

BAB II

KAJIAN PUSTAKA

2.1 Tinjauan Pustaka

Tujuan dari adanya tinjauan pustaka adalah untuk menunjang perancangan dan penelitian yang akan dilakukan. Tinjauan pustaka berasal dari karya ilmiah yang sudah diteliti sebelumnya yang berhubungan dengan penelitian dan perancangan ini. Berikut manfaat dari tinjauan pustaka:

1. Memberi gambaran metode dan teknik-teknik yang telah dilakukan untuk permasalahan yang akan dihadapi.
2. Memaparkan penelitian-penelitian serupa dengan penelitian yang akan dilakukan.
3. Mendeskripsikan sumber-sumber data atau judul pustaka yang berkaitan.

Adapun daftar tinjauan pustaka yang digunakan pada penelitian ini adalah:

1. Penelitian pertama berjudul “Rancang Bangun dan Analisis QoS (*Quality of Service*) menggunakan metode HTB (*Hierarchical Token Bucket*) Pada RT/RW Net Perumahan Prasanti 2” yang ditulis oleh [4]. Permasalahan biaya penanganan internet yang relatif mahal menjadi pokok masalah yang dihadapi oleh warga perumahan prasanti 2. Dalam menangani permasalahannya, peneliti berinisiatif membangun jaringan di perumahan prasanti 2. Selain membangun jaringan, penelitian ini juga berfokus menganalisis QoS dan implementasi algoritma HTB di jaringan yang mereka buat. Untuk mendapatkan hasil analisis QoS yang optimal, perlu diadakannya pengujian terhadap *bandwidth*. Parameter yang diuji dalam pengujian QoS ini adalah: *Delay*, *Throughput*, *Packet loss* dan *Jitter*. Skenario penelitian ini meliputi: pengujian koneksi

internet, pengujian *bandwidth* dalam keadaan ramai dan sepi, pengujian *failover* dan analisis QoS dengan jumlah *packet* 18,90, 32,76, dan 43,01 Mb. Kondisi internet ketika dalam keadaan sepi pengguna, *user* akan mendapatkan maksimal *bandwidth*, hal ini ditandai dengan warna merah pada bagian *queue tree list* atau dalam kondisi *max-limit*. Ketika kondisi internet ramai, QoS akan memajemen bandwidth dengan mengimplementasikan HTB untuk membagi rata bandwidth. Hasil dari analisis QoS sebelum dan setelah penerapan HTB menunjukkan setelah penerapan metode HTB, jaringan di perumahan prasasti 2 lebih optimal, baik dari sisi parameter *delay*, *throughput*, *packetloss*, dan *jitter*.

2. Penelitian kedua berjudul “Optimalisasi Jaringan Wi-Fi Menggunakan Metode Analisis *Quality Of Service Berdasarkan Tiphon*” yang ditulis oleh [6]. Objek penelitian ini adalah jaringan internet di Gedung D8 Universitas STIKUBANK Semarang. Tujuan dari penelitian ini adalah untuk menganalisis kualitas jaringan Wi-Fi yang terdapat di Gedung D8 Universitas STIKUBANK, sehingga dapat mengurangi penyalahgunaan gangguan sejak dini dan meningkatkan kenyamanan bagi pengguna internet di wilayah tersebut. Penelitian ini dalam standarisasi penilaian *Quality of Service (QoS)* memakai standar TIPHON (*Telecommunications and Internet Protocol Harmonization Over Network*). Hasil dari analisis QoS di Gedung D8 dengan 6 *Acces Point (AP)*: Ruang Wakil Rektor, Ruang YPPMI, Ruang TU, Ruang Sidang, Ruang Rektor, Ruang PSDM memenuhi syarat standar TIPHON dengan hasil memuaskan dan presentase diangka 91,67%.

3. Penelitian ketiga berjudul “Optimalisasi Jaringan Internet Dengan Optimalisasi *Load Balancing* Menggunakan Parameter QOS (Studi Kasus: SMK Bina Warga Lemahabang) yang ditulis oleh [7]. Penggunaan jaringan yang terus meningkat, diperlukannya dukungan jaringan internet yang stabil, tidak terkecuali di SMK Bina Warga Lemahabang. Penelitian ini menerapkan metode *load balancing* dalam menggunakan parameter QoS. Adapun tahapan penelitian menggunakan tahapan *Network Development Life Cycle (NDLC)*. Dalam penerapannya konfigurasi *load balancing* menggunakan protokol *Gateway Load Balancing (GLBP)* pada *routing EIGRP*. Hasil setelah konfigurasi *load balancing*, kualitas jaringan internet di SMK Bina Warga Lemahabang lebih kuat dan optimal. Hal ini ditandai dengan analisis QoS dengan parameter *delay* kategori sangat baik, *jitter* kategori baik, *throughput* mendapatkan kategori sangat baik dan *packet loss* sebesar 0%.
4. Penelitian keempat berjudul “Optimalisasi Jaringan *Wireless* Menggunakan *Quality of Service (QoS)* dan Algoritma *Hierarchical Token Bucket (HTB)* yang ditulis oleh [3]. Objek penelitian ini di laboratorium riset UAD. Jaringan di laboratorium riset UAD terdapat beberapa masalah seperti ketimpangan kecepatan internet antara satu dengan yang lainnya. Pengaturan *bandwidth* di laboratorium riset UAD masih menggunakan antrian FIFO. Dimana antrian FIFO memprioritaskan pengguna yang masuk terlebih dahulu. Ada 3 tahapan penelitian yang dilakukan, yaitu: perancangan arsitektur, pembangunan sistem dan pengujian. Pada tahap pengujian sistem, pengujian dilakukan sebelum dan sesudah diterapkannya *Quality of Service (QoS)* dan *Hierarchical Token Bucket*

(HTB). Hasil penelitian menunjukkan kualitas internet yang lebih baik dan pembagian *bandwidth* sesuai sasaran dapat mengurangi adanya pemborosan *bandwidth*.

- Penelitian kelima berjudul “Analisis Manajemen *Bandwidth* Menggunakan *Hierarchichal Token Bucket* Pada Router Dengan Standar Deviasi” yang ditulis oleh [8]. Penelitian ini bertujuan menganalisa dari pemerataan *bandwidth* yang terdapat pada PT ABC. Penelitian ini membandingkan perolehan deviasi (simpangan rata-rata) pada metode *Hierarchical Token Bucket (HTB)* dan metode *Per Connection Queue (PCQ)*. Hasil dari Analisa standar deviasi 2 metode pembagian *bandwidth* yaitu: metode *Hierarchical Token Bucket (HTB)* memiliki deviasi penggunaan *bandwidth* dengan nilai rata-rata paling kecil. Hasil ini menunjukkan metode *Hierarchical Token Bucket (HTB)* lebih mampu membagi *bandwidth* secara rasional dan dinamis dibanding metode *Per Connection Queue (PCQ)*.

Tabel 2. 1 Ringkasan Tinjauan Pustaka

No.	AUTHOR	JUDUL	OBJEK	METODE	HASIL
1.	Syaifudin Ahdan, Okta Firmanti, dan Suci Ramadona	Rancang Bangun dan Analisis QoS Menggunakan Metode HTB Pada RT/RW Net Perumahan Prasanti 2	Perumahan Prasanti 2	Metode Analisis <i>Quality of Service (QoS)</i> dan <i>Hierarchical Token Bucket (HTB)</i>	Hasil dari analisis QoS sebelum dan setelah penerapan HTB menunjukkan setelah penerapan metode <i>Hierarchical Token Bucket (HTB)</i> jaringan yang ada di perumahan prasasti 2 lebih optimal, baik dari sisi parameter

					<i>delay, throughput, packetloss, dan jitter.</i>
2.	Ahmad Akbar Resa dan Dewi Handayani UN	Optimalisasi Jaringan Wi-Fi Menggunakan Metode Analisis Quality of Service Berdasarkan Tiphon	Gedung D8 Universitas Stikubank (UNISBANK)	Metode Analisis <i>Quality of Service (QoS)</i>	Hasil dari analisis QoS di Gedung D8 dengan 6 <i>Access Point (AP)</i> : Ruang Wakil Rektor, Ruang YPPMI, Ruang TU, Ruang Sidang, Ruang Rektor, Ruang PSDM memenuhi syarat standar TIPHON dengan hasil memuaskan dan presentase diangka 91,67%.
3.	Sukma Prahara, Martanto, dan Irfan Ali	Optimalisasi Jaringan Internet Dengan Optimalisasi Load Balancing Menggunakan Parameter QoS	SMK Bina Warga Lembahang	Metode Analisis <i>Quality of Service (QoS)</i> dan <i>Network Development Life Cycle (NDLC)</i>	Hasil setelah konfigurasi load balancing, kualitas jaringan internet di SMK Bina Warga Lemahabang lebih kuat dan optimal. Hal ini ditandai dengan analisis QoS dengan parameter delay kategori sangat baik, jitter kategori baik, throughput mendapatkan kategori sangat baik dan packet loss sebesar 0%.
4.	M.A. Khairul Qalbi dan Imam Riadi	Optimalisasi Jaringan <i>Wireless</i> Menggunakan <i>Quality of</i>	Laboratorium riset Universitas Ahmad	Metode Analisis <i>Quality of Service</i>	Hasil pada tahap pengujian sistem, pengujian dilakukan sebelum

		<i>Service (QoS) dan Algoritma Hierarchial Token Bucket (HTB)</i>	Dahlan (UAD)	(QoS) dan <i>Hierarchial Token Bucket (HTB)</i>	dan sesudah diterapkannya <i>Quality of Service (QoS)</i> dan <i>Hierarchial Token Bucket (HTB)</i> , dimana sebelumnya memakai antrian FIFO dan kurang optimal. Hasil penelitian menunjukkan kualitas internet yang lebih baik dan pembagian <i>bandwidth</i> sesuai sasaran dapat mengurangi adanya pemborosan <i>bandwidth</i> .
5.	Pramudhita Ferdiansyah, Rini Indrayani, dan Subektiningsih	Analisis Manajemen <i>Bandwidth</i> Menggunakan <i>Hierarchichal Token Bucket</i> Pada <i>Router</i> Dengan Standar Deviasi	PT. ABC	Metode Analisis <i>Quality of Service (QoS)</i> , <i>Network Development Life Cycle (NDLC)</i> <i>Hierarchial Token Bucket (HTB)</i>	Hasil dari penelitian ini adalah membandingkan perolehan deviasi (simpangan rata-rata) pada metode <i>Hierarchial Token Bucket (HTB)</i> dan metode <i>Per Connection Queue (PCQ)</i> . Hasil dari Analisa standar deviasi adalah metode <i>Hierarchial Token Bucket (HTB)</i> memiliki deviasi penggunaan

					<i>bandwidth</i> dengan nilai rata-rata paling kecil. Hasil ini menunjukkan metode <i>Hierarchical Token Bucket (HTB)</i> lebih mampu membagi <i>bandwidth</i> secara rasional dan dinamis dibanding metode <i>Per Connection Queue (PCQ)</i> .
6.	Adrikni Alhamd	Optimalisasi Kinerja Jaringan Internet Menggunakan Analisis <i>Quality of Service (QoS)</i> dengan Metode <i>Hierarchical Token Bucket (HTB)</i>	STMIK Komputama	Metode Analisis <i>Quality of Service (QoS)</i> dan <i>Hierarchical Token Bucket (HTB)</i>	

Terdapat 5 referensi yang ditampilkan pada Tabel 2.1 yang memiliki keterkaitan dengan penelitian ini terutama pada jenis metode yang digunakan. Kelima referensi menggunakan metode analisis *Quality of Service* untuk mengevaluasi parameter jaringan seperti *delay*, *throughput*, *packet loss*, dan *jitter*. Penelitian ini dan penelitian lainnya (1 dan 4) menggunakan metode *Hierarchical Token Bucket (HTB)* untuk mengelola *bandwidth*. Ada juga di penelitian 5 adalah membandingkan performa dari metode *Hierarchical Token Bucket (HTB)* dan *Per Connection Queue (PCQ)*.

Untuk perbedaan kelima referensi dengan penelitian ini bisa dari berbagai sisi. Pertama yaitu sisi objek penelitian. Penelitian saya dan kelima referensi memiliki objek berbeda-beda. Kedua yaitu fokus spesifikasi dan hasil. Untuk penelitian ini dengan penelitian 1, 3, dan 4 fokus pada optimalisasi jaringan pada objek penelitian. Namun, untuk penelitian 2 berfokus hanya pada analisis dan untuk penelitian 5 fokus membandingkan deviasi 2 metode manajemen *bandwidth* yang berbeda. Terdapat metode tambahan untuk penelitian 3 dan 5 yaitu menggunakan metode *Network Development Life Cycle (NDLC)*.

2.2 Landasan Teori

2.2.1 Pengertian Optimalisasi

Optimasi menurut Kamus Besar Bahasa Indonesia (optimalisasi) adalah sebuah proses, pembuatan maupun cara-cara untuk mendapatkan hasil yang baru. Adapun menurut [9] analisis optimasi adalah suatu proses penguraian data awal yang telah dilakukan menggunakan metode sebelumnya. Adapun optimalisasi pada penelitian ini adalah upaya mengoptimalkan dan meningkatkan kualitas jaringan internet dari yang sudah ada sebelumnya di STMIK Komputama.

2.2.2 Definisi Jaringan Komputer

Jaringan komputer terdiri dari dua atau lebih komputer yang saling terhubung secara independen melalui media transmisi, baik itu menggunakan kabel maupun secara nirkabel. 2 perangkat komputer yang terinterkoneksi akan saling *sharing* informasi ataupun data, berbagi *resource* seperti file, *storage*, dll, setiap komputer yang terhubung pada sebuah jaringan disebut *node*. Komputer dapat dikatakan sebagai jaringan komputer ketika sebuah jaringan memiliki minimal 2

unit komputer atau lebih bahkan jutaan *node* yang saling terhubung satu sama lain [10].

2.2.3 Sejarah Jaringan Komputer

Jaringan komputer diinisiasi melalui proyek pengembangan komputer dari kelompok penelitian yang dipimpin oleh Howard Aiken seorang doktor dari Harvard University. Pada awalnya proyek ini dimanfaatkan agar dapat dipakai bersama-sama, yaitu 1 komputer untuk dipakai bersama. Pada tahun 1950, bermunculan berbagai jenis komputer, salah satunya *supercomputer* karena kebutuhan komputer untuk melayani beberapa lokasi di waktu yang sama. Sistem *Time Sharing Sistem (TSS)* ditemukan dan diterapkan pada jaringan komputer. Sistem ini memiliki beberapa terminal yang dirangkai secara seri pada komputer yang terhubung pada jaringan komputer [11].

Pada tahun 1970, terjadi permasalahan dimana harga komputer naik namun kebutuhan kerja meningkat. Konsep pemrosesan terdistribusi ditemukan untuk mengatasi permasalahan ini yaitu dengan 3 *host* melayani beberapa komputer seri yang terhubung dengan terminal [11]. Model jaringan terdistribusi merupakan model dengan beberapa *host* komputer yang mengerjakan pekerjaan besar yang dilakukan secara seri untuk melayani kebutuhan terminal-terminal yang tersambung secara paralel di setiap *host* komputer.

Pada tahun 1982 dibentuk suatu protokol resmi yaitu *Transmission Control Protocol/Internet Protocol (TCP/IP)*. 2 tahun setelahnya yaitu pada tahun 1984 diperkenalkan suatu sistem penamaan masing-masing pada unit komputer yang saling terhubung pada jaringan yaitu *Domain Name System (DNS)*.

2.2.4 Jenis-Jenis Jaringan Komputer

2.2.4.1 Berdasarkan Jangkauan Geografis

1) Jaringan Local Area Network (LAN)

Jaringan LAN merupakan sebuah jaringan komputer yang jangkauan wilayahnya kecil seperti jaringan komputer di lembaga pendidikan seperti kampus, sekolah, jaringan komputer di perusahaan dengan skala kecil. Kecepatan transfer sebuah LAN yang berbasis pada teknologi IEEE 802.3 Ethernet dan memanfaatkan perangkat switch adalah 10, 100 atau 100 Mbps. Selain Ethernet, saat ini teknologi 802.11b (*wifi*) sering juga digunakan untuk menciptakan jaringan LAN. LAN memberi kesempatan setiap komputer atau node untuk melakukan komputasi sendiri [12].

2) Jaringan Metropolitan Area Network (MAN)

Jaringan MAN merupakan jaringan yang cangkupannya lebih luas dari LAN dengan kecepatan transfer tinggi. MAN merupakan jaringan yang menghubungkan jaringan LAN satu dengan yang lainnya, MAN biasanya terletak pada suatu kampus atau wilayah yang agak luas antar 10 hingga 50 km [12].

3) Jaringan Wide Area Network (WAN)

WAN adalah jaringan komputer dengan jangkauan area yang besar. contoh penggunaan WAN adalah jaringan komputer antar kota, wilayah maupun negara. Karakteristik dari WAN adalah jaringan ini membutuhkan sebuah router dan saluran komunikasi publik. Setiap area lokal jaringan dapat bertukar informasi dengan pengguna komputer di area lainnya [12].

2.2.4.2 Berdasarkan Peranan Dalam Memproses Data

1) Jaringan *Peer to Peer*

Jaringan *peer to peer* dalam melakukan komunikasi tidak menggunakan komputer *server* sebagai *server*, setiap jaringan dapat mengatur jaringannya sendiri. Jaringan ini biasanya dimanfaatkan untuk pertukaran data secara langsung dan yang paling banyak dimanfaatkan dirumah. Jaringan ini memposisikan semua komputer pada posisi yang sama [11].

2) Jaringan *Client-Server*

Jaringan *client-server* adalah jaringan yang melibatkan dua atau lebih komputer, di mana satu komputer berfungsi sebagai *server* dan komputer lainnya sebagai *client*. *Server* bertanggung jawab untuk mengelola jaringan tersebut. *Server* utama digunakan untuk menyimpan data, yang kemudian dapat diakses oleh *client* yang terhubung dalam jaringan tersebut. Jaringan *client server* biasanya digunakan pada suatu bisnis [11].

2.2.4.3 Berdasarkan Media Transmisi

1) Jaringan *Wired*

Jaringan *wired* merupakan jaringan yang menggunakan kabel fisik untuk menghubungkan satu perangkat dengan perangkat lainnya agar terhubung. Contohnya komputer, *printer*, maupun *router*. Jaringan *wired* memiliki tingkat keamanan yang tinggi, karena untuk terhubung antar perangkat harus terhubung fisik melalui kabel, akses yang tidak sah akan lebih sulit dilakukan untuk jenis jaringan ini. Oleh karena itu, jaringan *wired* lebih cocok untuk lingkungan jaringan yang memerlukan perlindungan ketat.

Selain disisi keamanan, jaringan *wired* menawarkan keandalan yang lebih baik. Karena jaringan dihubungkan menggunakan kabel, sehingga untuk

kecepatan transfer *download* maupun *upload* tidak terinterferensi oleh perangkat lain.

2) Jaringan *Wireless*

Jaringan *wireless* adalah jaringan dengan teknologi terbaru yang digunakan sebagai opsi alternatif penggunaan teknologi kabel yang tidak memungkinkan di wilayah tertentu. Ada beberapa metode pengiriman data yang digunakan untuk menggantikan jaringan kabel yaitu gelombang radio, sinar inframerah, *bluetooth*, gelombang mikro dan cahaya. Meskipun berbeda frekuensi, semua teknologi pengiriman data *wireless* memanfaatkan prinsip gelombang, hal ini menyebabkan variasi dalam kecepatan dan jangkauan pengiriman [13]. Dilihat dari efisiensi tempat, biaya dan kemudahan instalasi, jaringan *wireless* lebih diunggulkan dibanding jaringan kabel.

IEEE 802.11 adalah serangkaian spesifikasi untuk *Medium Access Control (MAC)* dan *Physical Layer (PHY)* yang digunakan sebagai standar protokol jaringan nirkabel pada frekuensi 2.4, 3.6, 5, dan 60 GHz. Standar ini dikelola oleh *Institute of Electrical and Electronics Engineers (IEEE)*. dan mencapai tonggak teknis pada tahun 1997 [14].

2.2.5 TCP/IP

Transmission Control Protocol/Internet Protocol (TCP/IP) adalah standar komunikasi data yang digunakan oleh komunitas internet untuk bertukar informasi antara komputer di dalam jaringan internet. Protokol ini dikembangkan pada akhir 1970-an hingga awal 1980-an dengan tujuan menjadi standar untuk menghubungkan dan membangun jaringan area luas (WAN) antara komputer dan

jaringan. Keistimewaan TCP/IP sifatnya yang terbuka dan tidak bergantung pada mekanisme transportasi fisik, sehingga dapat digunakan di berbagai lingkungan [15].

Dalam perkembangannya, TCP/IP muncul sebelum model OSI. Model TCP/IP terdiri dari empat lapisan: *network access layer*, *internet layer*, *transport layer*, dan *application layer*. TCP/IP menggabungkan tiga lapisan teratas dari model OSI ke dalam satu lapisan aplikasi. Protokol TCP/IP mendefinisikan dua protokol utama, yaitu *Transmission Control Protocol (TCP)* dan *User Datagram Protocol (UDP)*, sedangkan lapisan ketiga didefinisikan sebagai *Internetworking Protocol (IP)*, dengan beberapa protokol pendukung lainnya [15].

Internet Protocol (IP), sebagai bagian dari TCP/IP, digunakan untuk mengalamatkan dan merutekan data antar *host* dalam jaringan berbasis TCP/IP. Versi yang paling umum digunakan saat ini adalah IPv4, namun di masa mendatang akan digantikan oleh IPv6. IP berperan sebagai protokol utama dalam hierarki TCP/IP, mengirimkan data dari satu titik ke titik lainnya tanpa perlu memelihara koneksi sesi. Meskipun tidak menjamin pengiriman data sampai ke tujuan, tanggung jawab itu diserahkan kepada protokol pada lapisan yang lebih tinggi, seperti TCP [15].

2.2.6 Mikrotik

Mikrotik merupakan perangkat jaringan berupa perangkat keras dan perangkat lunak yang digunakan sebagai *router*. Perangkat keras yang digunakan untuk *router* bisa berupa *router PC* (terinstalasi pada PC) maupun *router board*. Sedangkan perangkat lunak dikenal dengan Mikrotik *Router OS* [16]. Mikrotik

Router OS merupakan sistem operasi dan platform perangkat lunak yang dapat memfungsikan sebuah komputer menjadi sebuah router. Fitur-fitur pada Mikrotik *Router OS* mencakup fitur untuk *ip network* dan jaringan *wireless*. Penggunaan Mikrotik *Router OS* saat ini banyak dimanfaatkan oleh penyedia layanan internet (ISP) dan *provider hotspot* [17].

Instalasi *MikroTik Router OS* dapat dilakukan pada *personal computer* (PC) yang tidak memiliki *resource* yang cukup besar. Meskipun komputer yang digunakan untuk instalasi *MikroTik Router OS* tidak memerlukan *resource* yang cukup besar namun ketika *MikroTik Router OS* digunakan sebagai *gateway* dan *routing* yang rumit, disarankan memilih *resource PC* yang memadai untuk beban kerja yang lebih besar [18]. Mikrotik akan memantau koneksi didalam suatu jaringan.

2.2.7 *Quality of Service (QoS)*

Quality of Service (QoS) merupakan standar untuk menilai kualitas koneksi dalam sebuah jaringan dan memastikan proses pengiriman data berjalan lancar dengan kualitas koneksi yang optimal. Parameter-parameter QoS didapat dari penilaian dari sebuah *traffic* jaringan yang diterima disisi klien maupun *server*. Kualitas trafik jaringan dianggap baik apabila parameter QoS dapat dikendalikan dan terkelola dengan baik [19].

Masalah umum yang kerap muncul dalam jaringan internet meliputi ketersediaan *bandwidth*, *packet loss* (kehilangan data), *delay/latency* (keterlamabatan) dan *jitter (delay variance)*. Pada jaringan *layer 2* dan *layer 3* biasanya QoS diaktifkan. Adapun proses pemetaan pada *layer 2* ke *layer 3*

digunakan untuk memastikan kualitas komunikasi end-to-end [20]. Berikut Tabel 2.2 merupakan indeks parameter QoS berdasarkan [21].

Tabel 2. 2 Indeks Parameter *Quality of Service (QoS)*

Nilai	Presentase (%)	Indeks
3,8 – 4	95 – 100	Sangat Memuaskan
3 – 3,79	75 – 94,75	Memuaskan
2 – 2,99	50 – 74,75	Kurang Memuaskan
1 – 1,99	25 – 49,75	Jelek

Berikut merupakan parameter-parameter dari *Quality of Service (QoS)*:

1. *Throughput*

Throughput adalah kecepatan efektif transfer data yang diukur dalam *bit per second (bps)*. *Throughput* mengacu pada total jumlah paket yang berhasil mencapai tujuan selama jangka waktu tertentu, yang kemudian dibagi dengan durasi waktu tersebut [22]. Berikut Tabel 2.3 merupakan standar kategori *Throughput* berdasarkan [21].

Tabel 2. 3 Kategori *Throughput*

Kategori <i>Throughput</i>	<i>Throughput</i> (bps)	Indeks
Sangat Bagus	10	4
Bagus	75	3
Sedang	50	2

Jelek	<25	1
-------	-----	---

Persamaan perhitungan *Throughput*:

$$Throughput = \frac{\text{Paket data diterima}}{\text{Total waktu pengiriman data}} \quad (2.1)$$

2. *Packet Loss*

Packet Loss merupakan parameter yang mengacu pada kondisi yang menunjukkan jumlah paket data hilang yang terjadi karena *tabrakan (collision)* atau kepadatan lalu lintas (*congestion*) di dalam suatu jaringan [22]. Berikut Tabel 2.4 merupakan standar kategori *Packet Loss* berdasarkan [21].

Tabel 2. 4 Kategori *Packet Loss*

Kategori <i>Packet Loss</i>	<i>Packet Loss</i> (%)	Indeks
Sangat Bagus	0	4
Bagus	3	3
Sedang	15	2
Jelek	25	1

Persamaan perhitungan *Packet Loss*:

$$Packet Loss = \frac{\text{Paket data dikirim} - \text{paket data diterima}}{\text{Paket data dikirim}} \times 100 \quad (2.2)$$

3. *Delay (Latency)*

Delay merupakan parameter yang didapat dari waktu yang diperlukan data untuk mencapai tujuannya. *Delay* dapat dipengaruhi oleh faktor-faktor seperti jarak, media fisik, *congestion* (kemacetan), dan lama waktu yang dibutuhkan dalam

proses [22]. Berikut Tabel 2.5 merupakan standar kategori parameter *Latency* berdasarkan [21].

Tabel 2. 5 Kategori *Delay*

Kategori Delay	Besar delay (ms)	Indeks
Sangat Bagus	<150	4
Bagus	150 s/d 300	3
Sedang	300 s/d 450	2
Jelek	>450	1

Persamaan perhitungan *Delay (Latency)*:

$$\text{Delay} = \frac{\text{Paket length}}{\text{Link bandwidth}} \quad (2.3)$$

4. *Jitter* (Variasi Kedatangan Paket)

Jitter merupakan variasi dalam waktu kedatangan data yang disebabkan oleh fluktuasi dalam panjang antrian, waktu pemrosesan data, waktu pemrosesan data, dan proses penghimpunan ulang paket di akhir perjalanan. *Jitter* sering disebut sebagai variasi *delay* yang mengindikasikan seberapa besar variasi *delay* yang terjadi dalam transmisi data didalam jaringan [22]. Berikut Tabel 2.6 merupakan standar kategori parameter *Jitter* berdasarkan [21].

Tabel 2. 6 Kategori *Jitter*

Kategori Jitter	Jitter (ms)	Indeks
Sangat Bagus	0	4

Bagus	0 s/d 75	3
Sedang	75 s/d 125	2
Jelek	125 s/d 225	1

Persamaan perhitungan *Jitter*:

$$Jitter = \frac{\text{Total variasi delay}}{\text{Total paket yang diterima}} \quad (2.4)$$

$$\text{Total variasi delay} = \text{delay} - (\text{rata-rata delay}) \quad (2.5)$$

2.2.8 Bandwidth

Menurut Muh Sahal, S.Kom dalam bukunya yang berjudul *Administrasi Infrastruktur Jaringan*, *bandwidth* merupakan luas atau lebar pita frekuensi yang digunakan oleh sinyal dalam media transmisi. Dalam konteks jaringan komputer *bandwidth* dihitung dalam satuan *bits per second (bps)*. *Bandwidth* juga sering digunakan sebagai sinonim untuk data *transfer rate*, yang merupakan jumlah maksimum data yang dapat dibawa dari satu titik ke titik lain dalam jangka waktu tertentu. Fungsi *bandwidth* dalam penerapannya adalah untuk menghitung transaksi data, yaitu tidak hanya lebar jalur data, namun seberapa banyak data yang dilewatan dalam suatu media transmisi jaringan [19]. Dengan demikian *bandwidth* merupakan kecepatan maksimal yang mungkin terjadi dalam proses transfer data.

2.2.9 Manajemen Bandwidth

Manajemen *bandwidth* merupakan proses pengukuran dan pengontrolan komunikasi suatu jaringan guna mengatur bandwidth agar sesuai

kebutuhan. Manajemen *bandwidth* mengalokasikan prioritas jenis aliran data berdasarkan seberapa penting aliran data tersebut. Manajemen *bandwidth* diperlukan untuk membagi kapasitas yang ada dalam jaringan secara cepat dan tepat untuk setiap *client* dan *server*, karena besarnya *bandwidth* akan berdampak pada kecepatan transmisi data [19]. Dalam penerapannya layanan *Quality of Service* digunakan untuk menetapkan tipe-tipe suatu *traffic* jaringan dalam mengatur manajemen *bandwidth* suatu jaringan. *Bandwidth* dapat dikategorikan 2 macam sebagai berikut :

1. *Bandwidth* Digital

Bandwidth digital mengacu pada kapasitas volume data yang dikirimkan melalui saluran komunikasi dalam waktu tertentu, biasanya diukur dalam satuan *bit per second (bps)*.

2. *Bandwidth* Analog

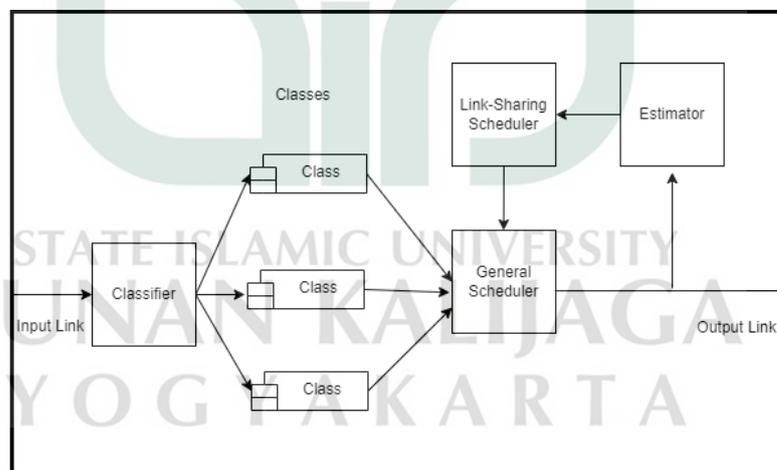
Bandwidth Analog merujuk pada selisih antara frekuensi terendah dan tertinggi dalam suatu rentang frekuensi tertentu, yang diukur dalam *hertz (Hz)*. Rentang frekuensi ini menentukan seberapa banyak informasi yang dapat ditransmisikan dalam satuan waktu.

2.2.10 Hierarchical Token Bucket

HTB adalah metode yang digunakan untuk mengatur distribusi *bandwidth* secara hierarkis, membaginya ke dalam berbagai kelas untuk mempermudah pengelolaan bandwidth. HTB diklaim menyediakan kemudahan penggunaan melalui teknik peminjaman dan pembagian trafik yang lebih tepat. Teknik antrian HTB memungkinkan pembatasan *traffic* pada setiap level dan

klasifikasi [23]. *Hierarchical Token Bucket (HTB)* adalah metode penjadwalan paket yang sering digunakan pada router berbasis linux, yang pertama kali dikembangkan oleh Martin Devera [22].

Cara kerja *Hierarchical Token Bucket (HTB)* sebenarnya tidak jauh berbeda dengan pendahulunya, yaitu *Class Based Queueing (CBQ)*. Perbedaan utamanya terletak pada penggunaan mekanisme *Deficit Round Robin (DRR)* dalam *General Scheduler* HTB. Selain itu, pada bagian blok umpan balik, HTB menggunakan *Token Bucket Filtering* sebagai estimatornya, menggantikan *Exponential Weighted Moving Average (EWMA)* yang digunakan pada CBQ. Perubahan ini membuat HTB lebih efisien dalam mengatur dan memprioritaskan lalu lintas jaringan dibandingkan dengan pendahulunya [24]. Berikut adalah ilustrasi cara kerja HTB yang dapat dilihat pada Gambar 2.1.



Gambar 2. 1 Cara Kerja HTB

1. *General Scheduler* HTB

Metode *Hierarchical Token Bucket (HTB)*, traffic jaringan dibagi menjadi beberapa aliran melalui struktur kelas yang hierarkis. Algoritma penjadwalan paket bekerja dengan memilih kelas terendah (*leaf class*) yang belum

mencapai batas *bandwidth*, kemudian mengirimkan paket mulai dari kelas dengan prioritas tertinggi hingga terendah. Jika semua kelas telah melebihi batas yang ditetapkan, sistem akan mencoba menemukan *leaf class* yang dapat meminjam *bandwidth* dari kelas induknya (*parent class*). Apabila peminjaman dari *parent class* tidak memungkinkan, proses dilanjutkan dengan mencoba meminjam *bandwidth* dari kelas di tingkat lebih tinggi, seperti *grand father class*.

2. *Deficit Round Robin (DRR)*

Deficit Round Robin (DRR) adalah skema penjadwalan yang merupakan modifikasi dari *Weighted Round Robin (WRR)*, memungkinkan setiap kelas memiliki panjang paket yang berbeda. DRR bekerja dengan memberikan hak kepada suatu kelas untuk mengirim paket dalam satu putaran jika ukuran paket tersebut sama dengan atau lebih kecil dari ambang batas yang disebut *Quantum*. *Quantum* ini berfungsi sebagai indikator layanan untuk menghitung status *deficit counter*. Jika paket pertama (*Head of Line/HOL*) memenuhi syarat, paket tersebut dikirim, dan *deficit counter* dikurangi sebesar ukuran paket tersebut. Jika tidak, paket HOL akan ditahan dan *deficit counter* tetap sama dengan nilai *Quantum* untuk putaran itu. Apabila tidak ada paket dalam antrian, *deficit counter* direset ke nilai 0.

3. Estimator

Hierarchical Token Bucket (HTB) memanfaatkan *Token Bucket Filter (TBF)* sebagai alat estimasi untuk menentukan status sebuah kelas dalam jaringan, apakah kelas tersebut berada dalam kondisi *underlimit*, *at limit*, atau *overlimit*. TBF beroperasi berdasarkan prinsip algoritma ember token, di mana setiap paket yang akan dikirim memerlukan sejumlah token sebagai izin untuk dikirimkan. Jika token

tidak tersedia atau jumlahnya tidak mencukupi, paket-paket tersebut harus menunggu sampai cukup token terkumpul untuk memungkinkan pengiriman. Dengan kata lain, TBF mengontrol laju pengiriman data dengan cara memastikan bahwa hanya paket-paket yang memiliki token yang dapat dikirimkan segera. Jika kelas dalam keadaan *underlimit*, maka pengiriman paket berjalan lancar tanpa hambatan. Namun, jika kelas berada pada kondisi *at limit* atau *overlimit*, proses pengiriman paket akan tertunda sampai jumlah token yang diperlukan kembali tersedia. Hal ini memungkinkan pengelolaan *bandwidth* yang lebih efektif, menghindari kemacetan jaringan, dan memastikan bahwa pengiriman data dilakukan sesuai dengan alokasi *bandwidth* yang telah ditetapkan.

2.2.11 Network Address Translation

Network Address Translation (NAT) menurut Iwan Sofana dalam buku berjudul CISCO CCNA & Jaringan Komputer adalah proses pengalihan alamat IP dari satu alamat ke alamat lain. Ketika suatu paket dialihkan menggunakan NAT pada suatu *link*, *link* tersebut akan menyimpan informasi asal paket tersebut sehingga komunikasi dapat berlangsung seperti biasa saat paket tersebut kembali dari tujuan. Tujuan utama NAT adalah untuk mengurangi jumlah alamat *IP publik* yang diperlukan oleh suatu organisasi atau perusahaan, baik untuk alasan ekonomi maupun keamanan. NAT memungkinkan jaringan dengan alamat *IP publik* atau *IP privat* yang belum terdaftar di internet untuk tetap terhubung dan mengakses internet [25]. NAT bekerja dengan mengalihkan paket data dari satu alamat IP ke alamat IP yang berbeda. Selama proses ini, NAT menyimpan informasi tentang sumber dan tujuan paket. Ketika paket tersebut kembali, NAT akan

mengirimkannya ke alamat yang semula, memastikan bahwa host hanya menerima paket yang diminta atau dikirim, sehingga komunikasi berjalan dengan lancar [26].



BAB III

METODE PENGEMBANGAN SISTEM

3.1 Objek Penelitian

Penelitian ini dilakukan di STMIK Komputama Cilacap yang beralamatkan di Jalan Raya Majenang Cimanggu KM 8 No. 99. Kab. Cilacap, Provinsi Jawa Tengah. Penelitian ini menganalisis *Quality of Service* (QoS) menggunakan metode algoritma *Hierarchical Token Bucket* (HTB) pada *network system* di STMIK Komputama Cilacap, *Quality of Service* (QoS) menjadi semakin penting dalam konteks aplikasi jaringan yang membutuhkan aliran data yang andal, responsif dan memiliki prioritas yang tepat untuk berbagai jenis layanan. Implementasi metode *Hierarchical Token Bucket* (HTB) menawarkan pendekatan hierarkis untuk mengalokasikan *bandwidth* sesuai kebutuhan dan prioritas layanan. Hal ini dapat mempermudah pengelola jaringan untuk mengalokasikan *bandwidth* sesuai dengan kebutuhan pengguna.

3.2 Metode Pengumpulan Data

Metode pengumpulan data merupakan metode yang digunakan untuk mengumpulkan informasi yang diperlukan guna menguji hipotesis. Metode ini sangat penting karena data yang dikumpulkan akan menjadi dasar analisis perancangan sebuah penelitian.

3.2.1 Studi Literatur

Studi literatur merupakan teknik atau cara yang digunakan untuk mengumpulkan, mempelajari serta menganalisis informasi dari berbagai sumber yang relevan untuk memberikan pemahaman yang komprehensif. Studi literatur

dilakukan oleh penulis melalui berbagai sumber seperti jurnal, buku, dan bahan bacaan yang berkaitan dengan tema.

3.2.2 Observasi

Dalam melakukan implementasi penelitian, peneliti terlebih dahulu melakukan observasi pada jaringan di STMIK Komputama, Observasi dilakukan untuk mengetahui pendataan alat yang dibutuhkan, kondisi topologi jaringan dan wawancara mengenai penelitian di objek penelitian. Untuk membuat mekanisme pengelolaan bandwidth, melibatkan tahapan wawancara. Wawancara dilakukan dengan dosen serta pengelola jaringan di STMIK Komputama. Berikut ditampilkan pada Tabel 3.1 data responden wawancara untuk penelitian ini.

Tabel 3. 1 Responden Wawancara

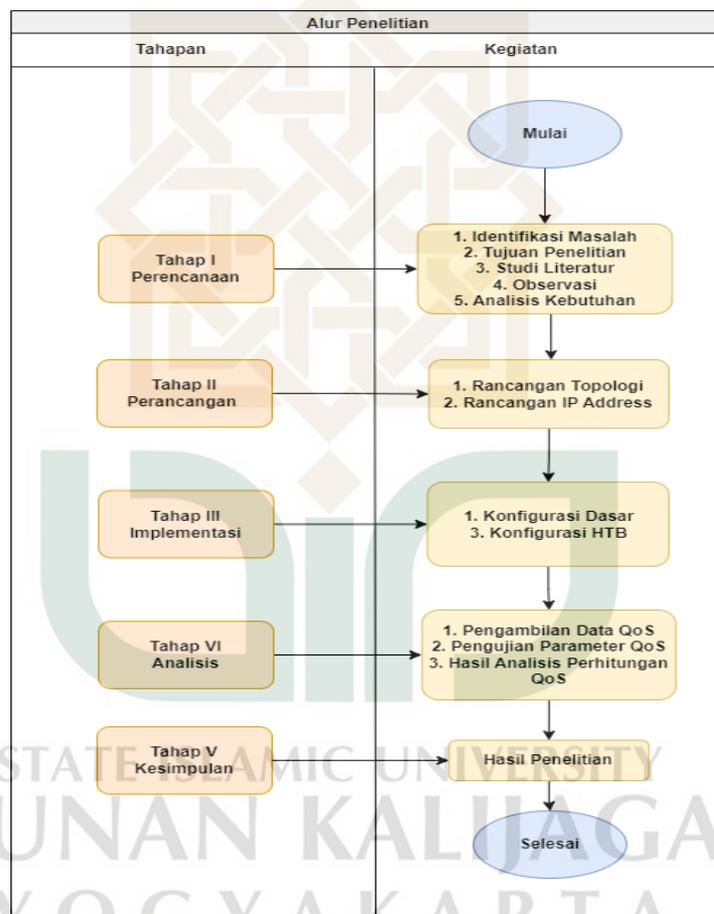
No	Nama	Pekerjaan
1	Slamet Cahyo Egy Sahputro, M.Kom	Dosen dan Pengelola Jaringan
2	Beny Riswanto, M.Kom	Dosen dan Pengelola Jaringan

3.2.3 Metode *Hierarchical Token Bucket (HTB)*

Implementasi HTB digunakan untuk mengalokasikan kebutuhan *bandwidth* pada jaringan *dedicated* di STMIK Komputama. Metode HTB menggunakan *queue tree*. *Queue Tree* merupakan batasan data untuk alamat IP tertentu, subnet, protokol, dan parameter lainnya [27]. *Queue Tree* memiliki *parent*

dan *child queue*. *Parent queue* akan membantu router dalam mengalokasikan *bandwidth*. Sedangkan mengatur limitasi pada setiap *client* [8].

Dalam implementasi HTB peneliti membuat diagram *flowchart* yang digunakan untuk mempermudah alur penelitian HTB yang akan dilakukan. Berikut pada Gambar 3.1 ditampilkan *flowchart* dari alur penelitian ini.



Gambar 3.1 Alur Penelitian

Berikut penjelasan dari Tabel 3.2 mengenai Alur Penelitian:

1. Tahap I Perencanaan

Tahapan ini meliputi identifikasi masalah yang mengerucut terhadap tujuan penelitian, kemudian melakukan studi literatur terkait pengelolaan *bandwidth* dan parameter QoS. Studi literatur diperoleh dari buku, jurnal, skripsi

yang terkait dengan tema penelitian. Setelah melakukan studi literatur, peneliti melakukan observasi langsung terhadap pihak terkait untuk kebutuhan rancangan analisis kebutuhan. Analisis kebutuhan berupa *hardware* dan *software* yang dibutuhkan selama penelitian.

2. Tahap II Perancangan

Tahap perancangan sangat penting guna memenuhi tahapan selanjutnya, yaitu tahap implementasi. Pada tahap perancangan terdiri dari 2 tahap yaitu rancangan topologi dan rancangan IP Address (pemberian *IP Address* pada *DHCP Client*).

3. Tahap III Implementasi

Tahap implementasi merupakan tahap penerapan konfigurasi pada mikrotik dan penerapan metode HTB untuk mengukur parameter *Quality of Service (QoS)* pada tahap analisis.

4. Tahap IV Analisis

Setelah pengambilan data tahap implementasi, analisis pertama yaitu mengukur performa jaringan atau menghitung parameter QoS yang terdiri dari *delay*, *jitter*, *throughput*, dan *packet loss*. Dalam melakukan analisis QoS terdapat beberapa tahap untuk memastikan bahwa parameter QoS sesuai dengan standar *TIPHON (Telecommunications and Internet Protocol Harmonization)*. Hasil rata-rata nilai parameter QoS akan dibandingkan dengan indeks nilai QoS yang harus dicapai.

5. Tahap V Kesimpulan

Tahap ini merupakan tahap dokumentasi dan pengambilan kesimpulan dan saran. Kesimpulan merupakan jawaban dari rumusan masalah pada penelitian.

Adapun saran merupakan harapan penulis untuk penelitian selanjutnya dan kekurangan-kekurangan yang terdapat pada penelitian.

3.3 Alat dan Bahan Penelitian

Pada penelitian terdapat alat dan bahan yang digunakan untuk mengoptimalkan suatu penelitian. Berikut merupakan alat dan bahan yang digunakan dalam penelitian ini.

3.3.1 Alat Penelitian

Alat yang dibutuhkan dalam penelitian disajikan pada Tabel 3.3 untuk perangkat kerasnya dan juga spesifikasinya.

Tabel 3. 2 Kebutuhan Alat Penelitian

No	Perangkat Keras	Spesifikasi
1	<i>Computer Client</i>	- Sistem Operasi: Windows 10 - Processor: Intel® Core™ i5
2	<i>Mikrotik Routerboard</i>	Mikrotik RB951U1-2nD
3	Kabel	Kabel Ethernet RJ 45 Kabel UTP

3.3.2 Bahan Penelitian

Bahan yang dibutuhkan dalam penelitian ini disajikan pada Tabel 3.4 untuk perangkat lunak dan fungsi dari perangkat lunak yang digunakan.

Tabel 3. 3 Kebutuhan Bahan Penelitian

No	Perangkat Lunak	Fungsi
----	-----------------	--------

1	<i>Mikrotik Router OS</i>	Sistem operasi yang digunakan untuk mikrotik
2	Windows 10	Sistem operasi pada semua komputer <i>client</i>
3	<i>Wireshark</i>	Alat pengukur untuk parameter <i>Quality of Service</i>
4	<i>Winbox</i>	Aplikasi yang digunakan untuk meremote mikrotik

BAB IV

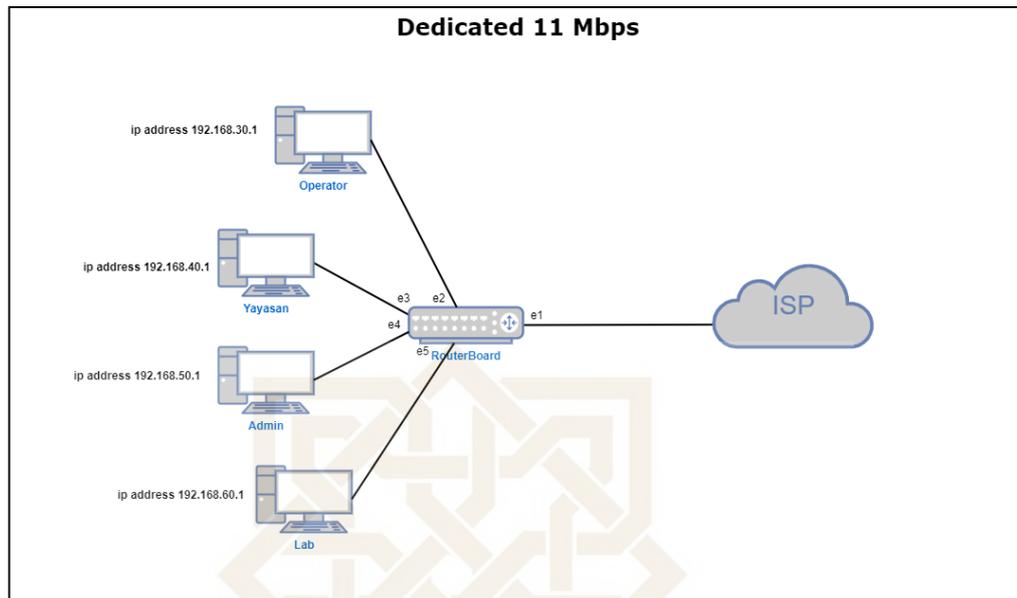
HASIL DAN PEMBAHASAN

4.1 Konfigurasi Awal Jaringan

Tahap konfigurasi awal jaringan mendukung kebutuhan konektivitas di STMIK Komputama Majenang yang memiliki berbagai ruangan yaitu ruang yayasan, ruang operator, ruang laboratorium, dan lain-lain. Konfigurasi jaringan ini bertujuan untuk mengoptimalkan penggunaan *bandwidth* secara rasional dan meningkatkan efisiensi dan keamanan jaringan melalui sistem manajemen QoS menggunakan metode *Hierarchical Token Bucket (HTB)*.

4.1.1 Desain Topologi Jaringan

Desain jaringan yang diterapkan di STMIK Komputama terdapat 2 jenis koneksi utama, yaitu koneksi *broadband* dan *dedicated*. Koneksi *Broadband* memiliki kecepatan 150 Mbps yang didistribusikan melalui *access point*. Pengaturan koneksi *broadband* dilakukan oleh penyedia layanan (ISP). Sedangkan untuk mendukung perangkat tertentu, penyedia layanan menyediakan kecepatan 11 Mbps yang dialokasikan untuk perangkat tertentu melalui koneksi kabel. Desain topologi jaringan dirancang untuk memastikan setiap perangkat yang memerlukan koneksi *dedicated* mendapatkan alokasi *bandwidth* yang tepat sesuai kebutuhan terkhusus untuk pengguna prioritas. Adapun untuk desain topologi jaringan *dedicated* di STMIK Komputama dapat dilihat pada Gambar 4.1.



Gambar 4. 1 Desain Topologi Jaringan

Gambar 4.1 menggambarkan tentang kondisi topologi untuk jaringan *dedicated* di STMIK Komputama. *Routerboard* akan mengatur *traffic* data antar *client*. *Port ethernet* 1 disambungkan ke internet yang sudah disediakan oleh penyedia layanan (ISP) dengan kecepatan 11 Mbps. Kemudian untuk *port ethernet* 2 akan disambungkan ke Operator, *port ethernet* 3 ke Yayasan, *port ethernet* 4 ke Admin dan *port operator* 5 ke operator Laboratorium.

4.1.2 Desain Komponen Jaringan

Perangkat yang diberikan *bandwidth dedicated* akan dihubungkan langsung ke *router* mikrotik melalui kabel yang akan disambungkan ke *port ethernet* pada *router*. Desain komponen jaringan meliputi *IP Address* dan *subnet* untuk tiap *client*, seperti yang disajikan pada Table 4.1

Tabel 4. 1 Konfigurasi Jaringan pada Port Ethernet

Port Ethernet	Client	IP Address	Subnet
2	Operator	192.168.30.1/24	192.168.30.0/24
3	Yayasan	192.168.40.1/24	192.168.40.0/24
4	Admin	192.168.50.1/24	192.168.50.0/24
5	Laboratorium	192.168.60.1/24	192.168.60.0/24

Setiap *client* memiliki pengaturan yang spesifik sesuai dengan fungsinya untuk memastikan alokasi *bandwidth* yang optimal dan aman. Konfigurasi *IP Address* akan digunakan sebagai alamat jaringan utama untuk melakukan komunikasi data antara *router* dan *client*. Sedangkan konfigurasi *subnet* penting untuk memastikan bahwa *client* dapat terhubung dan berkomunikasi dengan baik pada jaringan yang sudah dirancang

Pada konfigurasi *client* di ruang Operator, *port Ethernet* 2 digunakan dengan *IP Address* 192.168.30.1/24 dan *subnet* 192.168.30.0/24. Sementara itu, ruang Yayasan menggunakan *port Ethernet* 3 dengan *IP Address* 192.168.40.1/24 dan *subnet* 192.168.40.0/24. Untuk ruang Admin, *port Ethernet* 4 dengan *IP Address* 192.168.50.1/24 dan *subnet* 192.168.50.0/24, sedangkan Laboratorium terhubung melalui *port Ethernet* 5 dengan *IP Address* 192.168.60.1/24 dan *subnet* 192.168.60.0/24. Setiap *client* memiliki pengaturan yang spesifik sesuai dengan fungsinya untuk memastikan alokasi *bandwidth* yang optimal dan aman.

4.2 Alokasi *Bandwidth*

Dedicated bandwidth akan memastikan bahwa perangkat memiliki jalur koneksi yang stabil tanpa terganggu oleh *traffic* jaringan umum. Berikut merupakan pembagian *bandwidth* per *client* untuk jaringan *dedicated* 11 Mbps :

1. Operator menggunakan 1 Mbps untuk mendukung aktivitas administratif seperti pengelolaan sistem dan data.
2. Yayasan menggunakan 2 Mbps untuk mendukung aktivitas administratif yayasan dan komunikasi internal yang penting.
3. Admin menggunakan 5 Mbps untuk mendukung kebutuhan dalam administrasi seperti pengelolaan data mahasiswa, dosen, dan kegiatan di kampus.
4. Laboratorium menggunakan 3 Mbps, untuk keperluan admin yang berada di laboratorium kampus.

4.3 Implementasi

Pada tahapan implementasi terdapat 2 tahap konfigurasi, yaitu konfigurasi dasar dan konfigurasi HTB. Berikut merupakan tahapan dari kedua konfigurasi yang diterapkan:

4.3.1 Konfigurasi Dasar

1. Mengubah Nama *Interface*

Tahap awal konfigurasi dasar yaitu mengubah nama *interface* pada *router* mikrotik. Setiap *interface*, *ethernet* diberi nama sesuai dengan fungsinya dalam jaringan untuk mempermudah manajemen dan pemeliharaan jaringan. Misalnya ether 2 diubah menjadi ether2_operator untuk jaringan operator. ether 3 diubah menjadi Ether3_yayasan untuk jaringan pada operator yayasan. ether 4 diubah menjadi Ether4_administrasi untuk jaringan pada operator admin. ether 5 diubah menjadi ether5_lab_computer untuk jaringan operator di laboratorium komputer. Gambar 4.2 menunjukkan tampilan *interface list* yang sudah diubah pada mikrotik.

	Name	Type	Actual MTU	L2 MTU	Tx
R	Ether3_yayasan	Ethernet	1500	1598	
R	Ether4_admini...	Ethernet	1500	1598	
R	ether1	Ethernet	1500	1598	
R	ether2_operator	Ethernet	1500	1598	
R	ether5_lab_co...	Ethernet	1500	1598	

Gambar 4. 2 Tampilan *Interface List*

2. Konfigurasi Alamat *IP Address* Untuk Setiap *Interface*

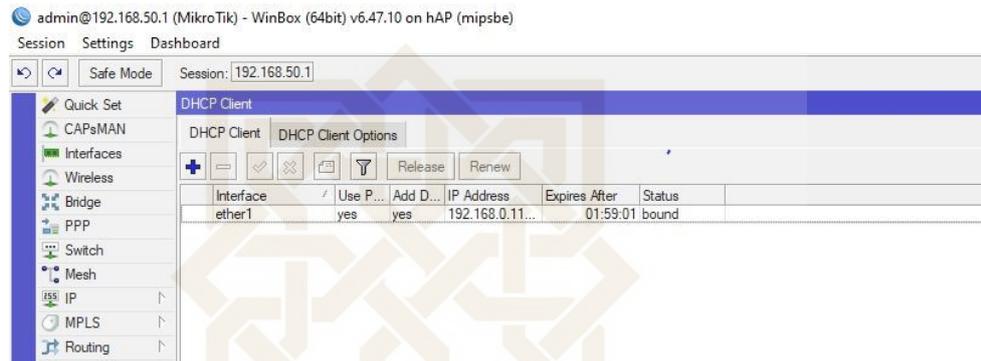
Setiap *interface ethernet* diberikan alamat IP unik beserta *subnet* yang sesuai, yang memungkinkan setiap segmen jaringan diidentifikasi dengan jelas. Misalnya *interface ether2_operator* diberikan alamat IP 192.168.30.1/24 dengan 192.168.30.0/24, dan seterusnya untuk *interface* lainnya. Tampilan setelah konfigurasi dapat dilihat pada Gambar 4.3, yang mencantumkan daftar alamat IP dan *subnet* yang diterapkan.

	Address	Network	Interface
D	192.168.0.112...	192.168.0.0	ether1
	192.168.10.1/...	192.168.10.0	wlan1
	192.168.30.1/...	192.168.30.0	ether2_operator
	192.168.40.1/...	192.168.40.0	Ether3_yayasan
	192.168.50.1/...	192.168.50.0	Ether4_administ...
	192.168.60.1/...	192.168.60.0	ether5_lab_com...

Gambar 4. 3 Tampilan *Address List*

3. Konfigurasi *DHCP Client*

Konfigurasi DHCP Client pada interface ether 1 berfungsi untuk mendapatkan alamat IP otomatis dari penyedia layanan (ISP). Hal ini menghilangkan kebutuhan pengaturan manual alamat IP pada router untuk koneksi internet. Tampilan DHCP Client ini ditunjukkan pada Gambar 4.4.



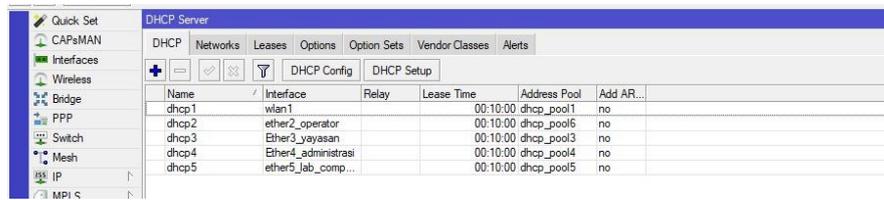
Gambar 4. 4 Tampilan *DHCP Client*

4. Konfigurasi IP Pool

IP Pool bertujuan untuk membuat beberapa DHCP pool yang mencakup berbagai subnet pada jaringan. DHCP pool berfungsi memberikan alamat IP dinamis kepada perangkat yang terhubung. Misalnya dhcp_pool6 mencakup rentang alamat IP 192.168.30.2 hingga 192.168.30.254 untuk interface ether2_operator dan seterusnya untuk interface lainnya.

5. Konfigurasi DHCP Server

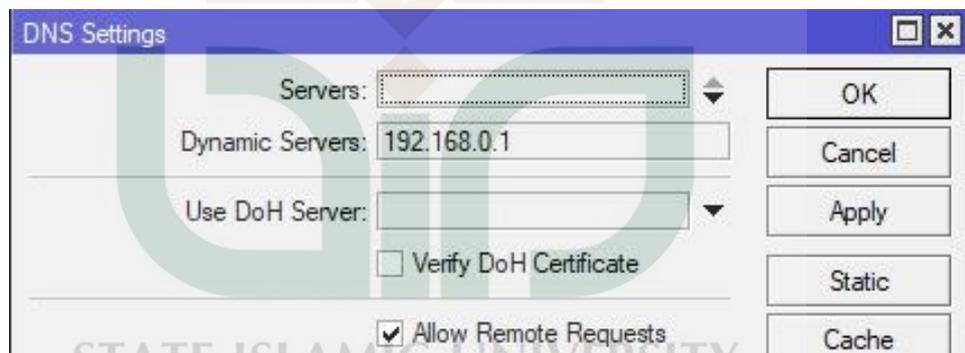
Setiap interface akan dikaitkan dengan DHCP pool yang sesuai sehingga dapat memberikan alamat IP dinamis kepada perangkat yang terhubung. Misalnya interface ether2_operator dihubungkan dengan pool dhcp_pool6, dan seterusnya. Tampilan DHCP server yang telah di setting dapat dilihat pada Gambar 4.5.



Gambar 4. 5 Tampilan *DHCP Server*

6. Konfigurasi DNS

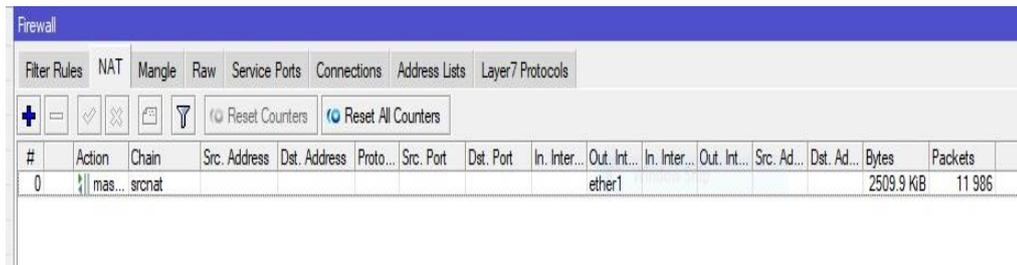
Konfigurasi DNS dilakukan agar *router* mikrotik dapat berfungsi sebagai server DNS untuk *client* pada jaringan lokal. Dengan mengaktifkan *allow remote request client* pada *DNS Setting* berfungsi untuk mengirim DNS ke *router*, kemudian akan meresolusi nama domain dengan menghubungkan *server* DNS yang ditentukan dalam pengaturan DNS *router*. Tampilan *DNS Setting* ini ditunjukkan pada Gambar 4.6.



Gambar 4. 6 Tampilan *DNS Setting*

7. Konfigurasi NAT

Konfigurasi NAT ditambahkan untuk menyamarkan alamat IP asal dari lalu lintas yang keluar dari *interface ethernet 1*. Pada Gambar 4.7, ditunjukkan bahwa NAT yang diterapkan adalah *masquerading*, dimana aturan NAT yang digunakan adalah *srcnat* (source NAT). Hal ini memungkinkan beberapa perangkat di jaringan untuk berbagi satu alamat IP publik saat mengakses internet. Berikut pada Gambar 4.7 merupakan tampilan *firewall* NAT setelah di *setting*.



#	Action	Chain	Src. Address	Dst. Address	Proto...	Src. Port	Dst. Port	In. Inter...	Out. Int...	In. Inter...	Out. Int...	Src. Ad...	Dst. Ad...	Bytes	Packets
0	mas...srcnat	srcnat						ether1						2509.9 KiB	11 986

Gambar 4. 7 Tampilan Firewall NAT

Konfigurasi dasar ini memastikan bahwa semua komponen jaringan bekerja secara efektif dengan setiap perangkat menerima konfigurasi yang tepat untuk beroperasi dengan optimal dalam jaringan yang telah dirancang.

4.3.2 Konfigurasi Hierarchical Token Bucket Menggunakan Queue Tree

Proses konfigurasi *Quality of Service* bertujuan untuk mengalokasikan *bandwidth* dan melakukan manajemen *bandwidth* secara rasional. Langkah-langkah ini mencakup pengaturan pada *firewall mangle* untuk menandai paket-paket data yang melewati *router* dan membuat *queue tree* untuk mengatur prioritas *bandwidth*.

1. Menandai Paket dan Koneksi dengan *Firewall Mangle*:

Tahap awal dalam konfigurasi QoS adalah menandai setiap paket dan koneksi data yang melewati *router* menggunakan *firewall mangle*. Hal ini penting karena penandaan menjadi dasar dalam pengaturan antrian *bandwidth* pada langkah selanjutnya. Setiap koneksi, baik *upload* maupun *download* diberi tanda khusus berdasarkan *interface* yang digunakan. Misalnya untuk *ethernet 2* yang digunakan oleh operator, koneksi *upload* diberi tanda “Koneksi upload ether 2” sementara koneksi *download* diberi tanda “Koneksi download ether 2”. Hal yang sama dilakukan untuk *ethernet 3, 4, dan 5*, yang masing-masing digunakan oleh yaysan,

admin, dan laboratorium. Tampilan pada mikrotik setelah semua aturan *mangle* yang diperlukan telah disetting ditunjukkan pada Gambar 4.8.

No	Action	Chain	Outgoing Interface	Source	Destination
4	mark connection	forward		192.168.40.0/24	
5	mark connection	forward	192.168.40.0/24		
6	mark packet	forward			
7	mark packet	forward			
8	mark connection	forward	192.168.50.0/24		
9	mark connection	forward		192.168.50.0/24	
10	mark packet	forward			
11	mark packet	forward			
12	mark connection	forward	192.168.30.0/24		
13	mark packet	forward			
14	mark connection	forward		192.168.30.0/24	
15	mark packet	forward			
16	mark connection	forward	192.168.60.0/24		
17	mark connection	forward		192.168.60.0/24	
18	mark packet	forward			
19	mark packet	forward			

Gambar 4. 8 Tampilan Firewall Mangle

2. Membuat Queue Tree untuk Quality of Service

Setelah menandai paket-paket data ditandai dengan *mangle*, langkah berikutnya adalah membuat *queue tree* untuk menerapkan kebijakan QoS. *Queue tree* memungkinkan kita mengatur prioritas *bandwidth*. Konfigurasi ini memberikan kontrol terhadap *bandwidth* yang dialokasikan kepada beberapa perangkat dan menghindari kelebihan penggunaan *bandwidth* serta ketimpangan antara besarnya *bandwidth upload* dan *download* pada perangkat. Untuk pembagian *bandwidth per client* disajikan pada Tabel 4.2.

Tabel 4. 2 Pembagian Bandwidth Jaringan Dedicated 11 Mbps

Client	Ethernet	Besar Bandwidth
Operator	Ethernet 2	1 Mbps
Yayasan	Ethernet 3	2 Mbps
Admin	Ethernet 4	5 Mbps

Laboratorium	Ethernet 5	3 Mbps
--------------	------------	--------

3. Implementasi dan Monitoring

Setelah semua konfigurasi selesai, *router* mikrotik akan mulai menerapkan kebijakan QoS yang telah ditentukan. Setiap perangkat yang terhubung ke jaringan akan menerima *bandwidth* sesuai dengan konfigurasi yang telah dibuat. Tampilan pada *router* mikrotik akan menunjukkan bagaimana *traffic* dibagi antara perangkat-perangkat tersebut. Kemudian memastikan bahwa setiap *user* mendapatkan *bandwidth* yang telah dialokasikan. Berikut ditampilkan pada Gambar 4.9 *setting queue list* untuk penerapan QoS.

total download ether 2	global		11M	0 bps	0 B 3008.3 ...	7 078
user download ether 2	total download ether 2	paket do...	1M	0 bps	0 B 3008.3 ...	7 078
total download ether 4	global		11M	0 bps	0 B 1088.3 ...	810 203
user download ether 4	total download ether 4	paket dp...	5M	0 bps	0 B 1088.3 ...	810 203
total download ether 5	global		11M	2.0 kbps	0 B 15.4 MiB	26 443
user download ether 5	total download ether 5	paket do...	3M	2.0 kbps	0 B 15.4 MiB	26 443
total download ether3	global		11M	0 bps	0 B 13.7 MiB	21 782
user download ether3	total download ether3	paket do...	2M	0 bps	0 B 13.7 MiB	21 782
total upload	global		11M	0 bps	0 B 189.9 ...	197 692
user upload	total upload	paket upl...	11M	0 bps	0 B 189.9 ...	197 692
total upload ether 2	global		11M	0 bps	0 B 2683.4 ...	2 945
user upload ether 2	total upload ether 2	paket upl...	1M	0 bps	0 B 2683.4 ...	2 945
total upload ether 4	global		11M	0 bps	0 B 41.9 MiB	413 770
user upload ether 4	total upload ether 4	paket upl...	5M	0 bps	0 B 41.9 MiB	413 770
total upload ether 5	global		11M	12.9 kb...	0 B 15.3 MiB	25 963
user upload ether 5	total upload ether 5	paket upl...	3M	12.9 kb...	0 B 15.3 MiB	25 963
total upload ether3	global		11M	0 bps	0 B 13.4 MiB	20 533
user upload ether3	total upload ether3	pake uplo...	2M	0 bps	0 B 13.4 MiB	20 533

Gambar 4. 9 Tampilan Queue List

4.4 Pengujian *Quality of Service*

Setelah mengimplementasikan metode *Hierarchical Token Bucket (HTB)* pada *router* mikrotik. Selanjutnya, dilakukan beberapa pengujian sistem yaitu pengujian koneksi dan pengujian analisis QoS.

4.4.1 Pengujian Koneksi

Pengujian ini bertujuan untuk mengukur perubahan kecepatan *download* dan *upload* sebelum dan sesudah konfigurasi HTB memakai alat speed tes yang andal yaitu *Ookla Speedtest*. Berikut disajikan pada tabel 4.3 dan 4.4 untuk perbandingan kecepatan jaringan sebelum dan sesudah adanya manajemen *bandwidth*.

Tabel 4. 3 Sebelum Implementasi HTB

<i>Client</i>	<i>Ethernet</i>	<i>Download (Mbps)</i>	<i>Upload (Mbps)</i>
Operator	<i>Ethernet 2</i>	3,62	10,57
Yayasan	<i>Ethernet 3</i>	2,56	10,11
Admin	<i>Ethernet 4</i>	3,37	11,60
Laboratorium	<i>Ethernet 5</i>	4,13	9,80

Tabel 4. 4 Sesudah Implementasi HTB

<i>Client</i>	<i>Ethernet</i>	<i>Download (Mbps)</i>	<i>Upload (Mbps)</i>
Operator	<i>Ethernet 2</i>	0,89	0,92
Yayasan	<i>Ethernet 3</i>	1,94	1,89
Admin	<i>Ethernet 4</i>	4,44	4,85
Laboratorium	<i>Ethernet 5</i>	2,90	2,81

Pengujian koneksi menggunakan *Ookla Speed Test* sebelum dan sesudah implementasi HTB pada *router* Mikrotik menunjukkan perubahan signifikan dalam distribusi *bandwidth*. Pembagian *bandwidth* yang sudah ditentukan, terbagi

sesuai kebutuhan *client*. Dengan menerapkan HTB, *bandwidth* antara *download* dan *upload* lebih merata, hal itu karena tidak adanya ketimpangan antara *upload* dan *download* pada setiap *client*.. Hal ini menunjukkan bahwa metode HTB efektif untuk mengelola trafik jaringan dan meningkatkan kualitas layanan secara keseluruhan dan rasional. Pada pengguna prioritas yaitu Admin memiliki kecepatan *bandwidth* lebih besar daripada *client* lainnya. *Bandwidth* yang sudah ditentukan untuk setiap *client* akan tetap konstan, baik saat lalu lintas jaringan sedang tinggi maupun rendah, untuk menghindari adanya perebutan *bandwidth*.

4.4.2 Pengujian dan Analisis QoS

Hasil pengujian yang telah dilakukan akan disajikan dalam bentuk tabel. Penyajian ini bertujuan untuk mempermudah analisis data mengenai kinerja jaringan terhadap hasil pengukuran *Quality of Service (QoS)* Berikut ini merupakan hasil kinerja dari masing-masing parameter *Quality of Service* yang meliputi *Throughput*, *Packet loss*, *Jitter*, dan *Delay*.

1. Hasil pengujian *Throughput*

Berikut disajikan pada Tabel 4.5 merupakan tabel pengukuran *throughput* sebelum implementasi HTB dan Tabel 4.6 merupakan tabel pengukuran setelah implementasi HTB.

Tabel 4. 5 Hasil Pengujian *Throughput* Sebelum Implementasi HTB

Nama Client	Throughput (kbps)
Operator	2582
Yayasan	6718
Admin	2323

Laboratorium	5098
--------------	------

Tabel 4. 6 Hasil Pengujian *Throughput* Sebelum Implementasi HTB

Nama Client	Throughput (kbps)
Operator	886
Yayasan	1221
Admin	4499
Laboratorium	1357

Dari hasil pengujian *throughput* pada Tabel 4.5 dan 4.6 terlihat jelas bahwa implementasi HTB berhasil mencapai tujuan utama penelitian yaitu mengatasi kehilangan kecepatan pada pengguna prioritas yaitu *client* Admin. Sebelum HTB di implementasikan admin memiliki *throughput* yang cukup rendah dibandingkan dengan client lainnya. Namun setelah implementasi HTB, *throughput* untuk client admin meningkat signifikan menunjukkan bahwa prioritas *bandwidth* telah dialokasikan secara rasional.

Untuk penurunan *throughput* pada *client* Operator, Yayasan, dan Laboratorium setelah implementasi HTB menunjukkan bahwa *bandwidth* telah dialokasikan ulang untuk memberi prioritas pada Admin. Meskipun *throughput* untuk *client* lain menurun, hasil ini dapat diterima karena tujuan utama adalah memastikan performa optimal untuk pengguna prioritas.

2. Hasil Pengujian *Packet Loss*

Berikut disajikan pada tabel 4.7 merupakan tabel pengukuran *packet loss* sebelum implementasi HTB dan tabel 4.8 merupakan tabel pengukuran setelah implementasi HTB.

Tabel 4. 7 Hasil Pengujian *Packet Loss* Sebelum Implementasi HTB

Nama Client	Packet loss (%)
Operator	0,0
Yayasan	0,6
Admin	0,0
Laboratorium	1,2

Tabel 4. 8 Hasil Pengujian *Packet Loss* Sesudah Implementasi HTB

Nama Client	Packet loss (%)
Operator	0,0
Yayasan	0,0
Admin	0,0
Laboratorium	0,4

Dari hasil pengujian *packet loss* yang disajikan pada Tabel 4.7 untuk sebelum implementasi dan Tabel 4.8 untuk sesudah implementasi terbukti efektif dalam meningkatkan keandalan jaringan dan mengurangi *packet loss*, terutama pada *client* yang sebelum implementasi mengalami masalah.

Setelah implementasi, *client* Yayasan yang sebelumnya mengalami *packet Loss* sebesar 0,6% kini tidak mengalami *packet Loss* sama sekali. Demikian pula untuk Laboratorium mengalami penurunan *Packet Loss* dari 1,2% menjadi 0,4%, menandakan peningkatan signifikan dalam keandalan jaringan. Untuk

pengguna prioritas yaitu Admin, stabilitas jaringan tetap terjaga dengan baik karena admin tidak mengalami packet loss sebelum maupun sesudah implementasi. Hal ini mengindikasikan implementasi HTB memberi dampak positif secara keseluruhan.

3. Hasil Pengujian *Delay*

Berikut disajikan pada Tabel 4.9 merupakan tabel pengukuran *delay* sebelum implementasi HTB dan Tabel 4.10 merupakan tabel pengukuran setelah implementasi HTB.

Tabel 4. 9 Hasil Pengujian *Delay* Sebelum Implementasi HTB

Nama Client	Delay (ms)
Operator	0,003
Yayasan	0,001
Admin	0,003
Laboratorium	0,001

Tabel 4. 10 Hasil Pengujian *Delay* Sesudah Implementasi HTB

Nama Client	Delay (ms)
Operator	0,007
Yayasan	0,006
Admin	0,001
Laboratorium	0,005

Dari hasil pengujian *delay* yang disajikan pada Tabel 4.9 untuk sebelum implementasi dan Tabel 4.10 untuk sesudah implementasi menunjukkan bahwa implementasi HTB telah memberi dampak yang signifikan terhadap *delay* yang diterima pada setiap *client*. Peningkatan *delay* di sisi Operator, Yayasan dan

Laboratorium mungkin disebabkan oleh penyesuaian alokasi *bandwidth* untuk memberikan prioritas yang lebih tinggi di sisi admin. Namun meskipun terdapat peningkatan *delay* pada *client* tertentu, hal ini menunjukkan bahwa HTB mengalokasikan sumber daya jaringan yang lebih efektif untuk memastikan kualitas layanan yang optimal bagi *client* yang paling membutuhkan.

4. Hasil Pengujian *Jitter*

Berikut disajikan pada Tabel 4.11 merupakan tabel pengukuran *Jitter* sebelum implementasi HTB dan Tabel 4.12 merupakan tabel pengukuran setelah implementasi HTB.

Tabel 4. 11 Hasil Pengujian *Jitter* Sebelum Implementasi HTB

Nama Client	<i>Jitter (ms)</i>
Operator	3,07
Yayasan	7,32
Admin	2,9
Laboratorium	3,3

Tabel 4. 12 Hasil Pengujian *Jitter* Sesudah Implementasi HTB

Nama Client	<i>Jitter (ms)</i>
Operator	1,44
Yayasan	2,72
Admin	1,2
Laboratorium	1,72

Dari hasil pengujian *jitter* yang disajikan pada tabel 4.11 untuk sebelum implementasi HTB dan 4.12 untuk sesudah implementasi HTB dapat disimpulkan bahwa implementasi HTB terbukti efektif dalam mengurangi *Jitter*. Implementasi HTB membantu meningkatkan stabilitas jaringan untuk semua *client*. Dengan *Jitter* yang lebih rendah, kualitas komunikasi akan semakin meningkat dan mengurangi gangguan dan penundaan yang tidak diinginkan.

4.4.3 Hasil Analisis *Quality of Service*

Untuk menganalisis layanan yang optimal dari hasil perhitungan QoS. Penulis menghitung presentase optimasi tiap parameter pada *client*. Berikut rumus untuk menghitung presentase optimasi:

$$\text{Optimasi} = \frac{\text{Nilai Baru (HTB)} - \text{Nilai Awal (Non HTB)}}{\text{Nilai Awal (Non HTB)}} \times 100\% \quad (4.1)$$

Hasil dari perhitungan presentase optimasi pada parameter QoS yang ditunjukkan pada Tabel 4.13 dan 4.14 berikut.

Tabel 4. 13 Analisis Parameter *Throughput* dan *Packet Loss*

No.	Client	Throughput		Optimasi	Packet Loss		Optimasi
		Non HTB	HTB		Non HTB	HTB	
1	Operator	2582	886	-65,69 %	0,0	0,0	0%
2	Yayasan	6718	1221	-81,82 %	0,6	0,0	100%
3	Admin	2323	4499	93,67 %	0,0	0,0	0%
4	Laboratorium	5098	1357	-73,38%	1,2	0,4	66,67%

Tabel 4. 14 Analisis Parameter *Delay* dan *Jitter*

No.	Client	Delay		Optimasi	Jitter		Optimasi
		Non HTB	HTB		Non HTB	HTB	

1	Operator	0,003	0,007	133,33%	3,07	1,44	-53,09%
2	Yayasan	0,001	0,006	500%	7,32	2,72	-62,86%
3	Admin	0,003	0,001	-66,67%	2,9	1,2	-58,62%
4	Laboratorium	0,001	0,005	400%	3,3	1,72	-47,88%

Berdasarkan Tabel 4.13 dan Tabel 4.14 untuk hasil optimasi pada masing-masing parameter QoS di setiap *client* terdapat beberapa nilai positif dan nilai negatif. Nilai positif pada *throughput* dan *packet loss* menunjukkan optimasi berjalan dengan baik. Sedangkan untuk nilai minus menunjukkan optimasi tidak berjalan dengan baik. Berbeda untuk parameter *delay* dan *jitter*. Ketika nilai positif maka menunjukkan penurunan kualitas jaringan, sedangkan nilai minus menunjukkan peningkatan kualitas jaringan. Berikut penjelasan untuk hasil analisis optimasi pada setiap parameter:

1. Dari hasil analisis *throughput*, terlihat bahwa *throughput* disisi Admin meningkat secara signifikan yaitu sebesar 93,67% setelah adanya penerapan HTB. Namun pada *client* Operator, Yayasan, dan Laboratorium, mengalami penurunan masing-masing sebesar -65,69%, -81,82%, dan -73,38%. Penurunan ini menandakan *bandwidth* yang dialokasikan untuk *client* ini telah dikurang untuk mengutamakan pengguna prioritas yaitu Admin.
2. Untuk hasil analisis *packet loss*, disisi *client* Admin dan Operator tidak terdeteksi adanya *packet loss*., Sedangkan di sisi Yayasan dan Laboratorium mengalami pengurangan yang cukup signifikan. Hal ini menunjukkan peningkatan kualitas stabilitas transmisi data setelah penerapan HTB, meskipun *throughput* berkurang.

3. Hasil analisis *delay* pada pengguna Admin mengalami penurunan sebesar 66,67%. Hal ini menunjukkan responsivitas jaringan untuk Admin meningkat secara signifikan. Untuk *client* Operator, Yayasan dan Laboratorium mengalami peningkatan. Peningkatan ini merupakan hasil dari penurunan prioritas mereka dalam alokasi *bandwidth* yang ada. Pengurangan *bandwidth* akan mengakibatkan responsivitas jaringan yang lebih rendah.
4. Hasil analisis untuk *jitter* disemua *client* mengalami penurunan. Hal ini menunjukkan peningkatan kualitas stabilitas koneksi setelah penerapan HTB. Meskipun di beberapa *client* Operator, Yayasan dan Laboratorium bukan pengguna prioritas, kualitas stabilitas jaringan tetap terjaga.

