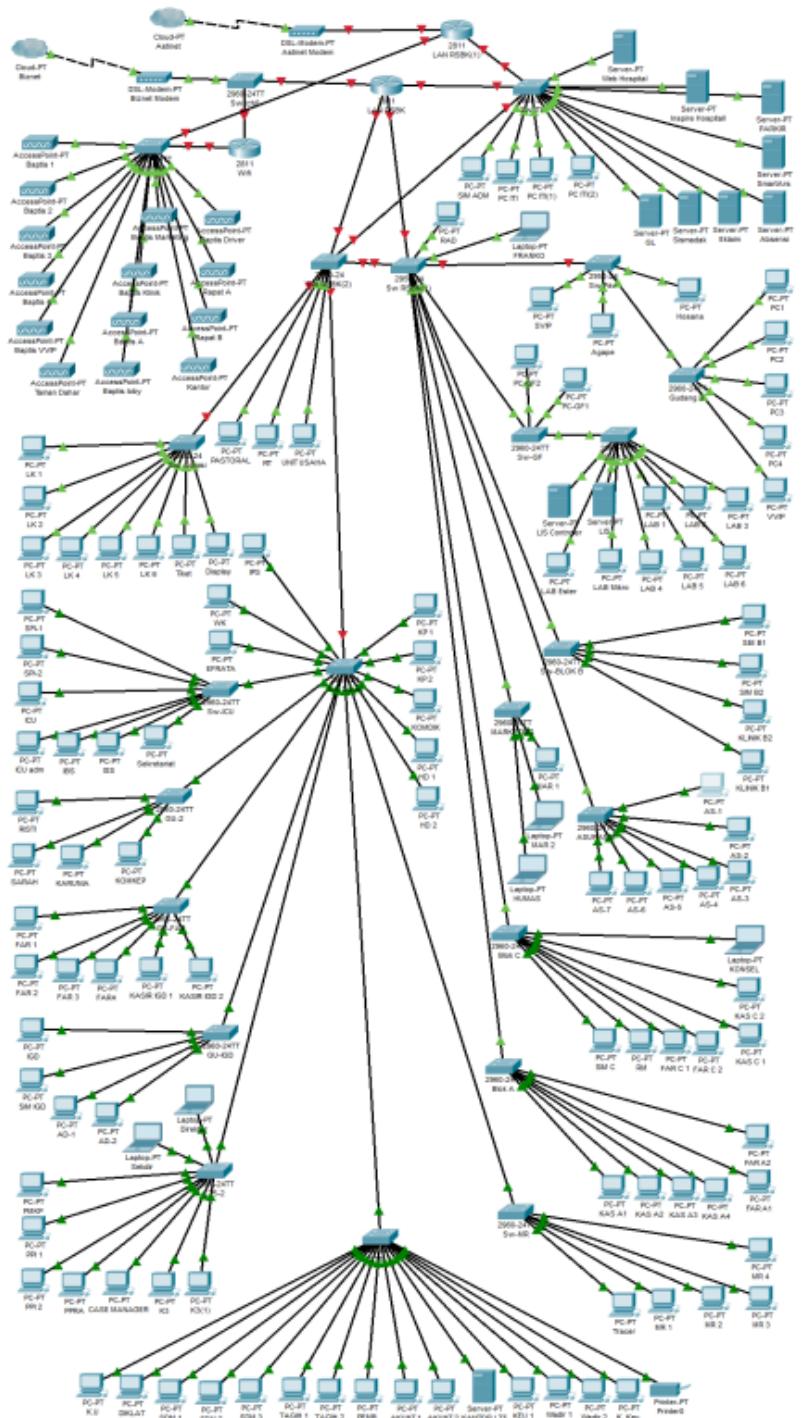
Berikut adalah daftar perangkat lunak yang diperlukan dalam penelitian kami. Pertama, *Notepad++v7.5.8* digunakan untuk menyimpan draft sementara daftar pustaka. Kedua, *Browser Google Chrome* yang digunakan untuk mencari referensi dan landasan teori. Ketiga, *Speedtest*, digunakan untuk pengujian *bandwidth*. Dan yang terakhir, *Network Analyzer Wireshark*, digunakan untuk Analisa HTB dan *Traffic Shaping*.

2.3 Desain Sistem Jaringan

2.3.1 Desain Topologi Jaringan

Berdasarkan hasil analisis yang telah dilakukan pada topologi jaringan, dapat diketahui bahwa semua jaringan hanya berpusat pada satu router LAN RSBK yang terhubung dengan ISP Biznet dan dilengkapi dengan beberapa *switch* yang digunakan untuk penghubung antar perangkat komputer sehingga dapat melakukan pertukaran paket serta meneruskan data ke berbagai perangkat tujuan. Terdapat 2 ISP pada topologi yaitu ISP Biznet dan ISP Astinet, namun infrastruktur jaringan yang ada hanya menggunakan 1 ISP yaitu Biznet, sedangkan ISP Astinet digunakan sebagai backup jika terjadi masalah pada ISP Biznet.

Perangkat yang terhubung dengan *switch* RSBK 1 menggunakan IP *Address* 190.164.9.1/24, sedangkan perangkat yang terhubung dengan *switch* RSBK 2 menggunakan IP *Address* 192.168.10.1/24. Dapat dilihat pada Gambar 1 di bawah ini.

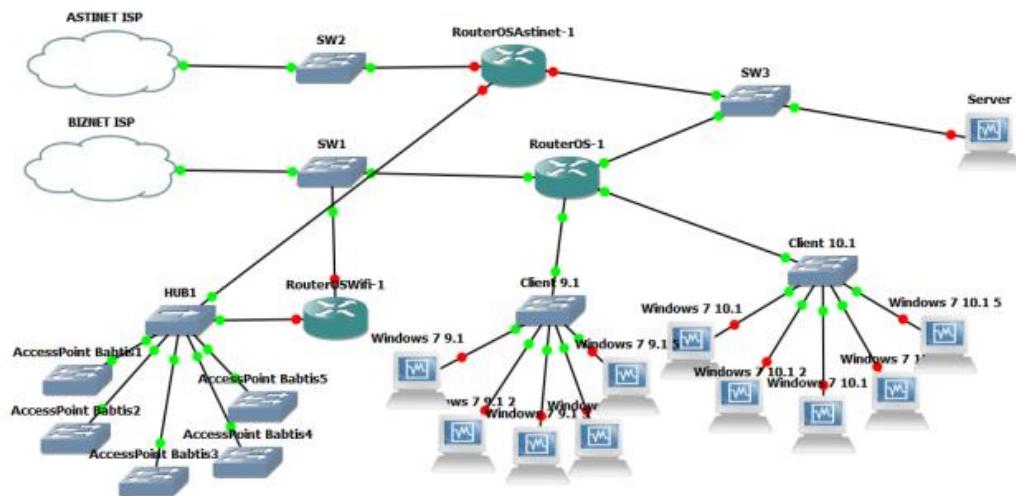


Gambar 1. Topologi Jaringan di Lapangan

2.3.2 Desain Topologi Baru Dengan Metode HTB

Desain topologi jaringan baru digunakan untuk menambahkan metode HTB (*Hierarchical Token Bucket*) dan metode *Traffic Shaping* ke dalam topologi lama yang sudah pernah dibuat sebelumnya. Pembuatan topologi jaringan baru menggunakan perangkat lunak GNS3, *Virtualbox*. Pada *Virtualbox* sudah terinstal sistem operasi *Windows 7*, di dalam *Windows 7* sudah terinstal juga perangkat lunak *Winbox* yang digunakan untuk pengaturan dan konfigurasi metode HTB serta *Traffic Shaping*. Setelah konfigurasi pada *Winbox* selesai, perangkat lunak *Virtualbox* akan dihubungkan dengan *GNS3* untuk membuat topologi baru. Jadi dapat disimpulkan perangkat lunak *GNS3* saling berintegrasi dengan *Virtualbox*.

Pada topologi baru ini terdapat 2 ISP kemudian dihubungkan dengan *switch* dan *Mikrotik RouterOS* serta dihubungkan dengan PC *client* *Windows 7*. PC *client* ini mencakup semua *client* yang memiliki IP *Address* 190.164.9.1/24 dan 192.168.10.1/24. Dapat dilihat pada Gambar 2 di bawah ini.



Gambar 2. Desain Topologi Baru Dengan Metode HTB.

2.4 Implementasi Simulasi

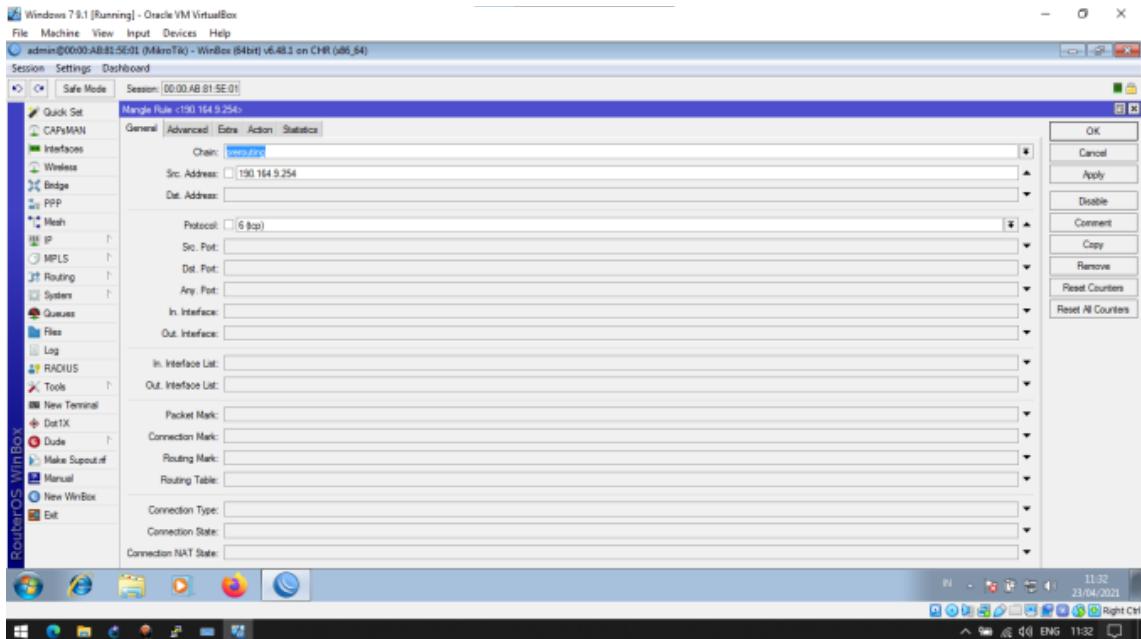
Implementasi sistem jaringan RS. Baptis dibangun secara simulasi dengan tahapan sebagai berikut:

1. Konfigurasi Mikrotik RouterOS.
IP Address:
 - a. 182.253.186.140/24 (pada ether1, Static)
 - b. 190.164.9.1/24 (pada ether2, DHCP)
 - c. 192.168.10.1/24 (pada ether3, DHCP)

2. Konfigurasi *VirtualBox Client 9.1*
 - a. IP Address: 190.164.9.1/24 (pada ether2, DHCP)
 - b. Berpusat pada *Switch RSBK 1*
3. Konfigurasi *VirtualBox Client 10.1*
 - a. IP Address: 192.168.10.1/24 (pada ether3, DHCP)
 - b. Berpusat pada *Switch RSBK 2*

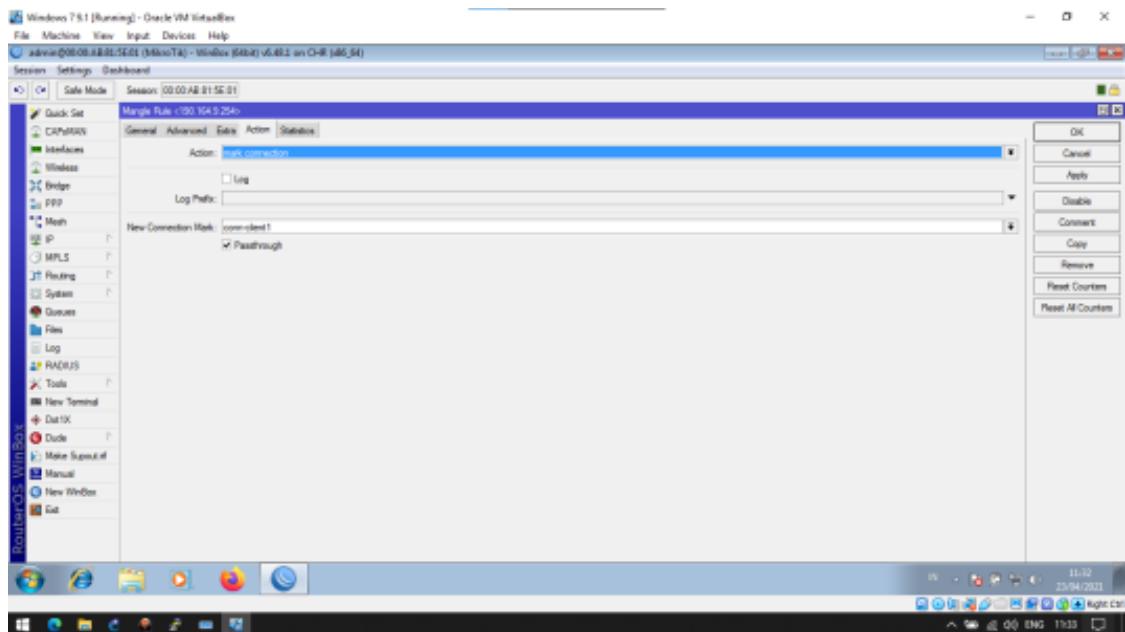
2.4.1 Konfigurasi Mangle

Queue tree dan bagaimana hubungannya dengan HTB. Hal yang sangat menentukan dalam pembuatan *queue tree* adalah *packet marks*. *Packet Marks* inilah yang harus dibuat menggunakan *mangle*. Pada menu IP → Firewall → Mangle hal pertama yang harus dilakukan adalah membuat koneksi menggunakan IP Address client dengan *chain prerouting*, kemudian klik tab menu *Action* dan pilih *actionnya* menjadi *mark connection* lalu beri nama pada *New Connection Mark* dan centang *passthrough*, kemudian *apply* → ok. Lakukan hal yang sama untuk setiap IP Address baik untuk *upload* maupun *download*. Dapat dilihat pada Gambar 3 dibawah ini.



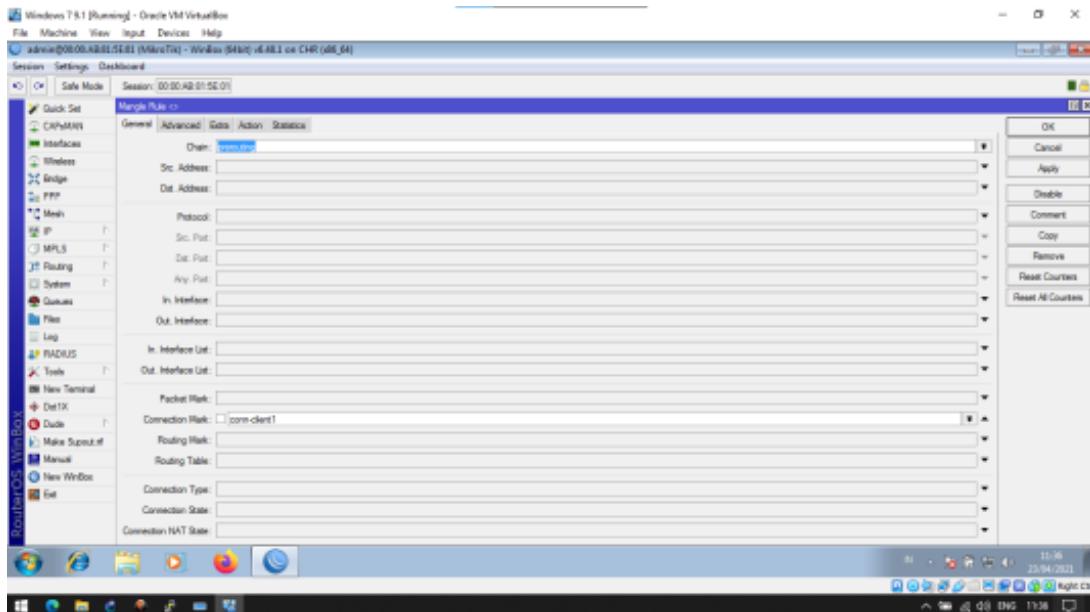
Gambar 3. Konfigurasi Conn-Mark Upload Client 9.254

Setelah membuat konfigurasi *Connection mark*, hal berikutnya membuat *Action Connection Mark* dengan action *Mark Connection* lalu *passthrough*. Lakukan hal yang sama untuk semua *Mark Connection client* setiap IP Address baik untuk upload maupun download yang memiliki proses sama. *Action Conn-Mark* dapat dilihat pada Gambar 4 dibawah ini.



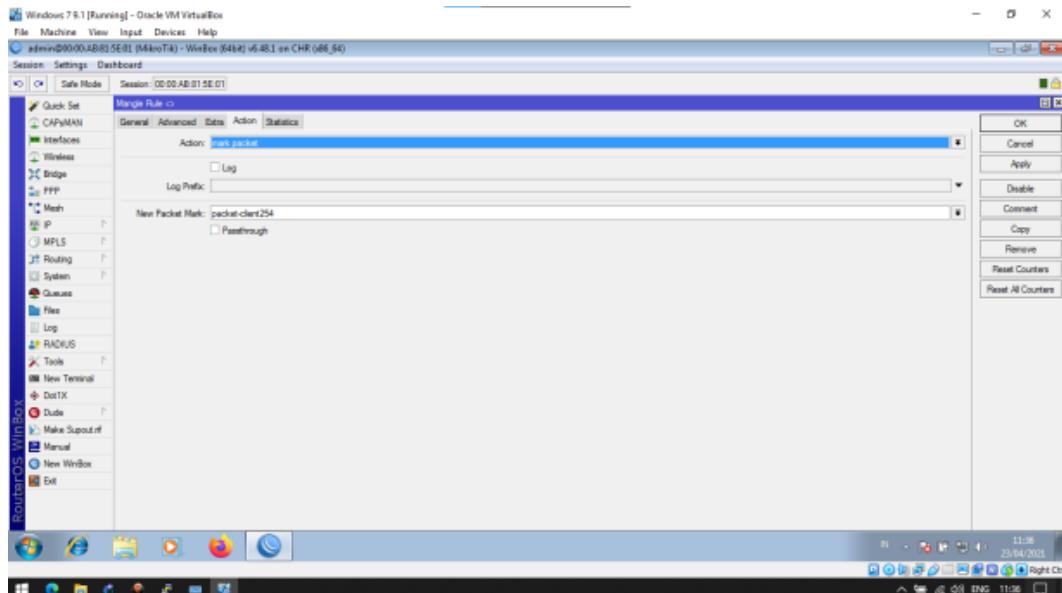
Gambar 4. Action Conn-Mark Upload Client 9.254

Hal kedua yang harus dilakukan untuk membuat *mangle* adalah membuat *packet marks* untuk menangkap koneksi yang sudah dibuat tadi. Cara membuatnya tambahkan *mangle* baru lalu pilih chain *prerouting* untuk upload dan chain *forward* untuk *download* → ganti *Connection Marks* sesuai dengan *Connection Marks* yang sudah dibuat sebelumnya → klik tab menu *action* → pilih *action* menjadi *Mark Packet* → beri nama pada *New Packet Marks* → jangan centang *passthrough* → apply dan ok. Lakukan pada semua IP Address *client* baik *upload* maupun *download*. Konfigurasi *Packet Marks* upload dapat dilihat pada Gambar 5 dibawah ini.



Gambar 5. *Packet Marks Upload Client 9.254*

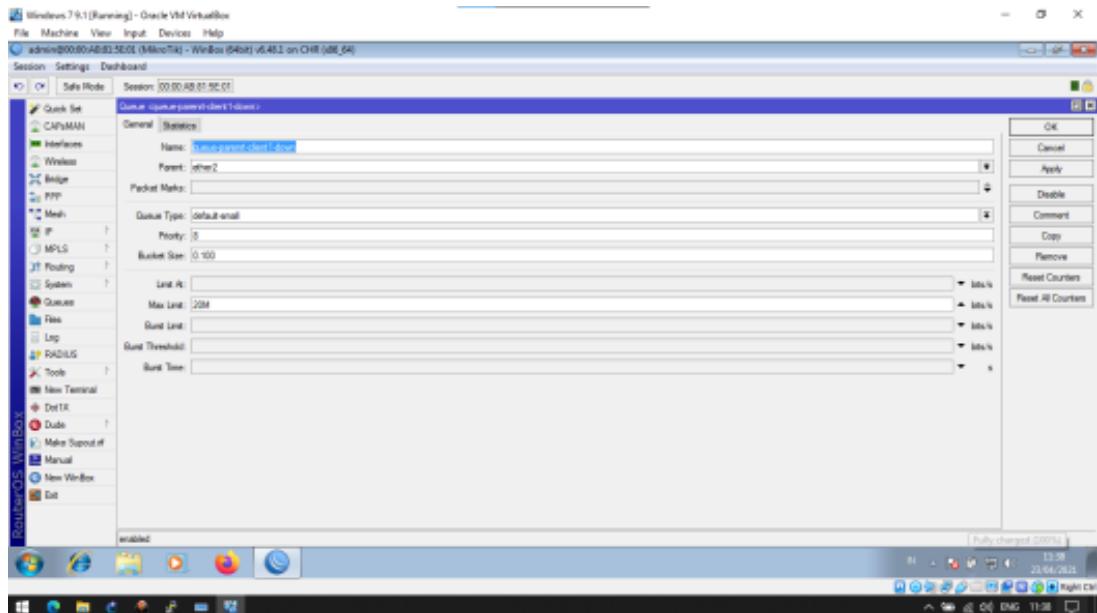
Seperti *Connection Mark*, *packet mark* juga membutuhkan action untuk menjadi aksi dari *packet mark* tersebut. Lakukan hal yang sama pada setiap *packet marks client* dengan konfigurasi yang sama baik untuk *upload* maupun *download*. Konfigurasi action *packet mark* dapat dilihat pada Gambar 6 dibawah ini.



Gambar 6. *Action Packet-Mark Upload Client 9.254*

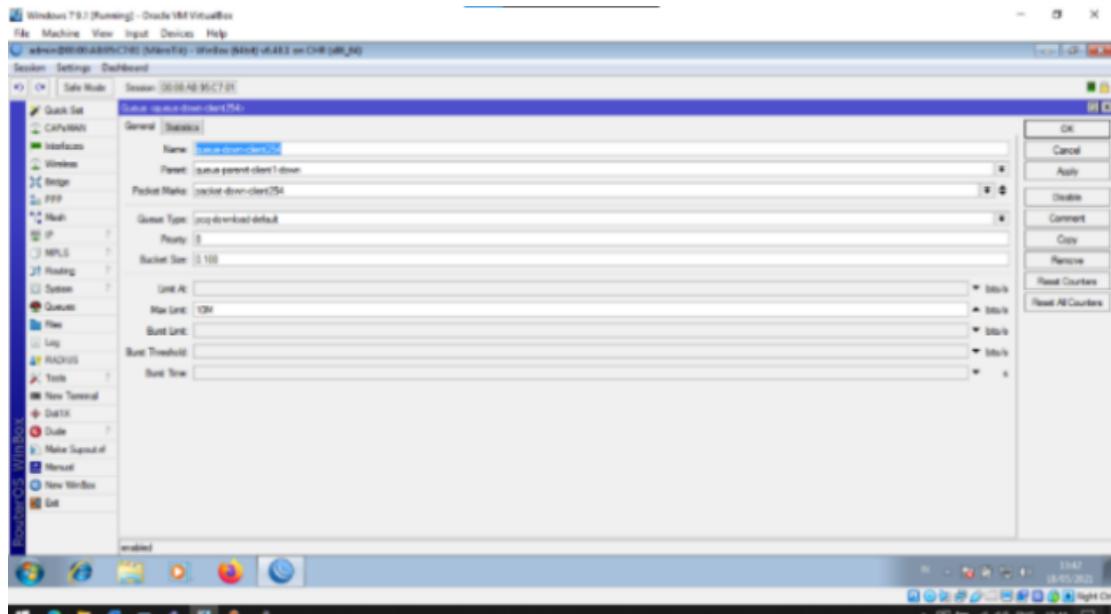
2.4.2 Konfigurasi Queue Tree HTB

Setelah melakukan konfigurasi *mangle*, hal selanjutnya yang harus dilakukan adalah konfigurasi *Queue Tree*. Pada menu *Queue* → *Queue Tree* dan tambahkan *Queue Tree* baru. Hal pertama kita akan membuat *parent*-nya. Beri nama *queue parent* sesuai keinginan, kemudian tambahkan *parent*, *parent* yang digunakan adalah *interface* keluar dari router, disini *interface* keluarnya adalah *ether2*. Untuk *parent* tidak perlu menentukan *packet marks*, *packet marks* nanti akan ditentukan oleh *child*. Selanjutnya beri limitasi pada *max limit*. Jadi pada *interface* tersebut secara total dapat menyalurkan *traffic* berapa banyak, kemudian *apply* → ok, lakukan hal yang sama untuk *upload* dan *download* setiap *client* dengan menggunakan proses yang sama. Konfigurasi *parent download client 1* dapat dilihat pada Gambar 7 dibawah ini.



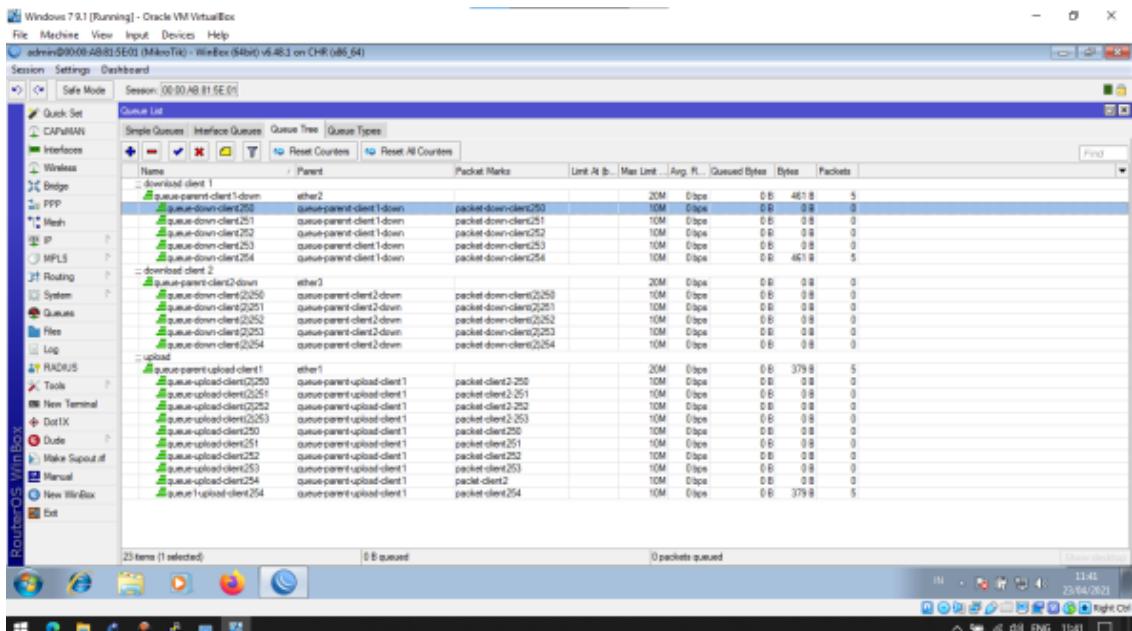
Gambar 7. Konfigurasi Parent Download Client 1

Mengenai HTB, tambahkan baru *Queue Tree* lagi untuk membuat *child*-nya. Beri nama *queue* sesuai keinginan, kemudian tentukan *packet marks*. *Packet marks* tersebut harus sama dengan yang sudah dibuat pada *mangle* tadi. Selanjutnya tentukan *parent*. Nama *parent* yang akan dipilih pastikan sudah sama dengan *parent* yang sudah dibuat tadi. Tentukan *max limitnya* berbeda dengan *parent*, kemudian *apply* → ok. Lakukan konfigurasi yang sama untuk setiap IP Address *client* baik *upload* maupun *download*. Konfigurasi *child download client 1* dapat dilihat pada Gambar 8 di bawah ini.

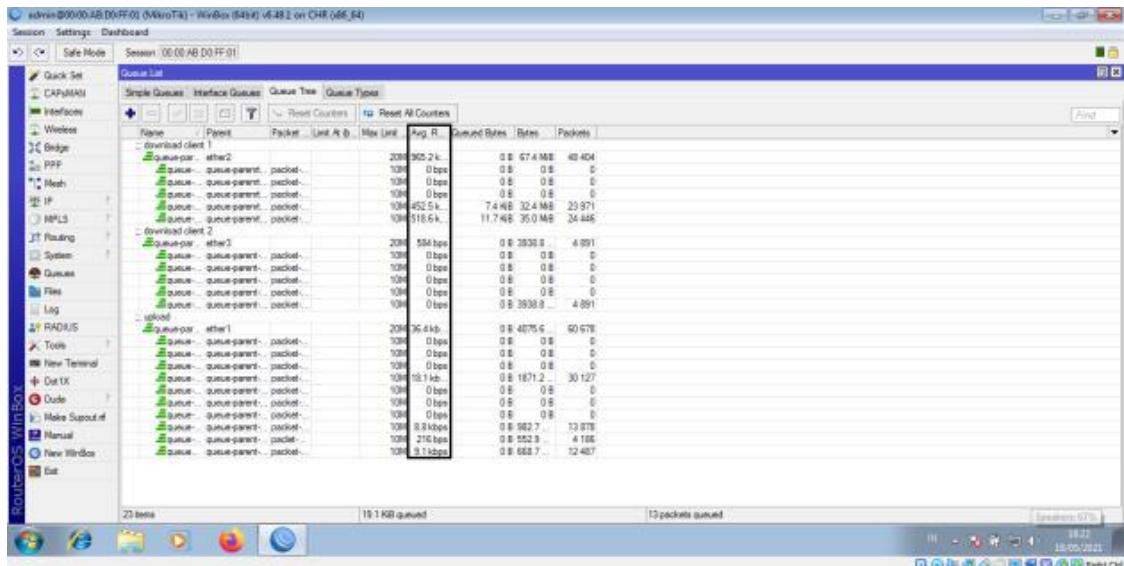


Gambar 8. Konfigurasi *Child Download Client* 1

Queue untuk *child*-nya sudah menjadi bagian dari *parent*. Pada gambar dibawah ini bisa kita lihat juga *traffic*-nya. *Traffic* yang tampak pada *parent* akan terlihat sebagai penjumlahan dari *traffic* keseluruhan *child*-nya. Disinilah fungsi HTB bekerja. HTB merupakan akumulasi dari *traffic* untuk *client*-nya. Dapat dilihat pada Gambar 9 dan Gambar 10 dibawah ini.



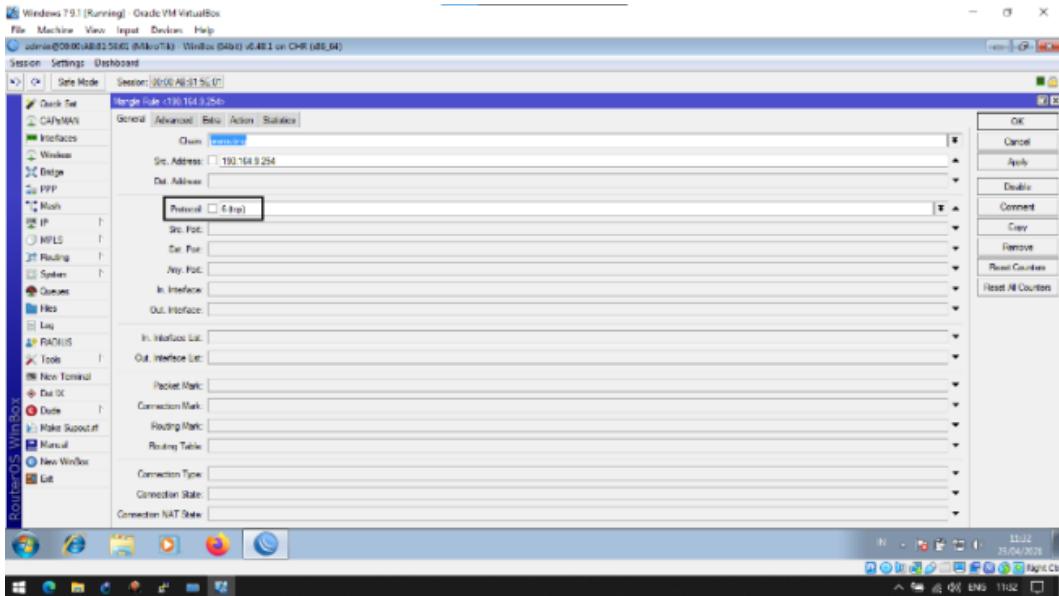
Gambar 9. Hasil Konfigurasi *Queue Tree*



Gambar 10. HTB sudah terlihat pada *Average rate*

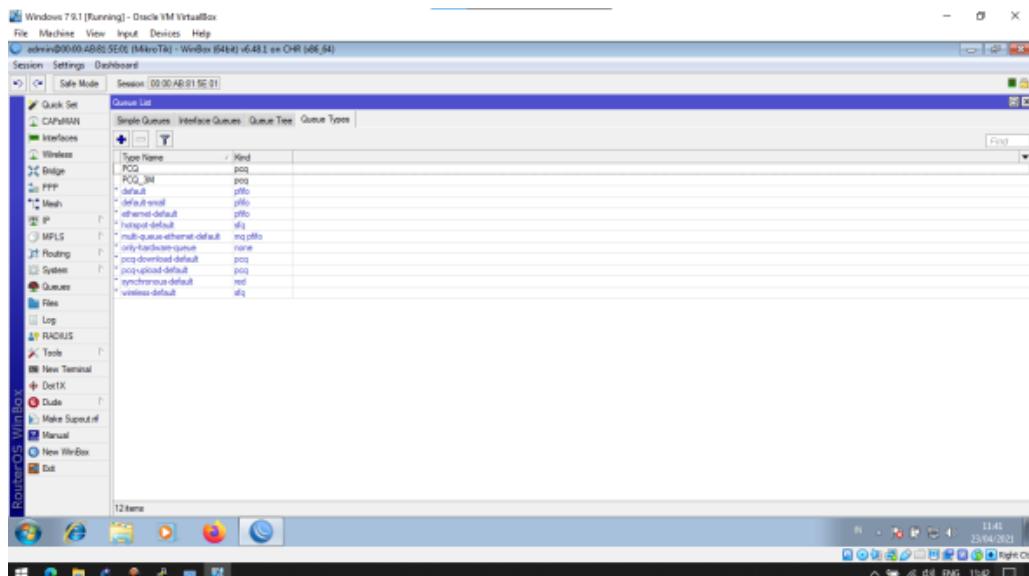
2.4.3 Konfigurasi Traffic Shaping

Traffic Shaping digunakan untuk mengoptimalkan *traffic* atau lalu lintas data. *Bandwidth* memiliki batasan yang harus diperhatikan, oleh karena itu perlu dilakukan optimisasi *traffic* untuk menghindari kepadatan data, tabrakan data, dan kehilangan data, maka dilakukan *priority traffic* pada lalu lintas jaringan. Konfigurasi *Traffic Shaping* sebagai berikut. Pada menu IP → Firewall → mangle, buka kembali *connection mark* yang sudah dibuat pada *mangle* tadi kemudian tambahkan *protocol 6(tcp)*. Tambahkan *protocol 6(tcp)* pada semua *connection mark* baik untuk *upload* maupun *download*. Dapat dilihat pada Gambar 11 dibawah ini.



Gambar 11. Tambahkan *Protocol TCP* Untuk setiap *client*

Hal kedua yang harus dilakukan adalah memastikan *Queue type* sudah ada *default* untuk *PCQ download* dan *PCQ upload*. Jika belum ada *default* maka tambahkan baru pada menu *Queue* → *Queue Type* → tambahkan *queue type* pada setiap bagian *child*. Berhubung pada penelitian ini sudah tersedia *default* jadi disini kita menggunakan *default* saja. Seperti Gambar 12 dibawah ini.



Gambar 12. *Queue Type Default*

3. Hasil dan Pembahasan

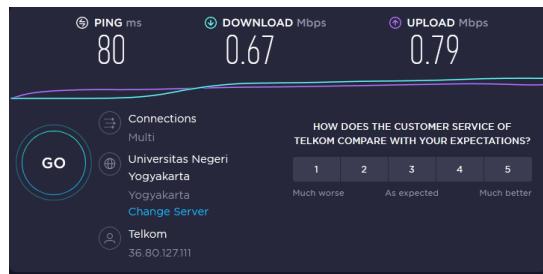
3.1 Evaluasi dan Hasil Peracangan

Berdasarkan perancangan sistem yang telah dilakukan selama masa penggerjaan penelitian, maka dibuat evaluasi sebagai bahan acuan untuk pengembangan sistem jaringan serta untuk membuktikan hasil dari implementasi perancangan sistem jaringan ini secara simulasi sebagai berikut. Pertama, pengujian metode HTB dan *Traffic Shaping* yang telah dikonfigurasi menggunakan perangkat lunak seperti *Speedtest*. Kedua, pengujian *Capture Data Traffic* pada masing-masing *client* menggunakan perangkat lunak *Wireshark Analyzer Traffic*. Ketiga, pengujian Analisa pengukuran parameter performa jaringan.

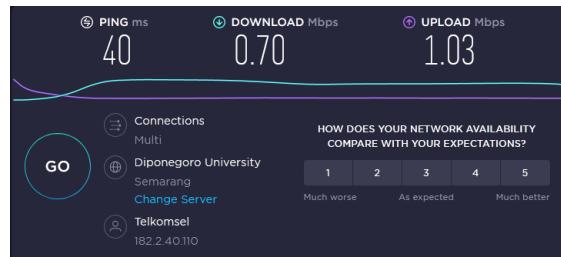
3.2 Hasil Pengujian

3.2.1 Hasil Pengujian Bandwidth Metode HTB dan Traffic Shaping

Pada Gambar 13 dilakukan pengujian bandwidth metode HTB dan *Traffic Shaping* dan pada Gambar 14 dilakukan pengujian bandwidth tanpa menggunakan metode HTB *Traffic Shaping* yang telah dikonfigurasi menggunakan perangkat lunak *Speedtest*. Pengujian dilakukan dengan cara PC *client* dihidupkan pada GNS3 kemudian masuk pada *Winbox* dan lakukan *test* kecepatan Internet menggunakan *Speedtest*, disini dapat dilihat bahwa metode HTB dan *Traffic Shaping* sudah berhasil dilakukan. Lakukan pengujian yang sama pada setiap *client*.



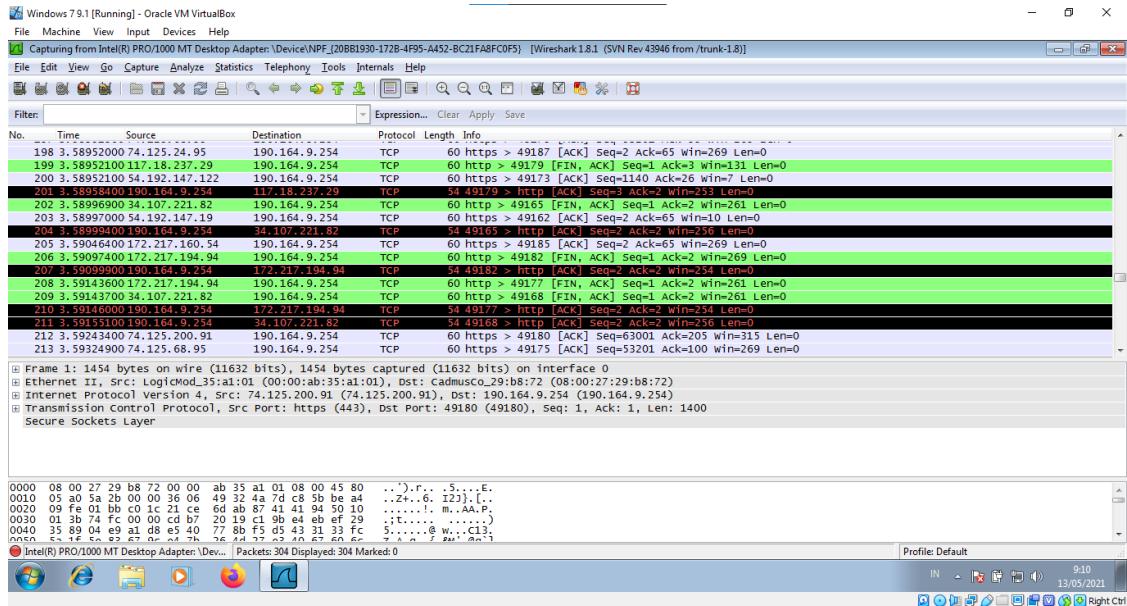
Gambar 13. Pengujian Bandwidth dengan metode HTB *Traffic Shaping*



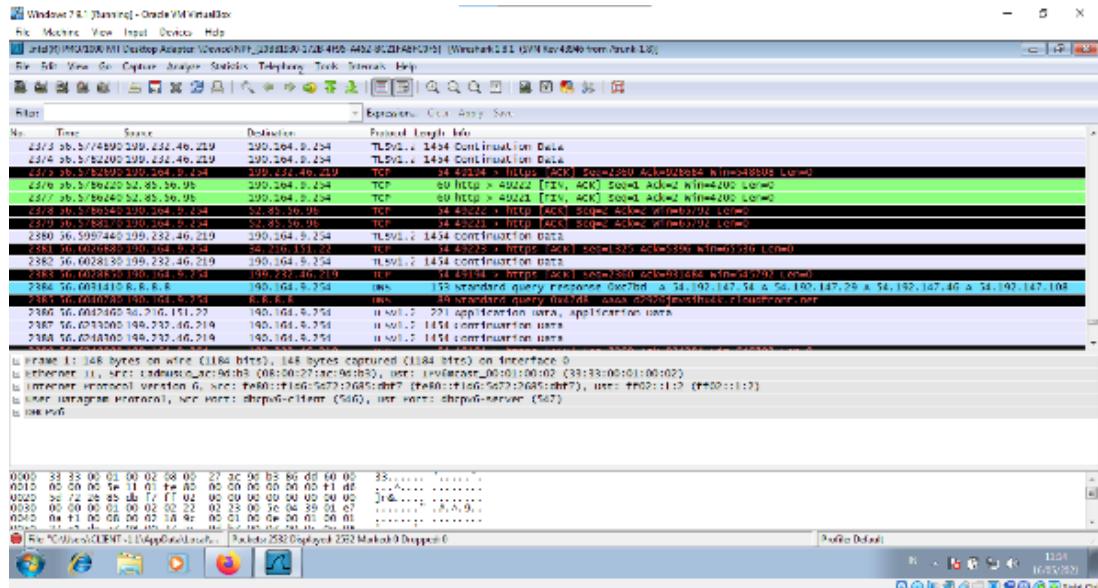
Gambar 14. Pengujian Bandwidth tanpa metode HTB *Traffic Shaping*

3.2.2 Hasil Pengujian *Capture Data Traffic* pada masing-masing *client*

Pengujian pada tahap ini cukup mudah untuk dilakukan menggunakan perangkat lunak *Wireshark Analyzer Traffic*. Caranya dengan membuka *interface list* yang ada pada *Wireshark Analyzer Traffic* yang memiliki paket dan tunggu hasilnya selama 1 menit. Lakukan pengujian untuk semua *client* yang menggunakan metode HTB *Traffic Shaping* dan tanpa menggunakan metode. Dapat dilihat pada Gambar 15 dan Gambar 16 dibawah ini.



Gambar 15, Capture Data Traffic dengan metode HTB Traffic Shaping



Gambar 16. Capture Data Traffic tanpa metode HTB Traffic Shaping

3.2.3 Analisa Hasil Pengukuran Parameter Performa Jaringan

Pada tahap ini dilakukan pengujian *performa* jaringan menggunakan perangkat lunak *Wireshark Analyzer Traffic*. Pada pengujian ini akan menggunakan parameter-parameter *QoS (Quality of Service)* yang terdiri dari *Throughput*, *Delay*, *Jitter*, dan *Packet Loss* dengan menggunakan rumus *TIPHON*. Lakukan perhitungan parameter *Throughput* untuk setiap IP Address *client* yang menggunakan metode HTB *Traffic Shaping* maupun yang tidak menggunakan

metode. Hasil perhitungan dapat dilihat pada Tabel 1 dan Tabel 2 dibawah ini.

Tabel 1. Pengukuran Parameter *Throughput* dengan *HTB Traffic Shaping*

No	IP Address Client	Rata-Rata Throughput (bps)	Indeks	Kategori
1	190.164.9.254	63,385	2	Sedang
2	190.164.9.253	22,51	1	Buruk
3	192.168.10.254	93,29	4	Sangat Baik
4	192.168.10.253	80,01	4	Sangat Baik

Tabel 2. Pengukuran Parameter *Throughput* tanpa *HTB Traffic Shaping*

No	IP Address Client	Rata-Rata Throughput (bps)	Indeks	Kategori
1	190.164.9.254	10,281	1	Buruk
2	190.164.9.253	5,895	1	Buruk
3	192.168.10.254	8,527	1	Buruk
4	192.168.10.253	10,505	1	Buruk

Perhitungan parameter yang kedua yaitu *Delay* dengan menggunakan standart *TIPHON* seperti pada Tabel 3 dibawah ini. *Delay* adalah waktu keterlambatan transmisi data dari pengirim ke penerima. Lakukan perhitungan parameter *Delay* untuk setiap *IP Address client* yang menggunakan metode *HTB Traffic Shaping* maupun yang tidak menggunakan metode. Hasil perhitungan dapat dilihat pada Tabel 3 dan Tabel 4 dibawah ini.

Tabel 3. Pengukuran Parameter *Delay* dengan *HTB Traffic Shaping*

No	IP Address Client	Rata-Rata Delay (ms)	Indeks	Kategori
1	190.164.9.254	10,2	4	Sangat Baik
2	190.164.9.253	1,128	4	Sangat Baik
3	192.168.10.254	287,3	3	Baik
4	192.168.10.253	130,5	4	Sangat Baik

Tabel 4. Pengukuran Parameter *Delay* tanpa *HTB Traffic Shaping*

No	IP Address Client	Rata-Rata Delay (ms)	Indeks	Kategori
1	190.164.9.254	23,7	4	Sangat Baik
2	190.164.9.253	10,8	4	Sangat Baik
3	192.168.10.254	6,8	4	Sangat Baik
4	192.168.10.253	83,6	4	Sangat Baik

Perhitungan parameter yang ketiga yaitu *Jitter* dengan menggunakan standart *TIPHON* seperti pada Tabel 5 dibawah ini. *Jitter* adalah variasi dari *delay* yang terjadi. Lakukan perhitungan parameter *Jitter* untuk setiap *IP Address client* yang menggunakan metode *HTB Traffic Shaping* maupun yang tidak menggunakan metode. Hasil perhitungan dapat dilihat pada Tabel 5 dan Tabel 6 dibawah ini.

Tabel 5. Pengukuran Parameter *Jitter* dengan *HTB Traffic Shaping*

No	IP Address Client	Rata-Rata Jitter (ms)	Indeks	Kategori
1	190.164.9.254	10,2	3	Baik
2	190.164.9.253	1,128	3	Baik
3	192.168.10.254	7,8	4	Sangat Baik
4	192.168.10.253	5,2	4	Sangat Baik

Tabel 6. Pengukuran Parameter *Jitter* tanpa *HTB Traffic Shaping*

No	IP Address Client	Rata-Rata Jitter (ms)	Indeks	Kategori
1	190.164.9.254	23,7	3	Baik
2	190.164.9.253	10,8	2	Sedang
3	192.168.10.254	6,8	1	Buruk
4	192.168.10.253	83,6	2	Sedang

Perhitungan parameter yang keempat yaitu *Packet Loss* dengan menggunakan standart *TIPHON* seperti pada Tabel 7 dibawah ini. *Packet Loss* adalah variasi dari *packet loss* yang terjadi. Lakukan perhitungan parameter *Packet Loss* untuk setiap *IP Address client* yang menggunakan metode *HTB Traffic Shaping* maupun yang tidak menggunakan metode. Hasil perhitungan dapat dilihat pada Tabel 7 dibawah ini.

Tabel 7. Pengukuran Parameter *Packet Loss* tanpa *HTB Traffic Shaping*

No	IP Address Client	Rata-Rata Packet Loss (%)	Indeks	Kategori
1	190.164.9.254	0 %	4	Sangat Baik
2	190.164.9.253	0 %	4	Sangat Baik
3	192.168.10.254	0 %	4	Sangat Baik
4	192.168.10.253	0 %	4	Sangat Baik

Tabel 8. Indeks Hasil Pengukuran Rata-Rata

Metode	Parameter Pengukuran	Hasil Pengukuran	Standar TIPHON	Indeks Nilai
HTB Traffic Shaping	<i>Throughput</i>	64,79	Sedang	2,75
	<i>Delay</i>	13,08	Sangat Baik	3,75
	<i>Jitter</i>	13,08	Baik	3,5
	<i>Packet Loss</i>	0 %	Sangat Baik	4
Hasil rata-rata				3,5
Tanpa HTB Traffic Shaping	<i>Throughput</i>	8,79	Buruk	1
	<i>Delay</i>	31,22	Sangat Baik	4
	<i>Jitter</i>	31,22	Sedang	2
	<i>Packet Loss</i>	0 %	Sangat Baik	4
Hasil rata-rata				2,75

Berdasarkan Tabel 8, hasil rata-rata indeks menggunakan metode HTB *Traffic Shaping* yaitu 3,5 dan menurut Indeks parameter performa jaringan, pengujian yang menggunakan metode HTB *Traffic Shaping* termasuk dalam kategori “Baik”, sedangkan hasil rata-rata indeks tanpa menggunakan metode HTB *Traffic Shaping* yaitu 2,75 dan menurut Tabel 8, pengujian tanpa menggunakan metode HTB *Traffic Shaping* termasuk dalam kategori “Kurang Baik”.

4. Kesimpulan

Berdasarkan hasil analisis performa metode HTB *Traffic Shaping*, dapat diambil kesimpulan sebagai berikut. Pertama, rata-rata *Throughput* menggunakan metode HTB *Traffic Shaping* yaitu 64,79 kategori “sedang”, tanpa metode HTB *Traffic Shaping* sebesar 8,79 termasuk dalam kategori “Buruk”. Kedua, waktu kedatangan paket atau *Delay* pada metode HTB *Traffic Shaping* yaitu 13,08 ms kategori “Sangat Baik”, sedangkan tanpa menggunakan metode HTB *Traffic Shaping* yaitu 31,22 ms juga kategori “Sangat Baik”. Ketiga, jumlah variasi *Delay* yang terjadi pada saat paket dikirimkan (*Jitter*) metode HTB *Traffic Shaping* yaitu 13,08 ms kategori “Baik”, tanpa menggunakan metode HTB *Traffic Shaping* yaitu 31,22 ms kategori “Sedang”. Keempat, tidak ada paket yang hilang saat pengujian menggunakan metode HTB *Traffic Shaping* dan tanpa metode yaitu 0% kategori “Sangat Baik”. Dan yang terakhir, performa jaringan setelah menggunakan metode HTB *Traffic Shaping* memiliki rata-rata 3,5 kategori “Baik”, sedangkan performa jaringan tanpa metode HTB *Traffic Shaping* memiliki rata-rata 2,75 kategori “Kurang Baik”.

Daftar Pustaka

- Budin, S., Riadi, I. 2019. *Traffic Shaping Menggunakan Metode HTB (Hierarchical Token Bucket) pada Jaringan Nirkabel*. Buletin Ilmiah Sarjana Teknik Elektro, 1(3), 144-152.
- Fuad, Annur. 2020. *Kumpulan Ilmu Jaringan*. Yogyakarta: Annur Fuad.
- Haryanto, Edy Victor. 2012. *Jaringan Komputer*. Yogyakarta: Penerbit Andi.
- Krisianto, Andy. 2014. *Panduan cPanel Web Hosting*. Jakarta Pusat: Elex Media Komputindo.
- Lisnawita, L. 2016. *Manajemen Bandwidth Menggunakan Metode Hierarchical Token Bucket*. Digital Zone: Jurnal Teknologi Informasi Dan Komunikasi, 7(1), 18-25.
- Linkman, L., Saputro, A. M., Wicaksono, A. S., Hartomo, F. H. T., & Jatun, M. N. 2019. *Manajemen Bandwidth Menggunakan Metode Hierarchical Token Bucket (HTB) di Farid. net*. Creative Information Technology Journal, 5(3), 209-218.
- Mulyanta, Edi S. 2005. *Kupas Tuntas Telepon Seluler Anda*. Yogyakarta: Penerbit Andi.
- Mulyanta, Edi S. 2005. *Pengenalan Protokol Jaringan Wireless Komputer*. Yogyakarta: Penerbit Andi.
- Rachmadi, Tri. 2020. *Jaringan Komputer*. Surakarta: Tiga Ebook.
- Ryan, Nathan Gusti. 2018. *Basic Computer Networking*. Surabaya: XP Solution Surabaya.
- Saputro, Joko. 2010. *Praktikum CCNA di Komputer Sendiri Menggunakan GNS3*. Jakarta Selatan: mediaKita.
- SyahPutra, Andry. 2002. *Jaringan Berbasis LINUX*. Yogyakarta: Penerbit Andi.
- Syaifulah, M. 2017. *Metode Traffic Shaping pada Layer 7 Protocol untuk Mengoptimalkan Kinerja Jaringan Komputer Menggunakan Mikrotik*. Jurnal Ilmu Komputer dan Bisnis, 8(1), 1890-1898.
- Wahana Komputer. 2006. *SPP Menginstalasi Perangkat Jaringan Komputer*. Jakarta: PT. Elex Media Komputindo.
- Wijaya, A.I., Handoko, L.B. and Kom, M., 2013. *Manajemen Bandwidth Dengan Metode HTB (Hierarchical Token Bucket) Pada Sekolah Menengah Pertama Negeri 5 Semarang*. J. Tek. Inform. Univ. Dian Nuswantoro Semarang, Indonesia.
- Wulandari, Rika. 2016. *Analisis QoS (Quality of Service) pada Jaringan Internet (Studi Kasus: UPT Loka Uji Teknik Penambangan Jampang Kulon - LIPI)*. Jurnal Teknik Informatika dan Sistem Informasi, 2(2), 162-172.