

# ANALISA PERBANDINGAN OPTIMALISASI MANAJEMEN BANDWIDTH MIKROTIK MENGGUNAKAN METODE *QUEUE TREE* DAN *SIMPLE QUEUE* (Studi Kasus Asrama Bogani Yogyakarta Ratmakan GM 1/693)

Moh Erzal Arighi Damopolii<sup>1</sup>, Suwanto Raharjo<sup>2</sup>, Joko Triyono<sup>3</sup>

<sup>1,2,3</sup> Informatika, Fakultas Teknologi Industri, Institut Sains & Teknologi AKPRIND Yogyakarta  
JI Kalisahak No. 28 Komplek Balapan Tromol Pos 45, Yogyakarta 55222 Telp : (0274) 563029  
Email: erzaliod@gmail.com<sup>1</sup>, wa2n@akprind.ac.id<sup>2</sup>, jack@akprind.ac.id<sup>3</sup>

## ABSTRACT

*The Internet has become a daily necessity of man in this day and age. with the internet facilitates some activities, mass internet use can lead to a decrease in network performance as network users grow. Doing bandwidth management can answer some of the problems that exist because by doing management can manage excessive bandwidth usage and limit bandwidth access of some users. Bandwidth management is done by using two methods to determine which method is most optimally applied in the Bogani Dormitory internet network between simple queue and queue tree. Before applying both methods, some rare such as collecting some information by analyzing the previous network before implementing bandwidth management. After all the analysis is done obtained results with several advantages of each method, simple queue is a method that is easy to apply while the queue tree must use mangle first in order to run to the maximum, the optimal method to be applied to the Bogani Dormitory network with queue tree method although in practice takes a lot of time and knowledge compared to simple queue.*

**Keywords:** *Bandwidth Management, Internet, Mikrotik, Simple Queue, Queue Tree*

## INTISARI

*Internet telah menjadi kebutuhan sehari – hari manusia pada jaman ini. dengan adanya internet mempermudah beberapa kegiatan-kegiatan, penggunaan internet secara masal dapat menyebabkan terjadinya penurunan performansi jaringan seiring bertambahnya pengguna jaringan. Melakukan manajemen bandwidth bisa menjawab beberapa masalah yang ada dikarenakan dengan melakukan manajemen bisa mengatur penggunaan bandwidth yang berlebihan dan membatasi akses bandwidth beberapa user. Manajemen bandwidth dilakukan dengan menggunakan dua metode untuk menentukan metode yang mana paling optimal diterapkan di jaringan internet Asrama Bogani antara simple queue dan queue tree. Sebelum melakukan penerapan terhadap kedua metode, dilakukan beberapa langkah seperti pengumpulan beberapa informasi dengan melakukan analisa jaringan terdahulu sebelum menerapkan manajemen bandwidth. Setelah semua selesai dilakukan analisis didapatkan hasil dengan beberapa keunggulan masing-masing metode, simple queue adalah metode yang mudah diterapkan sedangkan queue tree harus menggunakan mangle terlebih dahulu agar berjalan dengan maksimal, metode yang optimal untuk diterapkan pada jaringan Asrama Bogani dengan metode queue tree walaupun dalam prakteknya memakan banyak waktu dan pengetahuan dibandingkan simple queue.*

**Kata Kunci :** *Manajemen Bandwidth, Internet, Mikrotik, Simple Queue, Queue Tree*

## PENDAHULUAN

Perkembangan teknologi membuat akses komunikasi dalam bertukar informasi juga menjadi lebih cepat dan mudah karena adanya internet yang mempengaruhi pola hidup manusia dalam mencari sebuah informasi. Dengan tujuan memperlancar beberapa kebutuhan mengakses informasi maka penggunaan internet harus di ataur seperti jumlah pengguna satu jaringan internet

dan mengatur lalu lintas paket yang ada dalam jaringan *internet* dengan melakukan manajemen *bandwidth*.

Manajemen *bandwidth* merupakan cara untuk mengoptimalkan berbagai jaringan dengan penerapan layanan *Quality of Services (QoS)* untuk menetapkan tipe-tipe lalu lintas jaringan, sedangkan *QoS* merupakan kemampuan untuk menggambarkan suatu tingkat pencapaian dalam suatu sistem komunikasi data. Router MikroTik adalah alat yang menyediakan layanan manajemen *bandwidth* berupa *software* maupun *hardware* yang harus di *instal* pada suatu jaringan yang akan dilakukan manajemen *bandwidth*. *Bandwidth* merupakan bagian terpenting dalam melayani jasa layanan *internet*, oleh karena itu *bandwidth* harus diatur atau dimanajemen dengan baik agar dapat memaksimalkan koneksi *internet* (Susianto, 2016).

Penggunaan *internet* secara massal mengakibatkan menurunnya performa jaringan seiring dengan bertambahnya pengguna jaringan. Cara yang bisa dilakukan untuk mengurangi penurunan performansi jaringan yaitu dengan melakukan manajemen *bandwidth*. MikroTik RouterOS bagian OS turunan dari distro *linux debian* yang khusus digunakan sebagai *router* dan *gateway*. MikroTik memiliki *QoS* yang digunakan untuk mengatur penggunaan *bandwidth* secara rasional (Syukur, 2018).

Diperlukannya manajemen *bandwidth* di jaringan Asrama Bogani Yogyakarta dikarenakan kondisi pengguna yang terdiri dari sembilan pengguna aktif sebagai penghuni Asrama Bogani dan bahkan bisa melebihi jumlah tersebut dengan tambahan *user* yang berstatus tamu Asrama Bogani. Melihat kondisi yang ada jaringan yang tidak melakukan manajemen *bandwidth* akan mempengaruhi kecepatan akses saat seluruh *user* menggunakan jaringan Asrama Bogani dengan kebutuhan *bandwidth* yang berbeda sehingga menyebabkan *traffic internet* yang tidak lancar.

Penulisan penelitian ini menggunakan beberapa sumber yang berkaitan dengan objek penelitian sebagai referensi guna mendapatkan informasi tentang metode yang terkait dalam melakukan manajemen *bandwidth*. Informasi yang digunakan untuk penulisan ini menggunakan jurnal yang berkaitan dengan penelitian yang akan dilakukan. Penelitian yang dilakukan oleh Dirja (2018), penelitian dilakukan dengan adanya permasalahan aktifitas yang sangat tinggi dalam penggunaan *internet* di lingkungan politeknik Aceh baik digunakan untuk *browsing* informasi, melakukan *download data*, *social media* dan beberapa aktifitas lainnya, penelitian ini dilakukan untuk melihat perbandingan dua metode manajemen *bandwidth* yang akan dilakukan di Politeknik Aceh Selatan untuk mendapatkan metode yang sesuai dengan aktifitas penggunaan *internet* di lingkungan sekitar agar *user* dapat menggunakan *internet* dengan lancar dengan *bandwidth* yang sama pada setiap *user* yang melakukan akses *internet* di lingkungan tersebut. Hasil dari penelitian ini menunjukkan *simple queue* dengan kemudahan dalam melakukan konfigurasi juga tidak mudah ditembus oleh *download manager* namun terdapat banyak *bandwidth* yang terbuang berbeda dengan *queue tree* yang bisa menggunakan semua *bandwidth* dan memiliki beberapa kerumitan dalam melakukan konfigurasi ditambah dapat ditembus *download manager* tanpa melakukan setting *mangle*, dalam hal *throughput simple queue* memiliki lebih besar dari pada *queue tree* dan *delay* yang dihasilkan *simple queue* lebih besar dari pada *queue tree* dan *packet lost* yang dihasilkan juga lebih besar dari *queue tree*,

Penelitian yang dilakukan oleh Arif (2015), penelitian dilakukan berdasarkan permasalahan kebutuhan akses *internet* yang semakin besar dan kendala terhadap pekerjaan beberapa karyawan yang harus menggunakan *internet* menjadi terhambat karena kapasitas *bandwidth* yang belum terbagi merata untuk setiap *user* dan beberapa *user* karyawan harus memiliki prioritas yang tinggi untuk penggunaan *internet* di area PT. Endorsindo Makmur Selaras, hasil dan kesimpulan penelitian ini *bandwidth* dapat dibagi dengan jumlah pengguna yang ada dan pembagian *bandwidth* menggunakan *queue tree* dapat disesuaikan dengan kebutuhan setiap pengguna tanpa adanya *user* yang menggunakan *bandwidth* yang terlalu besar.

Penelitian yang dilakukan oleh Hadi & Wibowo (2018), penelitian dilakukan karena peningkatan kebutuhan akses *internet* pada Universitas Semarang dengan beberapa aktifitas seperti akses jurnal, *download*, *streaming* maupun hanya sekedar mengakses jaringan sehingga kebutuhan *bandwidth* meningkat lebih tinggi yang dapat mengakibatkan terganggunya beberapa aktifitas akademik yang lebih utama. Hasil kesimpulan penelitian metode *queue tree* memiliki optimasi yang baik dalam melakukan manajemen *bandwidth* dengan penggunaan metode tersebut

menunjukkan manajemen *bandwidth* berjalan dengan baik pada setiap *client* di setiap gedung dengan koneksi yang sangat baik.

Penelitian yang dilakukan oleh Faisal & Fauzi (2018), penelitian dilakukan dengan permasalahan adanya tarik menari kapasitas *bandwidth* antar pengguna sehingga diperlukan manajemen *bandwidth* menggunakan metode *queue tree* dan PCQ untuk memaksimalkan pembagian *bandwidth*, *queue tree* digunakan untuk implementasi fungsi yang kompleks dalam pelimitan *bandwidth* karena penggunaan *packet mark* yang lebih baik untuk membatasi satu arah koneksi saja baik *download* maupun *upload* sedangkan PCQ untuk mengenali arus dan dapat membagi *bandwidth* secara adil dan merata walaupun jumlah *client* yang banyak dan tidak tentu hanya perlu membuat satu atau dua konfigurasi *queue* untuk menggunakan PCQ. Hasil kesimpulan penelitian diperoleh *delay* dan *jitter* yang menggunakan manajemen *bandwidth* dengan metode *queue tree* dan PCQ menghasilkan nilai yang kecil dibandingkan tanpa metode tersebut dan *throughput* dengan metode *queue tree* dan PCQ menghasilkan nilai yang besar daripada tidak menggunakan metode tersebut, berdasarkan beberapa hasil diatas menunjukkan dengan penggunaan metode di atas lebih optimal dan tanpa ada *client* yang saling merebut *bandwidth*.

Penelitian yang dilakukan oleh Widayanto, Susilo, & Saputro (2016), Penelitian ini dilakukan dengan beberapa permasalahan pada jaringan yang diberikan ISP (*Internet Service Provider*) sebesar 15 Mbps dan total *bandwidth* sebelumnya sebesar 6 Mbps yang dibagi ke seluruh pengguna *internet* menggunakan metode *simple queue* dengan alokasi *bandwidth* yang telah ditentukan administrator jaringan pada laboratorium komputer dan adanya penambahan kapasitas maka kapasitas pada laboratorium komputer menjadi 2 Mbps dan dibutuhkan manajemen *bandwidth* untuk memaksimalkan pembagian *bandwidth* pada seluruh komputer yang ada di laboratorium komputer sehingga dilakukan perbandingan pada kedua metode manajemen *bandwidth simple queue* dan *queue tree*. Hasil kesimpulan penelitian dengan menggunakan Mikrotik RB-750 dan alokasi *download* dan *upload* sebesar 2 Mbps dengan metode konfigurasi yang dilakukan bergantian *simple queue* dan *queue tree* berdasarkan *Hierarchical Token Bucket* (HTB) total *inner queue upload* dan *download* dengan *max limit* 2 Mbps menjadi *limit-at* 102 Kbps dan *max limit* 2 Mbps setiap proses *upload* dan *download* direkam aktifitas jaringan menggunakan *wireshark* dan digunakan untuk bahan analisis penentuan nilai parameter QoS.

## METODOLOGI PENELITIAN

### Langkah dan *Diagram* Alir Langkah Penelitian

#### Langkah Penelitian

Langkah-langkah yang dilakukan dalam penelitian ini dibagi ke dalam 5 tahap yang dapat diuraikan sebagai berikut :

1. Analisis Aktifitas *Internet*

Proses untuk mendapatkan informasi tentang aktifitas yang sering dilakukan di jaringan *internet* Asrama Bogani termasuk design atau topologi arsitektur jaringan yang ada. Pada tahap ini perlu menganalisa apa saja aktifitas yang sering dilakukan, informasi aktifitas yang di dapat nantinya akan menjadi poin untuk melakukan manajemen *bandwidth* sehingga nantinya aktifitas yang penting dan yang banyak di lakukan pengguna yang akan menjadi prioritas dalam pembagian *bandwidth* dibandingkan dengan aktifitas yang lainnya.

2. Menentukan Metode Manajemen *Bandwidth*

Pada bagian ini akan ditentukan metode yang diperlukan untuk melakukan manajemen *bandwidth* berdasarkan aktifitas *internet* pada bagian sebelumnya, bagian ini akan digunakan metode manajemen *bandwidth simple queue* dan *queue tree* untuk menentukan metode mana yang paling optimal untuk jaringan Asrama Bogani dengan beberapa aktifitas yang sering dilakukan. Digunakannya dua metode tersebut karena terdapat beberapa perbedaan dari dua metode tersebut dari tingkat efisiensi melakukan konfigurasi dan beberapa fitur yang ada pada kedua metode tersebut.

3. Eksperimen Metode Manajemen *Bandwidth*

Dalam tahap eksperimen dilakukan uji coba langsung pada jaringan dengan menggunakan dua metode yang ada secara bergantian, pada bagian ini akan dilakukan konfigurasi untuk melimitasi beberapa IP *Address* pengguna *internet* dengan menentukan jumlah besaran *upload* dan *download* sesuai dengan jumlah kapasitas *bandwidth* yang disediakan penyedia layanan dan melakukan beberapa *packet mark* pada aktifitas *internet* agar tidak mengganggu aktifitas lain.

4. Analisa Metode Manajemen *bandwidth*

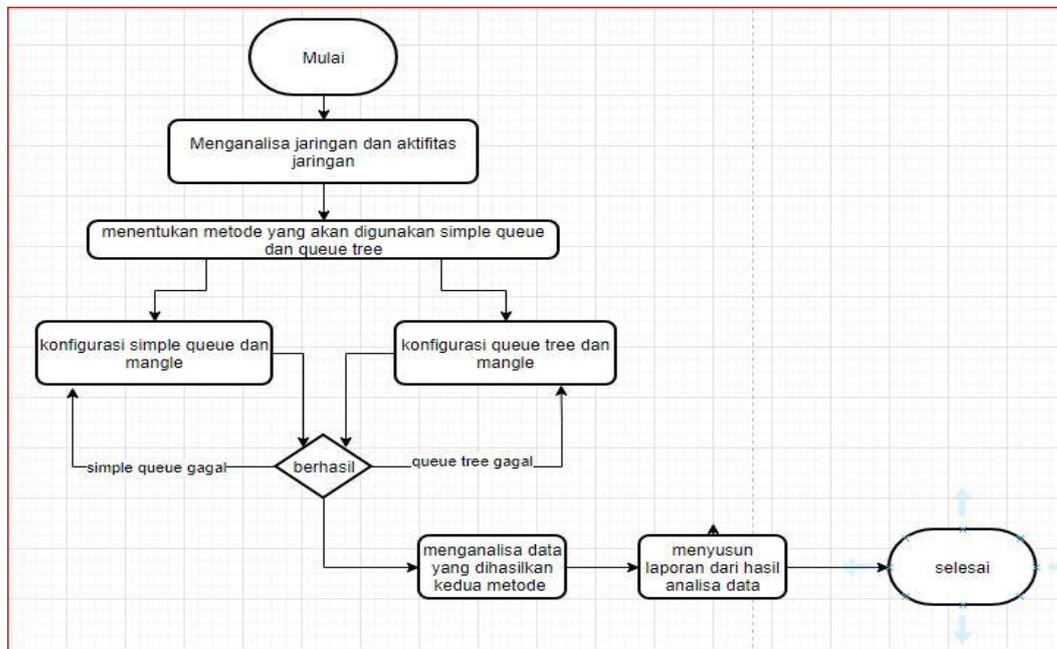
Setelah beberapa tahap di atas terpenuhi, hasil dari beberapa tahap diatas biasanya di uji terlebih dahulu. Pada tahap ini hasil dari ketiga tahap di atas dilakukan analisa sesuai dengan beberapa acuan parameter untuk menentukan metode yang paling optimal sebelum diterapkan pada jaringan *internet* Asrama Bogani. Beberapa parameter yang banyak menjadi acuan untuk menentukan metode yang optimal pada suatu jaringan dengan diperolehnya beberapa nilai yang dihasilkan oleh parameter seperti *throughput* semakin besar nilai yang di hasilkan maka semakin baik *transfer* data suatu jaringan, *delay* semakin kecil nilai yang dihasilkan maka semakin baik jaringan tersebut begitu juga dengan *packet lose* dan *jitter*.

5. Penerapan Metode Hasil Analisis

Penerapan dilakukan setelah hasil yang didapat pada bagian analisa telah diketahui, metode yang paling optimal yang telah melalui tahap analisa metode nantinya akan diterapkan agar beberapa aktifitas *internet* di Asrama Bogani tidak terganggu demi kenyamanan pengguna jaringan.

Diagram Alir Penelitian

Diagram alir penelitian manajemen *bandwidth* menggunakan dua metode manajemen *bandwidth* MikroTik pada jaringan *internet* Asrama Bogani Yogyakarta, dengan menganalisis aktifitas *internet* maka akan diprioritaskan aktifitas apa yang penting dan akan dikonfigurasi menggunakan *mangle* agar aktifitas yang menjadi prioritas tidak terganggu dengan yang lain. Gambar 1. penggambaran *diagram* alir penelitian:



Gambar 1. Diagram Alir Penelitian

HASIL DAN PEMBAHASAN

Hasil dan Pembahasan Penelitian

Pada analisa hasil menggunakan *wireshark* dan untuk melakukan *test* pada *wireshark* pertama pilih *capture* data sesuai dengan *interface* yang digunakan untuk terhubung ke jaringan disini digunakan *wifi* untuk koneksi jaringan selanjutnya *wireshark* akan melakukan analisa data dengan *user* melakukan *download file* sebesar 50 Mb lalu akan diambil data dengan tiga tahap yang akan dibagi menjadi kategori aktifitas rendah , aktifitas sedang dan aktifitas tinggi yang akan diterapkan pada kedua metode yang akan diuji.

Selanjutnya melakukan perhitungan mencari *jitter*, *throughput*, *delay* dan *packet lost* untuk masing-masing kategori aktifitas dengan dua metode yang ada, berikut ini akan dijelaskan rumus untuk mencari beberapa data di atas.

a. Throughput

*Throughput* merupakan kecepatan transefer data efektif, yang diukur dalam bps. *Throughput* juga merupakan total kedatangan paket yang sukses yang diamati pada tujuan selama interval waktu tertentu dibagi oleh waktu interal durasi tersebut.

Tabel 1. kategori standar *tiphon* untuk *throughput*.

**Tabel 1. Tabel Standarisasi Throughput dari Tiphon**

Kategori Degradasi	<i>Throguhput</i>	Indeks
Sangat Bagus	100 %	4
Bagus	75 %	3
Sedang	50 %	2
Jelek	< 25 %	1

$$Throughput = \frac{\text{Jumlah data yang dikirim}}{\text{Waktu pengiriman data}}$$

$$Throughput = \frac{\text{Total throughput}}{\text{Jumlah bandwidth}} \times 100\%$$

Untuk mencari *Throughput* menggunakan persamaan di atas dengan menggunakan data *wireshark* untuk mengetahui jumlah data yang dikirim pada *wireshark* digunakan data yang ditampilkan dengan nama *bytes* sedangkan untuk waktu pengiriman data menggunakan data dengan nama *Time Span*, lalu akan mendapatkan hasil dari *Throughput*.

b. Packet Lost

*Packet lost* adalah parameter suatu kondisi yang menunjukkan jumlah total paket data yang hilang, terjadi karena *collison* (tabrakan) dan *congestion* (kemacetan) semua aplikasi karena transmisi akan mengurangi efisiensi jaringan secara keseluruhan meskipun jumlah *bandwidth* cukup tersedia untuk aplikasi-aplikasi tersebut.

Tabel 2. kategori standar *tiphon* untuk *packet lost*.

**Tabel 2. Tabel Standarisasi Packet Lost dari Tiphon**

Kategori Degradasi	<i>Packet Lost</i>	Indeks
Sangat Bagus	0 %	4
Bagus	3 %	3
Sedang	15 %	2
Jelek	25 %	1

$$Packet Lost = \frac{\text{Total paket dikirim} - \text{Total paket diterima}}{\text{Total paket diterima}} \times 100\%$$

Untuk mencari hasil dari *packet lost* menggunakan persamaan di atas, dalam *wireshark* bisa melihat hasil dari *packet lost* dengan melakukan *filter* pada data dengan perintah *tcp.analysis.lost\_segment* nantinya *wireshark* akan menampilkan paket yang hilang lalu disini bisa juga menampilkan data pada bagian *statistic* dengan memilih *capture file properties* dan data akan ditampilkan.

c. Delay

*Delay* merupakan waktu yang dibutuhkan data untuk menempuh jarak dari sumber ke tujuan.

Tabel 3. kategori standar *tiphon* untuk *delay*.

**Tabel 3. Tabel Standarisasi Delay dari Tiphon**

Kategori Degradasi	<i>Delay</i>	Indeks
Sangat Bagus	< 150 ms	4
Bagus	150 ms s/d 300 ms	3
Sedang	300 ms s/d 450 ms	2
Jelek	> 450 ms	1

$$Rata - rata\ delay = \frac{Total\ delay}{Total\ paket\ yang\ diterima}$$

Untuk mencari *delay* menggunakan rumus di atas dan untuk mencari rata-rata *delay* akan digunakan data *time* pada *wireshark* lalu akan dibuat menjadi dua dengan nama *time 1* dan *time 2* selanjutnya pengambilan data *time* pada *wireshark* dari no satu sampai paling bawah sedang untuk *time 2* dimulai mengambil data dari no dua sampai paling bawah lalu kurangi data *time 2* dengan data *time 1* dan akan didapatkan hasil *delay* lalu jumlahkan semua sehingga total *delay* akan didapatkan selanjutnya mencari total paket jumlah data harus dikurangi dengan data yang gagal dikirim (*Packet Lost*) dan akan didapatkan total paket data yang diterima terakhir mencari rata-rata *delay* dengan rumus di atas.

d. Jitter

*Jitter* adalah variasi kedatangan paket, yang diakibatkan oleh variasi-variasi dalam panjang antrian, dalam waktu pengolahan data, dan juga dalam waktu penghimpunan ulang paket-paket di akhir perjalanan.

Tabel 4. kategori standar *tiphon* untuk *jitter*.

**Tabel 4. Tabel Standarisasi Jitter dari Tiphon**

Kategori Degradasi	Jitter	Indeks
Sangat Bagus	0 ms	4
Bagus	0 ms s/d 75 ms	3
Sedang	75 ms s/d 125 ms	2
Jelek	125 ms s/d 225 ms	1

$$Jitter = \frac{Total\ variasi\ delay}{Total\ Paket\ Yang\ Diterima}$$

Total variasi delay diperoleh dengan penjumlahan:

$$((delay\ 2 - delay\ 1) + (delay\ 3 - delay\ 2) + ..... + (delay\ n - delay\ (n-1)))$$

Untuk mencari *jitter* menggunakan data yang didapat dari perhitungan *delay* dimana untuk mencari total variasi *delay* menggunakan penjumlahan di atas lalu akan didapatkan hasil total variasi delay.

Setelah mengetahui langkah-langkah untuk mendapatkan hasil *throughput*, *packet lost*, *delay* dan *jitter* selanjutnya menghitung data yang telah didapatkan dengan *wireshark* lalu menampilkan data yang dihasilkan berupa tabel, data yang akan diolah nanti berdasarkan rumus di atas terbagi dalam tiga kategori yang dilakukan pada kedua metode yang digunakan ketiga kategori tersebut adalah aktifitas rendah, aktifitas sedang dan aktifitas tinggi.

Tabel 5. *simple queue* menggunakan *wireshark* kategori aktifitas rendah.

**Tabel 5. Tabel Hasil Simple Queue Aktifitas Rendah**

Hasil pengujian <i>simple queue</i> Kategori aktifitas rendah dengan user aktif 3 perangkat				
No	Parameter	Hasil	Indeks	Standar Tiphon
1	<i>Throughput</i>	251638.5 bps (100 %)	4	Sangat Bagus
2	<i>Packet Lost</i>	2.569674 (2.6 %)	3	Bagus
3	<i>Delay</i>	3.18549935 s (3 ms)	4	Sangat Bagus
4	<i>Jitter</i>	3.185446911 s (3 ms)	3	Bagus

Tabel 5. adalah hasil dari *simple queue* data yang didapatkan dengan *wireshark* dihitung menggunakan rumus yang ada pada penjelasan sebelumnya untuk *wireshark* sendiri ada beberapa data yang sudah diolah sehingga tinggal menggunakan data tersebut seperti perhitungan *throughput* dan *packet lost* bisa didapatkan tanpa menghitung manual lagi.

Data di atas menunjukkan nilai *throughput* dengan persentase 100 % dan memenuhi syarat *tiphon* pada kategori sangat bagus, *packet lost* hasil yang ditampilkan adalah

persentase data yang hilang sebesar 2,6 % dan mendapat kategori bagus, *delay* mendapatkan hasil 3 ms dan masi pada kategori sangat bagus sedangkan *jitter* mendapatkan hasil yang sama dengan *delay* yaitu 3 ms dengan kategori bagus, hasil diatas didapatkan dengan percobaan *download* file 50 Mb pada dua perangkat yang aktif dan yang lainnya melakukan aktifitas yang lainnya.

**Tabel 6. Tabel Hasil Simple Queue Aktifitas Rendah**

Hasil pengujian <i>queue tree</i> Kategori aktifitas rendah dengan user aktif 3 perangkat				
No	Parameter	Hasil	Indeks	Standar Tiphon
1	<i>Throughput</i>	584311.9182 bps (100 %)	4	Sangat Bagus
2	<i>Packet Lost</i>	2.299156976 (2.3 %)	3	Bagus
3	<i>Delay</i>	1.403221252 s (1 ms)	4	Sangat Bagus
4	<i>Jitter</i>	1.403197874 s (1 ms)	3	Bagus

Tabel 6. adalah tampilan hasil dari *queue tree* data yang diperoleh menggunakan *wireshark* dengan hasil *throughput* yang didapatkan mempunyai persentase 100 % dan dikategorikan sangat bagus, *packet lost* data yang didapat adalah persentase dari data yang hilang sebesar 2,3 % dan mendapatkan kategori bagus, *delay* dan *jitter* yang didapatkan memiliki nilai yang sama yaitu 1 ms tetapi untuk kategori *delay* mendapatkan hasil sangat bagus sedangkan untuk *jitter* mendapatkan kategori bagus.

Tabel 7. *simple queue* menggunakan *wireshark* kategori aktifitas sedang.

**Tabel 7. Tabel Hasil Simple Queue Aktifitas Sedang**

Hasil pengujian <i>simple queue</i> Kategori aktifitas sedang dengan user aktif 5 perangkat				
No	Parameter	Hasil	Indeks	Standar Tiphon
1	<i>Throughput</i>	204.3255817 bps (100 %)	4	Sangat Bagus
2	<i>Packet Lost</i>	2.37793461 (2.4 %)	3	Bagus
3	<i>Delay</i>	3.912423736 s (3 ms)	4	Sangat Bagus
4	<i>Jitter</i>	3.912367386 s (3 ms)	3	Bagus

Tabel 7. adalah hasil dari *simple queue* dengan kategori aktifitas sedang dengan dua pengguna melakukan *download* pada *file* 50 Mb dan yang lainnya melakukan aktifitas *streaming* atau aktifitas lainnya, data *throughput* yang didapat persentasenya 100% dan mendapat kategori sangat bagus, *packet lost* yang diperoleh dengan persentase data yang hilang sebesar 2.4 % dan masih dalam kategori bagus, *delay* dan *jitter* mendapatkan nilai yang sama sebesar 3 ms tetapi memperoleh kategori yang berbeda untuk *delay* mendapat kategori sangat bagus sedangkan untuk *jitter* mendapatkan kategori bagus. Tabel 6. *queue tree* menggunakan *wireshark* kategori aktifitas sedang.

**Tabel 8. Tabel Hasil Queue Tree Aktifitas Sedang**

Hasil pengujian <i>queue tree</i> Kategori aktifitas sedang dengan user aktif 5 perangkat				
No	Parameter	Hasil	Indeks	Standar Tiphon
1	<i>Throughput</i>	352828.2185 bps (100 %)	4	Sangat Bagus
2	<i>Packet Lost</i>	1.541223546 (1.5 %)	3	Bagus
3	<i>Delay</i>	2.352377896 s (2 ms)	4	Sangat Bagus
4	<i>Jitter</i>	2.352342106 s (2 ms)	3	Bagus

Tabel 8. adalah hasil *queue tree* dengan melakukan percobaan yang sama dengan *simple queue* dimana ada dua *user* yang akan melakukan *download file* sebesar 50 Mb, *throughput* yang dihasilkan mendapat persentase 100 % dengan kategori sangat bagus, *packet lost* data yang diperoleh dengan persentase paket yang hilang 1.5 % dan

dikategorikan bagus, *delay* dan *jitter* mendapatkan hasil yang sama 2 ms tetapi mendapatkan kategori yang berbeda dimana *delay* mendapatkan kategori sangat bagus sedangkan *jitter* mendapatkan kategori bagus.

**Tabel 9. Tabel Hasil Simple Queue Aktifitas Tinggi**

Hasil pengujian <i>simple queue</i> Kategori aktifitas tinggi dengan user aktif 9 perangkat				
No	Parameter	Hasil	Indeks	Standar Tiphon
1	<i>Throughput</i>	248042.7373 bps (100 %)	4	Sangat Bagus
2	<i>Packet Lost</i>	2.760509177 (2.8 %)	3	Bagus
3	<i>Delay</i>	3.194702388 s (3 ms)	4	Sangat Bagus
4	<i>Jitter</i>	3.194662982 s (3 ms)	3	Bagus

Tabel 9. adalah hasil *simple queue* dengan kategori aktifitas tinggi dengan *user* yang melakukan *download* hanya 1 perangkat saja dan sisanya melakukan aktifitas yang lain, *throughput* yang dihasilkan mendapat persentase 100 % dengan kategori sangat bagus, *packet lost* yang didapat dengan persentase paket yang hilang 2.8 % dan masih dalam kategori bagus, *delay* dan *jitter* mendapatkan nilai yang sama yaitu 3 ms tetapi untuk kategori *delay* mendapatkan kategori sangat bagus sedangkan *jitter* mendapatkan kategori bagus.

Tabel 10. *queue tree* menggunakan *wireshark* kategori aktifitas tinggi.

**Tabel 10. Tabel Hasil Queue Tree Aktifitas Tinggi**

Hasil pengujian <i>queue tree</i> Kategori aktifitas tinggi dengan user aktif 9 perangkat				
No	Parameter	Hasil	Indeks	Standar Tiphon
1	<i>Throughput</i>	854335.0215 bps (100 %)	4	Sangat Bagus
2	<i>Packet Lost</i>	1.526381449 (2.5 %)	3	Bagus
3	<i>Delay</i>	0.976615983 s (0 ms)	4	Sangat Bagus
4	<i>Jitter</i>	0.976601649s (0 ms)	4	Sangat Bagus

Tabel 10. adalah hasil dari *queue tree* yang didapatkan dengan kategori aktifitas tinggi dimana kali ini hanya melakukan *download* file 50 Mb pada satu perangkat saja sedangkan yang perangkat lain melakukan aktifitas *streaming* dan lainnya, *throughput* yang didapatkan persentasenya 100 % dan masuk dalam kategori sangat bagus, *packet lost* yang dihasilkan dengan persentase data yang hilang sebesar 2.5 % dengan kategori bagus, *delay* dan *jitter* menghasilkan nilai yang sama yaitu 0 ms dengan kategori yang sama yaitu sangat bagus.

## KESIMPULAN

Pada bagian ini disimpulkan hasil kesimpulan penelitian ini yang akan dijabarkan dalam poin-poin berikut ini :

- a. Untuk melakukan konfigurasi kedua metode mempunyai keunggulan dan kekurangan masing-masing. *Simple queue* memiliki keunggulan dalam efisiensi waktu dan kemudahan dalam melakukan manajemen *bandwidth* sedangkan untuk kelemahannya tidak bisa memaksimalkan *bandwidth* yang tidak terpakai, *queue tree* memiliki kekurangan dalam bagian konfigurasi karena sangat rumit dan butuh pengetahuan lebih tetapi mempunyai kelebihan bisa memaksimalkan *bandwidth* yang sedang tidak dipakai.
- b. Penerapan unsur-unsur konfigurasi pada kedua metode sangat mempengaruhi hasil yang menjadikan salah satu metode sangat optimal untuk digunakan.
- c. Berdasarkan kesimpulan poin pertama dan kedua *queue tree* lebih optimal untuk digunakan di Asrama Bogani dengan konfigurasi dan pembagian *bandwidth* pada koneksi *youtube* dan koneksi umum ketika *bandwidth* dari salah satu koneksi tidak terpakai maka koneksi lain bisa memakainya, tidak seperti pada metode *simple queue* yang tidak bisa menggunakan *bandwidth* seluruhnya walaupun *bandwidth* sedang tersedia dengan konfigurasi yang dilakukan.

**Daftar Pustaka**

- Anhar. (2016). *Panduan Bijak Belajar Internet untuk Anak*. Adamssein Media.
- Arif. (2015). *Manajemen Bandwidth Simple Queue Dan Queue Tree Pada PT. Endorsindo Makmur Selaras*.
- Bryan Gambrel & Microsoft Corporation. (2011). *Networking fundamentals, exam 98-366*. John Wiley & Sons, Inc.
- Citraweb Solusi Teknologi, PT. (n.d.). *Memisahkan Traffic Game Online dan Browsing*. Retrieved 12 7, 2020, from [www.mikrotik.co.id](http://www.mikrotik.co.id): [http://www.mikrotik.co.id/artikel\\_lihat.php?id=106](http://www.mikrotik.co.id/artikel_lihat.php?id=106)
- Dirja. (2018). *Implementasi Metode Simple Queue dan Queue Tree Untuk Optimasi Manajemen Bandwidth Jaringan Komputer Di Politeknik Aceh Selatan*.
- Faisal, I., & Fauzi, A. (2018). *Analisa QoS Pada Implementasi Manajemen Bandwidth menggunakan metode Queue Tree dan PCQ (Per Connection Queueing)*.
- Hadi, S., & Wibowo, R. (2018). *Implementasi Manajemen Bandwidth Menggunakan Queue Tree Pada Universitas Semarang*.
- Hart, T. (2017). *Networking with MikroTik*.
- Lowe, D. (2011). *Networking For Dummies*. Hoboken: Wiley Publishing, Inc.
- Park, K. I. (2005). *QoS In Packet Networks*. Boston: Springer Science + Business Media, Inc.
- Susianto, D. (2016). *Implementasi Queue Tree Untuk Manajemen Bandwidth Menggunakan Router Board Mikrotik*.
- Syukur, A. (2018). *Analisis Management Bandwidth Menggunakan Metode Per Connection Queue (PCQ) dengan Authentikasi RADIUS*.
- Towidjojo, R. (2015). *Mikrotik Kungfu*. Jasakom.
- Widayanto, A. E., Susilo, D., & Saputro, F. H. (2016). *Manajemen Bandwidth Dengan Simple Queue dan Queue Tree Di laboratorium Komputer Universitas Sahid Surakarta*.