

Analisis Unjuk Kerja TCP Window Size 64k Menggunakan Algoritma TCP New Reno pada Jaringan Wired dan Wireless

Salman¹, Jafaruddin Gusti Amri Ginting², Reni Dyah Wahyuningrum³

^{1,2,3})Program Studi S1 Teknik Telekomunikasi, Institut Teknologi Telkom Purwokerto
Jl. D.I. Panjaitan No. 128, Purwokerto Selatan 53147
salmanmen98@gmail.com

ABSTRAK

Kemacetan jaringan merupakan masalah serius di jaringan internet yang dapat menimbulkan peningkatan jumlah packet loss. Kemacetan juga dapat menjadi beban jaringan sehingga dapat memperlambat koneksi jika tidak ditangani dengan baik, bahkan dapat menyebabkan kelumpuhan dalam jaringan. Transport Control Protocol (TCP) menyediakan mekanisme transfer data yang dapat diandalkan (reliable), sehingga aliran data yang dibaca TCP receiver tidak rusak, tanpa duplikasi, dan berurutan. Pada penelitian ini, penulis melakukan simulasi menggunakan TCP Window Size 64K dengan Algoritma TCP New Reno pada Wired Fast Ethernet dan Wireless 802.11n. Penelitian ini melakukan simulasi untuk mengatasi kemacetan pada jaringan wired dan wireless lalu membandingkan hasil dari parameter Quality of Service (QoS) yang diuji diantaranya adalah throughput, delay, jitter, dan packet loss. Proses simulasi penelitian ini menggunakan Software Riverbed sebagai simulator untuk merancang dan mencari nilai parameter QoS pada jaringan wired dan wireless tersebut. Hasil dari penelitian ini menunjukkan bahwa throughput dan delay yang dihasilkan pada jaringan Wireless 802.11n memiliki performa yang lebih baik daripada Wired Fast Ethernet dengan nilai throughput 13050,6 bit/sec dan nilai delay 0,000253344 sec. Adapun jaringan Wired Fast Ethernet memiliki performa jitter dan packet loss yang lebih baik daripada Wireless 802.11n dengan nilai jitter 0,0000000000003 sec dan nilai packet loss 0%.

Kata Kunci: Wired Fast Ethernet, Wireless 802.11n, TCP Window Size 64K, TCP New Reno, Quality of Service (QoS)

ABSTRACT

Congestion Network congestion is a serious problem in internet networks that can cause an increase in the number of packet loss. Congestion can also become a burden on the network so that it can slow down the connection if not handled properly, and can even cause paralysis in the network. The Transport Control Protocol (TCP) provides a reliable data transfer mechanism, so that the data streams that are read by the TCP receiver are not damaged, without duplication, and sequential. In this study, the authors simulated using TCP Window Size 64K with the New Reno TCP Algorithm on Wired Fast Ethernet and Wireless 802.11n. This study conducted a simulation to solve congestion on wired and wireless networks and then compared the results of the Quality of Service (QoS) parameters tested, including throughput, delay, jitter, and packet loss. The simulation process of this research uses Riverbed Software as a simulator to design and find the QoS parameter values on the wired and wireless networks. The results of this study indicate that the resulting throughput and delay on the Wireless 802.11n network has better performance than Wired Fast Ethernet with a throughput value of 13050.6 bits / sec and a delay value of 0.000253344 sec. The Wired Fast Ethernet network has better jitter and packet loss performance than Wireless 802.11n with a jitter value of 0.0000000000003 sec and a packet loss value of 0%.

Keywords: wired fast ethernet, wireless 802.11n, tcp window size 64k, tcp new reno, quality of service (qos)

1. PENDAHULUAN

Pada saat ini, berbagai aktivitas manusia tidak lepas dari jaringan internet. Komunikasi dan pertukaran informasi saat ini pengguna sangat menuntut kecepatan pengiriman, keamanan, dan juga jaminan agar informasi sampai kepada penerima secara utuh. Penelitian dan pengembangan teknologi di bidang telekomunikasi

khususnya *internet* berkembang dengan pesat. Berbagai inovasi dilakukan oleh banyak peneliti untuk menciptakan suatu teknologi jaringan yang efisien dalam melakukan komunikasi. Pengembangan tidak berhenti pada suatu titik saja, tetapi dilakukan secara terus menerus untuk memenuhi kebutuhan informasi yang semakin banyak dan juga cepat. Media transmisi saat ini

terbagi menjadi dua bagian yaitu jaringan kabel *wired* dan jaringan *wireless*.

IEEE (*Institute of Electrical Electronics Engineers*) 802.11n dikeluarkan pada tahun 2009 dan baru di produksi perangkat maupun *device* nya baru-baru ini. Standart ini secara teori mampu mentransferkan kecepatan data hingga 600 *Mbps* tetapi ketika di uji coba oleh *WiFi alliance* yaitu badan yang menguji standart ini kecepatan nya hanya 450 *Mbps*[2].

Fast Ethernet merupakan salah satu tipe *ethernet* yang mempunyai kecepatan hingga 100 *Mbps*, dan telah disetujui oleh IEEE802.3u dengan standarisasi yang digunakan adalah 100*BaseFX*, 100*BaseT*, 100*BaseT4*, 100*BaseTX*. *Fast Ethernet* merupakan teknologi jaringan yang masih banyak digunakan saat ini [3].

Transmission Control Protocol (TCP) *Window Size* 64K merupakan salah satu ukuran TCP *Window* yang memiliki kinerja yang lebih baik dan lebih stabil terhadap response yang diberikan dibandingkan TCP *Window Size* yang lainnya dikarenakan semakin besar ukuran TCP *Window Size* maka semakin besar data yang dapat ditransmisikan [4].

TCP *New Reno* merupakan pengembangan dari TCP *Reno* yang hanya dapat menangani satu segmen paket data yang hilang sehingga dapat menangani pengiriman ulang paket data hilang lebih dari satu dalam satu *window* tanpa menurunkan nilai *threshold* [5].

RIVERBED dapat digunakan untuk meneliti kinerja jaringan *wired* dan *wireless*. Software ini memiliki kelebihan-kelebihan untuk mendesain jaringan berdasarkan perangkat yang ada di pasaran, protokol, layanan, dan teknologi yang sedang *trend* di dunia telekomunikasi. *Riverbed Modeler* mengakselerasikan *R&D network*, mengurangi *time-to-market*, dan meningkatkan kualitas produk [6].

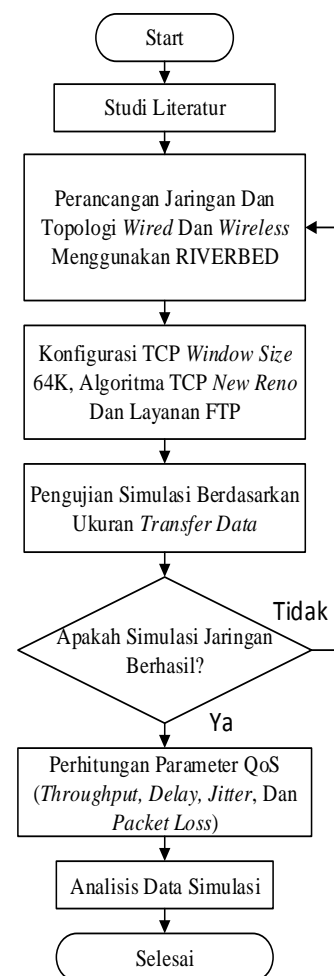
Ada beberapa layanan yang digunakan dalam melakukan komunikasi data. Dalam penelitian ini akan menggunakan layanan komunikasi data yang digunakan salah satunya adalah layanan *File Transfer Protocol* (FTP). Media transmisi yang digunakan yaitu kabel *wired* dan *wireless*. Metode transmisi yang digunakan yaitu TCP *Window size* 64K dengan metode penelitian ini menggunakan algoritma TCP *New Reno*. Simulasi yang akan digunakan untuk melakukan pengujian layanan yaitu software *RIVERBED Modeler* 17.5. Pengujian dilakukan dengan menganalisis beberapa parameter yang digunakan untuk mengetahui QoS jaringan yang

diberikan. Parameter tersebut diantaranya *Throughput*, *Delay*, *Jitter*, dan *Packet Loss*.

Atas dasar permasalahan performansi sistem *wired* dan *wireless*, maka penelitian ini melakukan analisis perbandingan terhadap media transmisi kabel *Wired fast ethernet* dan *Wireless* 802.11n. Analisis perbandingan digunakan untuk mengetahui nilai QoS dari TCP *Window Size* 64K dengan algoritma TCP *New Reno* pada jaringan *Wired Fast Ethernet* dan *Wireless* 802.11N. Maka dari itu penulis mengambil judul “Analisis Unjuk Kerja TCP *Window Size* 64K Menggunakan Algoritma TCP *New Reno* pada Jaringan *Wired* dan *Wireless*”.

2. METODOLOGI PENELITIAN ALUR PENELITIAN

Penelitian dilakukan dengan melakukan beberapa tahap yaitu tahap pengumpulan informasi (Studi Literatur), tahap perancangan sistem, tahap perancangan simulasi, tahap pengumpulan data, dan yang terakhir adalah tahap analisis data simulasi.



Gambar 1 Flowchart diagram alur penelitian.

Pada Gambar 1.1 menunjukkan *flowchart* alur kerja dari penelitian yang akan dilakukan dan menunjukkan langkah-langkah untuk penelitian terhadap perbandingan jaringan *wired* dan *wireless* pada TCP Window Size 64K. Alur kerja dimulai dengan melakukan studi literatur agar dapat membantu proses pada penelitian ini. Selanjutnya, perancangan jaringan 4 user, 8 user, 16 user, dan 32 user menggunakan topologi *star* pada *wired* dan mode infrastruktur pada *wireless* di simulasi *Riverbed* modeler 17.5. Pada tahap selanjutnya yaitu konfigurasi metode transmisi TCP Window Size 64K, metode *congestion control* Algoritma TCP *New Reno* dan Layanan FTP. Pada tahap selanjutnya dilakukan pengujian besar transfer data sebanyak *low load* (25% x Total Bandwidth), *medium load* (50% x Total Bandwidth), dan *high load* (75% x Total Bandwidth) mengetahui nilai parameter-parameter yang akan dianalisis nantinya. Setelah simulasi dapat berjalan dengan baik kemudian tahap selanjutnya melakukan perhitungan parameter QoS (*Throughput*, *Delay*, *Jitter*, dan *Packet Loss*). Pada tahap ini parameter QoS yang dapat diambil dari simulasi *RIVERBED* adalah *delay* dan *packet loss*, sedangkan perhitungan *throughput* dan *jitter* akan dilakukan menggunakan rumus. Kemudian selanjutnya akan dilakukan analisis data simulasi untuk mengetahui perbandingan jaringan *wired* dan *wireless* berdasarkan hasil QoS yang sudah didapatkan

ALAT DAN BAHAN

HARDWARE

Beberapa perangkat *hardware* yang dibutuhkan untuk melakukan konfigurasi terhadap sistem, diantaranya yaitu laptop yang digunakan untuk melakukan proses simulasi dan *software* simulasi yaitu menggunakan *RIVERBED Modeler*.

SOFTWARE

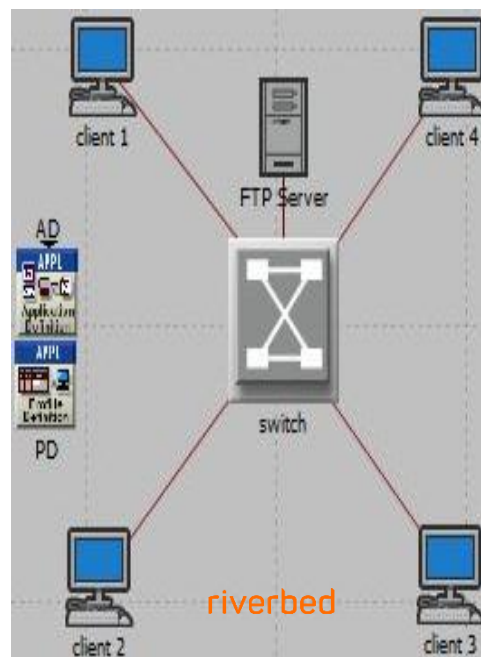
Perangkat lunak yang digunakan, diantaranya yaitu *RIVERBED Modeler* merupakan *tools* simulasi jaringan yang menyediakan Jaringan Virtual Lingkungan dengan model yang seluruh jaringan, termasuk *router*, *switch*, protokol, *server*, dan aplikasi individu. *Software* yang digunakan pada simulasi jaringan ini menggunakan *Software Riverbed Modeler Academic* versi 17.5. *Software* ini digunakan untuk merancang dan membangun simulasi kontrol kepadatan lalu lintas jaringan.

TOPOLOGI JARINGAN

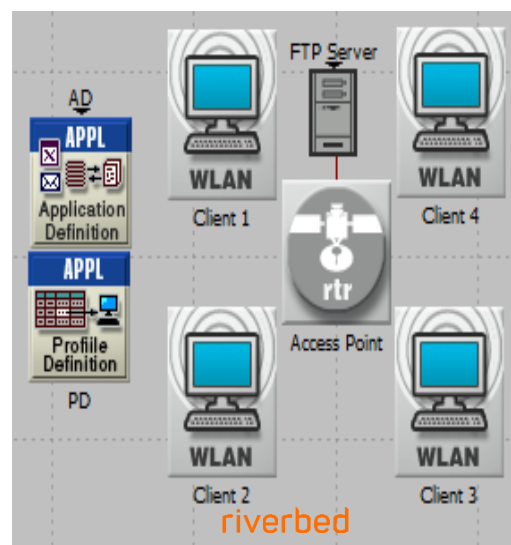
Pada penelitian ini dilakukan pengujian layanan pada FTP menggunakan Algoritma TCP *New Reno*. Pengujian dilakukan dengan menggunakan ukuran TCP Window size 64K pada

jaringan *wired* dan *wireless* sehingga mendapatkan nilai dari parameter QoS. Pengujian tersebut bertujuan untuk mengetahui kualitas jaringan yang diberikan. *Switch* maupun *access point* tersebut dapat menjadi *node* tengah agar jaringan komputer dapat terhubung satu sama lain.

Topologi yang digunakan yaitu menggunakan topologi *star* pada jaringan *wired* seperti yang ditunjukkan pada Gambar 1.2. Topologi *Star* pada jaringan *wired* akan menggunakan *switch*. Mode infrastruktur pada jaringan *wireless* ditunjukkan pada Gambar 1.2. Sedangkan infrastruktur yang digunakan pada jaringan *wireless* adalah *access point*.



Gambar 2. Topologi jaringan penelitian.



Gambar 3 Perancangan topologi jaringan *wireless*.

Pada Gambar 1.2 dan 1.3 terdapat beberapa jenis *node* yang digunakan pada topologi jaringan topologi *star*. *Node* tersebut yaitu *Application Defintion* atau *Application Config*, *Profile Defintion* atau *Profile Config*, dan Komputer Pengguna (*User*). *Application Config* merupakan *node* yang digunakan untuk mengkonfigurasi layanan-layanan yang akan digunakan. Dalam penelitian ini menggunakan layanan *FTP*. Sedangkan *Profile Config* merupakan *node profile* yang digunakan untuk menjalankan layanan yang telah didefinisikan pada *node application* /.

PARAMETER PENELITIAN

PARAMETER KINERJA

Pada penelitian ini penulis telah menentukan beberapa parameter kinerja yang dapat digunakan untuk mengukur unjuk kerja dari layanan aplikasi yang akan diteliti menggunakan *TCP Window Size* yang berbeda dengan Algoritma *TCP Reno*. Parameter kinerja ini menjadi fokus dalam setiap pengujian yang dilakukan sebagai berikut :

a) **Throughput**

Throughput merupakan jumlah total kedatangan paket yang sukses yang diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut. *Throughput* diukur dalam *bps (bit per second)*.

Perhitungan *throughput* dinyatakan sebagai

$$\text{Throughput} = \frac{\text{Jumlah data yang dikirim}}{\text{Waktu pengiriman data}} \quad (3.1)$$

b) **Delay**

Delay yaitu waktu yang dibutuhkan paket untuk mencapai tujuan, karena adanya antrian, atau mengambil rute yang lain untuk menghindari kemacetan yang diukur dalam 's' (*second*). *Delay* dapat dipengaruhi oleh jarak, media fisik, kongesti atau juga waktu proses yang lama.

Perhitungan *delay* dinyatakan sebagai

$$\text{Delay} = \frac{\text{Waktu penerimaan paket} - \text{Waktu pengiriman paket}}{\text{Total Paket yang diterima}} \quad (3.2)$$

c) **Jitter**

Jitter atau variasi *delay*, berhubungan erat dengan *latency*, yang menunjukkan banyaknya variasi *delay* pada taransmisi data di jaringan. *Delay* antrian pada *router* dan *switch* menyebabkan *jitter*. Hal ini diakibatkan oleh variasi-variasi panjang antrian, waktu pengolahan data, dan waktu penghimpunan ulang paket-paket di akhir perjalanan *jitter*.

Perhitungan *jitter* dinyatakan sebagai

$$\text{Jitter} = \frac{\text{Total variasi Delay}}{\text{Total Paket diterima}} \quad (3.3)$$

d) **Packet Loss**

Packet Loss merupakan suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang, dapat terjadi karena *collision* dan *congestion* pada jaringan.

Perhitungan *Packet Loss* dinyatakan sebagai [19]

$$\text{Packet Loss} = \frac{(\text{Packet data dikirim} - \text{packet data diterima})}{\text{Packet data dikirim}} \times 100. \quad (3.4)$$

2.5 PARAMETER PENGUJIAN

Pada penelitan ini, penulis menentukan parameter yang digunakan dalam melakukan pengujian agar hasil data yang didapatkan sesuai dengan yang diharapkan. Tabel 1.1 menunjukkan parameter pengujian yang digunakan untuk penelitian ini

Tabel 1. Parameter pengujian.

Parameter	Ukuran	Load
Wired Fast Ethernet	100Mbps	Load (25% x Total Bandwidth)
		Load (50% x Total Bandwidth)
		Load (75% x Total Bandwidth)
Wireless 802.11n	600Mbps	Load (25% x Total Bandwidth)
		Load (50% x Total Bandwidth)
		Load (75% x Total Bandwidth)

SKENARIO PENELITIAN

Mengacu pada topologi jaringan yang dirancang, pengujian sistem dibuat menjadi empat skenario yaitu menggunakan 4 *client*, 8 *client*, 16 *client*, dan 32 *client* pada layanan aplikasi *FTP* dengan besar *transfer data* pada setiap skenario masing-masing sebanyak (25% x *Total Bandwidth*), (50% x *Total Bandwidth*), dan (75% x *Total Bandwidth*) seperti pada Tabel 3.3. Pengujian ini dilakukan dengan menempatkan media transmisi yang berbeda, yang bertujuan untuk mengetahui kualitas jaringan yang diberikan selama proses simulasi. Hasil keluaran parameter *QoS* dari 4 skenario yang dirancang, dapat dianalisa untuk perbandingan kualitas pada jaringan *wired* dan *wireless* yang telah

dikonfigurasi TCP *Window Size* 64K dengan Algoritma TCP *New Reno* untuk melakukan *congestion control* yang dilakukan pada layanan FTP. Untuk mempermudah pemahaman mengenai skenario yang dijalankan pada pengujian jaringan ditunjukkan pada Tabel 1.2.

Tabel 2 Skenario pengujian sistem.

Skenario	Jumlah User	Besar Transfer Data	Parameter QoS	TCP Window Size	Media Transmisi	Layanan
Skenario 1	4 user	25% x Total Bandwidth	Throughput Delay Jitter dan Packet Loss	64K	Kabel (Wired) dan Nirkabel (Wireless)	FTP
		50% x Total Bandwidth				
		75% x Total Bandwidth				
Skenario 2	8 user	25% x Total Bandwidth	Throughput Delay Jitter dan Packet Loss	64K	Kabel (Wired) dan Nirkabel (Wireless)	FTP
		50% x Total Bandwidth				
		75% x Total Bandwidth				
Skenario 3	16 user	25% x Total Bandwidth	Throughput Delay Jitter dan Packet Loss	64K	Kabel (Wired) dan Nirkabel (Wireless)	FTP
		50% x Total Bandwidth				
		75% x Total Bandwidth				
		25% x Total				

Skenario 4	32 user	Bandwidth				
		50% x Total Bandwidth				
		75% x Total Bandwidth				

3. HASIL DAN ANALISA HASIL SIMULASI DAN ANALISA

Simulasi jaringan pada penelitian ini menggunakan *software RIVERBED Modeler Academic* versi 17.5 untuk melakukan pengujian pada jaringan *wired* dan *wireless*. Hasil simulasi yang akan dianalisa terdiri dari 4 skenario. Dimulai dari Skenario 4 *Client*, Skenario 8 *Client*, Skenario 16 *Client*, dan Skenario 32 *Client*. Dari masing-masing skenario tersebut akan dilakukan pengujian dengan mengirimkan besar *file transfer data* sebesar 25% x *total bandwidth*, 50% x *total bandwidth*, dan 75% x *total bandwidth*. Jaringan yang dibuat dalam skenario adalah dengan menggunakan 4 *client*, 8 *client*, 16 *client*, dan 32 *client* dan 1 buah *server* untuk layanan *File Transfer Protocol* (FTP). Kemudian pada jaringan *wired* akan ditambahkan sebuah *switch* untuk menghubungkan antara *client* dan *server* menggunakan kabel *fast ethernet*. Sedangkan pada jaringan *wireless* akan ditambahkan sebuah *access point* untuk menghubungkan dari *server* menggunakan kabel *fast ethernet* ke *access point*, kemudian dari *access point* akan dihubungkan ke *client* menggunakan *wireless 802.11n*.

Besar *transfer data* 25% x *total bandwidth* diperoleh dengan mencari 25% dari *total bandwidth* pada jaringan *wired fast ethernet* dan *wireless 802.11n* lalu dibagi jumlah *client*. Pada Skenario 4 *Client* diperoleh nilai *transfer data* setiap *client* yaitu sebesar 819.200 *Byte* pada jaringan *wired fast ethernet*. Sedangkan nilai *transfer data* pada jaringan *wireless 802.11n* adalah sebesar 495.200 *Byte*. Pada Skenario 8 *Client* diperoleh nilai *transfer data* setiap *client* sebesar 409.600 *Byte* dan pada jaringan *wireless* sebesar 2.457.600 *Byte*. Kemudian pada Skenario 16 *Client* diperoleh nilai *transfer data* setiap *client* sebesar 204.800 *Byte* dan pada jaringan *wireless* sebesar 1.228.800 *Byte*. Pada Skenario 32 *Client* diperoleh nilai *transfer data* setiap *client* sebesar 102.400 *Byte* dan pada jaringan *wireless* sebesar 614.400 *Byte*.

Besar *transfer data* 50% x *total bandwidth* diperoleh dengan mencari 50% dari *total bandwidth* pada jaringan *wired fast ethernet* dan *wireless 802.11n* lalu dibagi jumlah *client*. Pada Skenario 4 *Client* diperoleh nilai *transfer data* setiap *client* yaitu sebesar 1.638.400 *Byte* pada jaringan *wired fast ethernet*. Sedangkan nilai *transfer data* pada jaringan *wireless 802.11n* adalah sebesar 9.830.400 *Byte*. Pada Skenario 8 *Client* diperoleh nilai *transfer data* setiap *client* sebesar 819.200 *Byte* dan pada jaringan *wireless* sebesar 4.915.200 *Byte*. Kemudian pada Skenario 16 *Client* diperoleh nilai *transfer data* setiap *client* sebesar 409.600 *Byte* dan pada jaringan *wireless* sebesar 2.457.600 *Byte*. Pada Skenario 32 *Client* diperoleh nilai *transfer data* setiap *client* sebesar 204.800 *Byte* dan pada jaringan *wireless* sebesar 1.228.800 *Byte*.

Besar *transfer data* 75% x *total bandwidth* diperoleh dengan mencari 75% dari *total bandwidth* pada jaringan *wired fast ethernet* dan *wireless 802.11n* lalu dibagi jumlah *client*. Pada Skenario 4 *Client* diperoleh nilai *transfer data* setiap *client* yaitu sebesar 2.457.600 *Byte* pada jaringan *wired fast ethernet*. Sedangkan nilai *transfer data* pada jaringan *wireless 802.11n* adalah sebesar 14.745.600 *Byte*. Pada Skenario 8 *Client* diperoleh nilai *transfer data* setiap *client* sebesar 1.228.800 *Byte* dan pada jaringan *wireless* sebesar 7.372.800 *Byte*. Kemudian pada Skenario 16 *Client* diperoleh nilai *transfer data* setiap *client* sebesar 614.400 *Byte* dan pada jaringan *wireless* sebesar 3.686.400 *Byte*. Pada Skenario 32 *Client* diperoleh nilai *transfer data* setiap *client* sebesar 307.200 *Byte* dan pada jaringan *wireless* sebesar 1.843.200 *Byte*.

Hasil yang didapatkan dari simulasi berupa grafik dalam bentuk linear. Kemudian akan dilakukan perbandingan parameter *throughput*, *delay*, *jitter*, dan *packet loss* pada jaringan *wired* dan *wireless* dari masing-masing skenario. Grafik tersebut ditampilkan dengan perbedaan warna yang masing-masing warna merepresentasikan hasil output parameter *client*. Analisa perbandingan ini akan lebih fokus pada hasil output parameter *Quality of Service (QoS)* yang memperlihatkan perbedaan kualitas performa layanan FTP pada jaringan *wired* dan *wireless* menggunakan *Transmission Control Protocol Window Size 64K*, juga mekanisme *congestion control* yang ada pada *Transmission Control Protocol New Reno*.

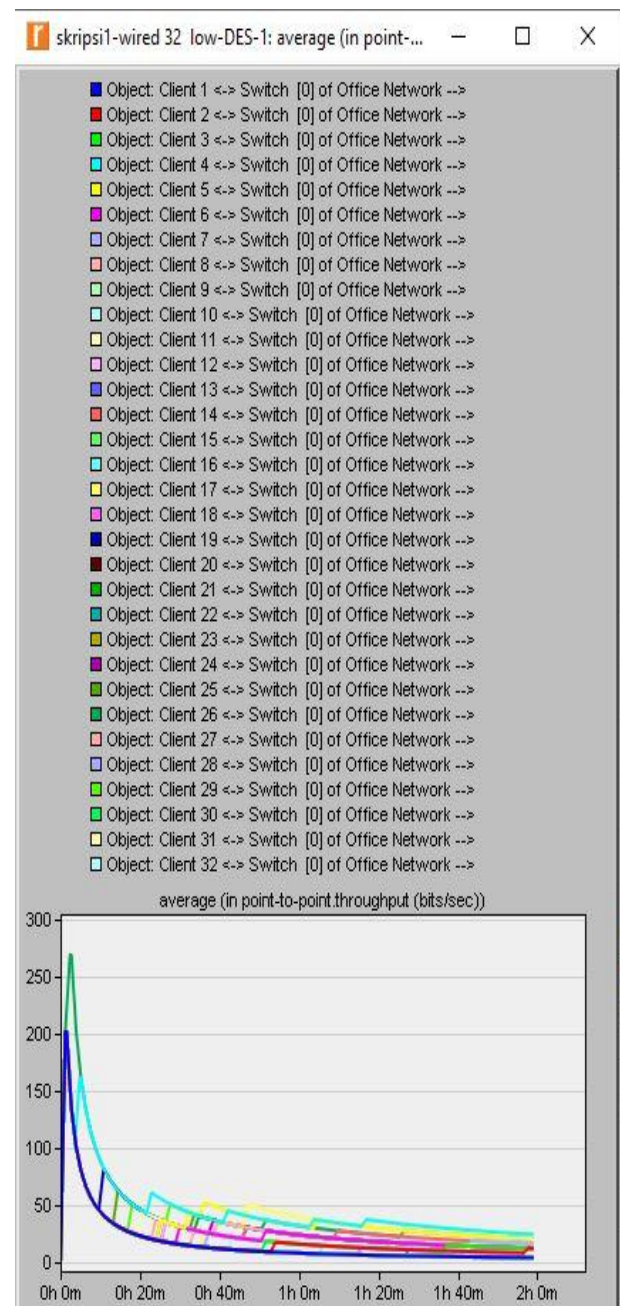
PARAMETER THROUGHPUT

Throughput merupakan jumlah berapa banyak data yang dapat ditransfer dari sumber ke tujuannya dalam jangka waktu tertentu.

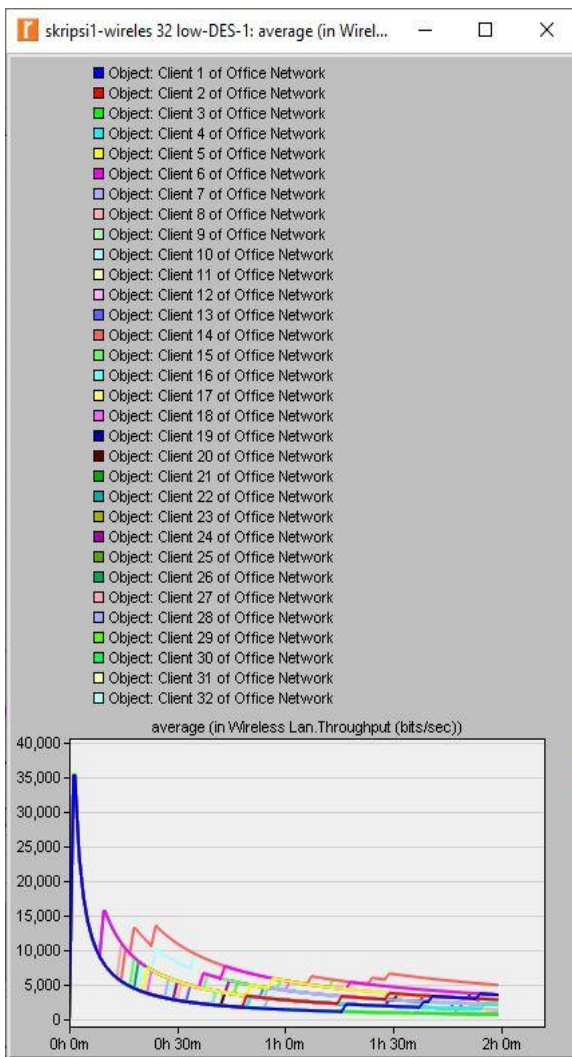
Throughput mengukur tingkat keberhasilan pesan sampai di tujuannya. Semakin besar nilai *throughput*, maka akan semakin baik. Besarnya *throughput* akan mempengaruhi kualitas kinerja jaringan tersebut.

SKENARIO 32 CLIENT

Skenario 32 *client* dilakukan pada 32 *client* dengan besar *transfer data* yang berbeda-beda pada jaringan *wired fast ethernet* dan *wireless 802.11n*. Besar *transfer data* yang dilakukan yaitu 25% x *Total Bandwidth*, 50% x *Total Bandwidth*, dan 75% x *Total Bandwidth*.



(a)



(b)

Gambar 4 Perbandingan throughput (a) transfer data 75% dari total bandwidth fast ethernet; (b) transfer data 75% dari total bandwidth wireless 802.11n.

Pada Gambar 4 (a) dan (b) menunjukkan perbedaan grafik nilai *throughput* yang dihasilkan dari *software* simulasi *riverbed* pada jaringan *fast ethernet* dan *wireless 802.11n*. Pada sumbu x mewakili duration (*hour*) sedangkan pada sumbu y mewakili nilai *throughput* (*bit/sec*).

Pada Gambar 4 terdapat *client* yang menampilkan mekanisme *congestion* dan ada yang tidak. Mekanisme *TCP New Reno* dalam melakukan *congestion control* adalah dengan adanya fase *slow start*, fase *fast retransmit*, fase *fast recovery* dan *congestion avoidance*. Adapun *client* yang tidak memiliki *congestion* hanya mengalami fase *slow start* saja. Fase *slow start* terjadi dengan adanya kenaikan nilai *throughput* dimana *TCP* mencoba mengetahui kapasitas ruang

buffer yang dimiliki oleh penerima. Selanjutnya pada fase *fast retransmit* dan fase *fast recovery*. Fase ini terjadi dengan adanya penurunan nilai *throughput* yang disebabkan adanya *packet drop* yang terjadi saat pengiriman *packet*. Selanjutnya pada fase *congestion avoidance*. Fase ini terjadi dengan adanya kenaikan nilai *throughput* secara linier dimana *TCP* berusaha menghindari kemacetan saat proses pengiriman paket.

Berdasarkan pada gambar tersebut, *average* kejadian fase *slow start* pada simulasi 32 *client* dengan *transfer data 75%* dari *total bandwidth* terjadi pada detik 0-144 *sec* pada jaringan *fast ethernet* dengan nilai *throughput* 531.366 *bits/sec* dan terjadi pada detik 0-72 *sec* pada jaringan *wireless 802.11n* dengan nilai *throughput* 116145 *bits/sec*.

Setelah mengalami fase *slow start*, kemudian *TCP* akan menurunkan jumlah *packet* yang dikirimkan untuk menghindari *congestion*. Setelah itu, Jika *packet* terus turun hingga simulasi berakhir, maka *client* tidak mengalami *congestion*. Adapun jika terjadi *congestion*, maka terjadi fase *retransmit* dan fase *fast recovery*. *Average* kejadian fase *retransmit* dan fase *fast recovery* pada simulasi 32 *client* dengan *transfer data 75%* dari *total bandwidth* terjadi pada detik 360-5328 *sec* pada jaringan *fast ethernet* dengan nilai *throughput* 49.4757 *bits/sec* dan terjadi pada detik 216-6408 *sec* pada jaringan *wireless 802.11n* dengan nilai *throughput* 14781.5 *bits/sec*.

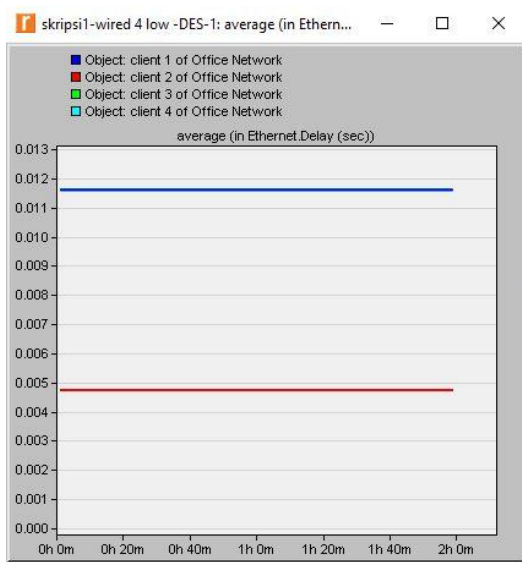
Setelah mengalami fase *retransmit* dan fase *fast recovery*, kemudian *TCP* akan menaikkan jumlah *packet* yang dikirimkan untuk menghindari *packet loss* yang disebut fase *congestion avoidance*. *Average* kejadian fase *congestion avoidance* pada simulasi 32 *client* dengan *transfer data 75%* dari *total bandwidth* terjadi pada detik 504-5400 *sec* pada jaringan *fast ethernet* dengan nilai *throughput* 94.8989 *bits/sec* dan terjadi pada detik 288-6480 *sec* pada jaringan *wireless 802.11n* dengan nilai *throughput* 23899.6 *bits/sec*. Selanjutnya, *packet* akan kembali turun hingga simulasi berakhir.

PARAMETER DELAY

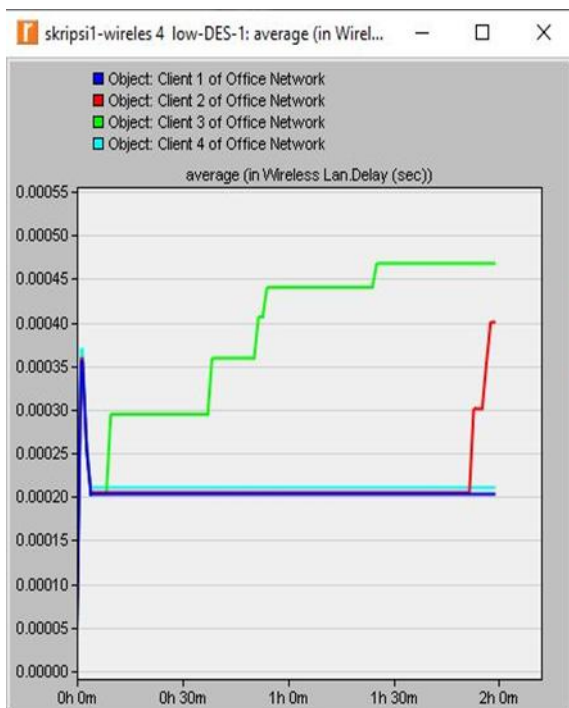
Delay merupakan total waktu yang dilalui suatu paket dari pengirim ke penerima melalui jaringan. Jumlah waktu yang dibutuhkan untuk menunggu tergantung pada besaran antrian. Jika antrian kosong, maka akan ditransmisikan segera, tetapi jika paket tersebut berada di belakang paket lain, maka perlu menunggu paket di depan untuk ditransmisikan terlebih dahulu. Semakin kecil nilai *delay*, maka akan semakin baik kualitas jaringan tersebut.

SKENARIO 4 CLIENT

Skenario 4 *client* dilakukan pada 4 *client* dengan besar *transfer data* yang berbeda-beda pada jaringan *wired fast ethernet* dan *wireless 802.11n*. Besar *transfer data* yang dilakukan yaitu 25% x *Total Bandwidth*, 50% x *Total Bandwidth*, dan 75% x *Total Bandwidth*.



(a)



(b)

Gambar 5 Perbandingan delay (a) transfer data 25% dari total bandwidth fast ethernet; (b) transfer data 25% dari total bandwidth wireless 802.11n;

Pada Gambar 5 (a) dan (b) menunjukkan perbedaan grafik nilai *delay* yang dihasilkan dari *software* simulasi *riverbed* pada jaringan *fast ethernet* dan *wireless 802.11n*. Pada sumbu x mewakili *duration (hour)* sedangkan pada sumbu y mewakili nilai *delay (sec)*.

Pada Gambar 5 menampilkan mekanisme *congestion control* diantaranya : *slow start*, *fast retransmit*, *fast recovery* dan *congestion avoidance*. *Average* kejadian fase *slow start* pada simulasi 4 *client* dengan *transfer data* 25% dari *total bandwidth* terjadi pada detik 0-72 *sec* pada jaringan *fast ethernet* dengan nilai *delay* 0.009887067 *sec* dan jaringan *wireless 802.11n* dengan nilai *delay* 0.000360804 *sec*.

Average kejadian fase *fast retransmit* dan fase *fast recovery* pada simulasi 4 *client* dengan *transfer data* 25% dari *total bandwidth* terjadi pada detik 2088-6840 *sec* pada jaringan *fast ethernet* dengan nilai *delay* 0.011603119 *sec* dan terjadi pada detik 504-6696 *sec* pada jaringan *wireless 802.11n* dengan nilai *delay* 0.000204209 *sec*.

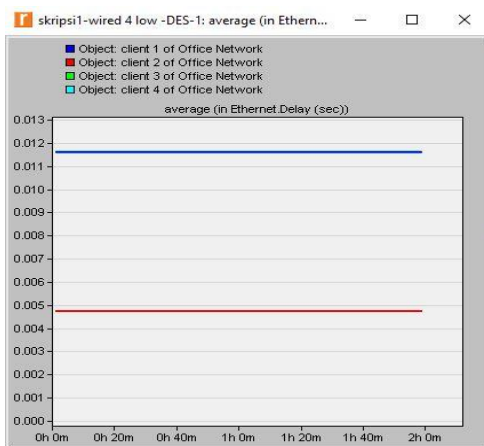
Average kejadian fase *congestion avoidance* pada simulasi 4 *client* dengan *transfer data* 25% dari *total bandwidth* terjadi pada detik 2160-6912 *sec* pada jaringan *fast ethernet* dengan nilai *delay* 0.011602978 *sec* dan terjadi pada detik 576-6768 *sec* pada jaringan *wireless 802.11n* dengan nilai *delay* 0.000297695 *sec*.

PARAMETER JITTER

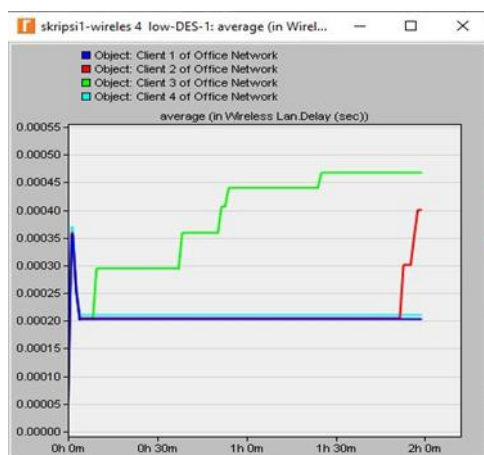
Jitter merupakan variasi dari *delay* yang diakibatkan oleh variasi-variasi dalam panjang antrian, dalam waktu pengolahan data, dan juga dalam waktu penghimpunan ulang paket-paket di akhir perjalanan *jitter*. *Jitter* yang terjadi mendekati nol maka kecepatan jaringan tersebut dikatakan baik. Namun sebaliknya jika tidak mendekati nol maka kecepatan pengiriman data dapat dikatakan kurang baik dan akan terjadi kehilangan data dalam proses pengiriman data/*packet loss*.

SKENARIO 4 CLIENT

Skenario 4 *client* dilakukan pada 4 *client* dengan besar *transfer data* yang berbeda-beda pada jaringan *wired fast ethernet* dan *wireless 802.11n*. Besar *transfer data* yang dilakukan yaitu 25% x *Total Bandwidth*, 50% x *Total Bandwidth*, dan 75% x *Total Bandwidth*.



(a)



(b)

Gambar 6 Perbandingan Jitter (a) Transfer Data 75% dari Total Bandwidth Fast Ethernet; (b) Transfer Data 75% dari Total Bandwidth Wireless 802.11n.

Pada Gambar 6 (a) dan (b) menunjukkan perbedaan grafik nilai *jitter* yang dihasilkan dari *software* simulasi *riverbed* pada jaringan *fast ethernet* dan *wireless 802.11n*. Pada sumbu x mewakili duration (*hour*) sedangkan pada sumbu y mewakili nilai *jitter* (*sec*).

Pada Gambar 6 menampilkan mekanisme congestion control diantaranya : slow start, fast retransmit, fast recovery dan congestion avoidance.

Average kejadian fase *slow start* pada simulasi 4 *client* dengan *transfer data* 75% dari *total bandwidth* terjadi pada detik 0-72 *sec* pada jaringan *fast ethernet* dengan nilai *jitter* 0 *sec* dan terjadi pada detik 0-144 *sec* pada jaringan *wireless 802.11n* dengan nilai *jitter* 0.000103817 *sec*.

Average kejadian fase *fast retransmit* dan fase *fast recovery* pada simulasi 4 *client* dengan *transfer data* 75% dari *total bandwidth* terjadi pada

detik 1800-5832 *sec* pada jaringan *fast ethernet* dengan nilai *jitter* 0.00000000002 *sec* dan terjadi pada detik 288-6768 *sec* pada jaringan *wireless 802.11n* dengan nilai *jitter* 0.000078 *sec*.

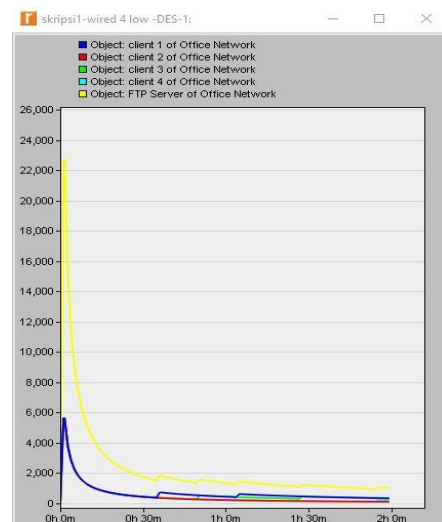
Average kejadian fase *congestion avoidance* pada simulasi 4 *client* dengan *transfer data* 75% dari *total bandwidth* terjadi pada detik 1872-5904 *sec* pada jaringan *fast ethernet* dengan nilai *jitter* 0 *sec* dan terjadi pada detik 360-6840 *sec* pada jaringan *wireless 802.11n* dengan nilai *jitter* 0 *sec*.

PARAMETER PACKET LOSS

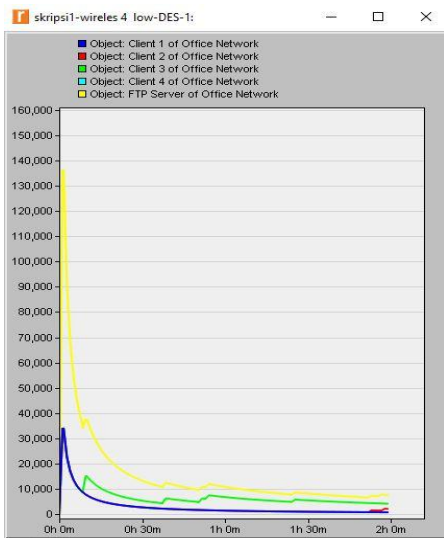
Packet loss adalah besar kegagalan saat mentransmisikan suatu paket data kepada alamat tujuan yang menyebabkan hilangnya beberapa data dalam proses pengiriman. *Packet loss* biasanya diberi satuan dalam bentuk persen (%) untuk menandai berapa persen jumlah paket yang tidak dapat dikirimkan dari total keseluruhan semua paket. Semakin kecil jumlah *packet loss* yang didapatkan maka semakin baik pula jaringan tersebut untuk digunakan.

SKENARIO 4 CLIENT

Skenario 4 *client* dilakukan pada 4 *client* dengan besar *transfer data* yang berbeda-beda pada jaringan *wired fast ethernet* dan *wireless 802.11n*. Besar *transfer data* yang dilakukan yaitu 25% x *Total Bandwidth*, 50% x *Total Bandwidth*, dan 75% x *Total Bandwidth*.



(a)



(b)

Gambar 7. Perbandingan packet loss ftp (a) transfer data 75% dari total bandwidth fast ethernet; (b) transfer data 75% dari total bandwidth wireless 802.11n.

Pada Gambar 7. (a) dan (b) menunjukkan perbedaan grafik nilai *traffic sent* dan *traffic received* dari *software* simulasi *riverbed* pada jaringan *fast ethernet* dan *wireless 802.11n*. Pada sumbu x mewakili duration (*hour*) sedangkan pada sumbu y mewakili nilai *packet loss* (%).

Pada Gambar 7. menampilkan mekanisme adanya *packet loss* pada grafik yang dimiliki oleh penerima. menunjukkan perbedaan nilai *packet loss* yang dihasilkan pada jaringan *fast ethernet* dan *wireless 802.11n*. *Average* nilai *packet loss* pada simulasi 4 *client* dengan transfer data 75% dari *total bandwidth* adalah 0% pada jaringan *fast ethernet* dan 0.465909091% pada jaringan *wireless 802.11n*.

PERBANDINGAN HASIL ANALISA DAN SIMULASI

Berikut ini merupakan perbedaan hasil keseluruhan *average* dari simulasi yang diteliti dapat dilihat pada tabel 4 berupa nilai *average throughput, delay, jitter* dan *packet loss*. Pada keseluruhan nilai *average throughput* dan *delay, wireless 802.11n* memiliki performa yang lebih baik daripada jaringan *wired fast ethernet*. Namun pada keseluruhan nilai *jitter* dan *packet loss, wired fast ethernet* lebih baik daripada *wireless 802.11n*. Adapun secara keseluruhan hasil simulasi, jaringan *wireless 802.11n* memiliki *delay* yang lebih cepat daripada *fast ethernet*. Namun *fast ethernet* memiliki nilai *packet loss* yang lebih rendah daripada *wireless 802.11n*. Hal itu dikarenakan *wireless 802.11n* memiliki media transmisi yang

secara bebas dikirimkan melalui media udara. Hal itu menyebabkan *packet data* dapat dikirimkan lebih cepat daripada jaringan *fast ethernet*, namun nilai *delay*-nya tidak stabil, sehingga rentan terjadinya tabrakan data maupun gangguan yang dapat menyebabkan *packet loss*. Sedangkan *wired fast ethernet* memiliki media transmisi yang dikirimkan secara terarah melalui media kabel, sehingga pesan yang dikirimkan jarang terjadi *packet loss*. Hal itu menyebabkan *packet data* yang dikirimkan membutuhkan *delay* yang lebih lama dari *wireless 802.11n*, namun nilai *delay*-nya lebih stabil.

Tabel 4. Perbandingan hasil analisa dan simulasi.

No	Parameter	Jumlah Client	Besar Transfer Data	Media Transmisi	Nilai Average
1.	Throughput (bit/s)	4	25% x Total Bandwidth	Fast Ethernet	4743.51
				802.11n	31695
			50% x Total Bandwidth	Fast Ethernet	9950.45
				802.11n	70442.9
			75% x Total Bandwidth	Fast Ethernet	16685.2
				802.11n	117920
		8	25% x Total Bandwidth	Fast Ethernet	73.1374
				802.11n	18077.5
			50% x Total Bandwidth	Fast Ethernet	135.755
				802.11n	35183.5
			75% x Total Bandwidth	Fast Ethernet	683.041
				802.11n	53365
16	25% x Total Bandwidth	Fast Ethernet	41.4668		
		802.11n	8662.02		
	50% x Total Bandwidth	Fast Ethernet	81.83		
		802.11n	17883.5		
	75% x Total Bandwidth	Fast Ethernet	127.784		
		802.11n	23250.1		
32	25% x Total Bandwidth	Fast Ethernet	23.8764		
		802.11n	4276.78		
	50% x Total Bandwidth	Fast Ethernet	44.4094		
		802.11n	8518.71522		

2.	Delay (sec)	4	75% x Total Bandwid th	Fast Ethernet	63.2686
				802.11n	13050.6
			25% x Total Bandwid th	Fast Ethernet	0.00989
				802.11n	0.00025 3344
		50% x Total Bandwid th	Fast Ethernet	0.01545 84	
			802.11n	0.00027	
		75% x Total Bandwid th	Fast Ethernet	0.03494	
			802.11n	0.00033	
		8	25% x Total Bandwid th	Fast Ethernet	0.0057
				802.11n	0.00029 9773
			50% x Total Bandwid th	Fast Ethernet	0.0101
				802.11n	0.00030 0879
			75% x Total Bandwid th	Fast Ethernet	0.01131
				802.11n	0.00029 9326
		16	25% x Total Bandwid th	Fast Ethernet	0.00289
				802.11n	0.00029 0379
			50% x Total Bandwid th	Fast Ethernet	0.00569
				802.11n	0.00028 1857
			75% x Total Bandwid th	Fast Ethernet	0.00533
				802.11n	0.00026 359
32	25% x Total Bandwid th	Fast Ethernet	0.00145		
		802.11n	0.00028 8424		
	50% x Total Bandwid th	Fast Ethernet	0.0029		
		802.11n	0.00028 552		
	75% x Total Bandwid th	Fast Ethernet	0.00433		
		802.11n	0.00028 1482		
3.	Jitter (sec)	4	25% x Total Bandwid th	Fast Ethernet	0.00000 00014
				802.11n	0.00000 59
			50% x Total Bandwid th	Fast Ethernet	0.00004 66
				802.11n	0.00000 59
			75% x Total Bandwid th	Fast Ethernet	0.00000 000000 03
				802.11n	0.00000 63

4.	Packet Loss (%)	8	25% x Total Bandwid th	Fast Ethernet	0.00000 111		
				802.11n	0.00000 61		
			50% x Total Bandwid th	Fast Ethernet	0.00001 95		
				802.11n	0.00000 64		
			75% x Total Bandwid th	Fast Ethernet	0.00002 46		
				802.11n	0.00000 62		
		16	25% x Total Bandwid th	Fast Ethernet	0.00000 0027		
				802.11n	0.00000 61		
			50% x Total Bandwid th	Fast Ethernet	0.00000 195		
				802.11n	0.00000 61		
			75% x Total Bandwid th	Fast Ethernet	0.00001 35		
				802.11n	0.00000 58		
		32	25% x Total Bandwid th	Fast Ethernet	0.00000 0084		
				802.11n	0.00000 63		
			50% x Total Bandwid th	Fast Ethernet	0.00000 02		
				802.11n	0.00000 62		
			75% x Total Bandwid th	Fast Ethernet	0.00000 102		
				802.11n	0.00000 62		
		4.	Packet Loss (%)	4	25% x Total Bandwid th	Fast Ethernet	0
						802.11n	0.09090 9091
50% x Total Bandwid th	Fast Ethernet				0		
	802.11n				0.11111 1111		
75% x Total Bandwid th	Fast Ethernet				0		
	802.11n				0.46590 9091		
8	25% x Total Bandwid th			Fast Ethernet	0		
				802.11n	0		
	50% x Total Bandwid th			Fast Ethernet	0		
				802.11n	0.19903 3816		
	75% x Total Bandwid th			Fast Ethernet	0		
				802.11n	0.125		
16	25% x Total Bandwid th	Fast Ethernet	0				
		802.11n	0.08513 9319				
	50% x Total	Fast	0				
		Ethernet	0				

		<i>Bandwidth</i>	802.11n	0
		75% x <i>Total Bandwidth</i>	<i>Fast Ethernet</i>	0
			802.11n	0
	32	25% x <i>Total Bandwidth</i>	<i>Fast Ethernet</i>	0
			802.11n	0.02900 988
		50% x <i>Total Bandwidth</i>	<i>Fast Ethernet</i>	0
			802.11n	0.03592 0852
		75% x <i>Total Bandwidth</i>	<i>Fast Ethernet</i>	0
			802.11n	0.05216 8022

4. KESIMPULAN

Dari hasil simulasi dan analisa yang telah dilakukan maka dapat disimpulkan sebagai berikut :

Berdasarkan perbandingan skenario yang telah dilakukan, semakin banyak jumlah *client* yang melakukan *transfer data* dengan jumlah besar, maka nilai *throughput* akan semakin besar. Jaringan *Wireless* 802.11n dengan jumlah 32 *client* dan besar *transfer data* 75% dari *total bandwidth* memiliki skenario terbaik pada parameter *throughput* . Semakin sedikit jumlah *client* yang melakukan *transfer data* dengan jumlah kecil, maka nilai *delay* akan semakin kecil. Jaringan *Wireless* 802.11n dengan jumlah 4 *client* dan besar *transfer data* 25% dari *total bandwidth* memiliki skenario terbaik pada parameter *delay*. Semakin sedikit jumlah *client* yang melakukan *transfer data* dengan jumlah besar, maka nilai *jitter* akan semakin mendekati angka 0 *sec*. Jaringan *Wireless* 802.11n dengan jumlah 4 *client* dan besar *transfer data* 75% x *total bandwidth* memiliki skenario terbaik pada parameter *jitter*.

TCP *Window Size* 64k menggunakan TCP *New Reno* pada jaringan *Wired Fast Ethernet* mempunyai kabel yang melindungi data saat di kirim dari FTP *server* ke *client* sehingga mengalami jaringan yang lebih stabil, selain itu juga jarang mengalami *packet loss*, namun lebih lambat dalam waktu pengiriman datanya daripada jaringan *Wireless* 802.11n. Hal ini terjadi pada hasil simulasi dan analisa yang telah dilakukan, dimana nilai *average jitter* terbaik dengan nilai *average* 0,0000000000003 *sec*. *Packet loss* terbaik pada keseluruhan skenario pada jaringan *Wired Fast Ethernet* dengan nilai *average packet loss* 0%.

TCP *Window Size* 64k menggunakan TCP *New Reno* pada jaringan *Wireless* 802.11n menggunakan gelombang udara dalam proses transmisi data, sehingga jaringan dapat dikirim dan

diterima lebih cepat, selain itu juga *Wireless* 802.11n mempunyai *bandwidth* yang lebih besar daripada *Wired Fast Ethernet* sehingga jaringan yang dikirim dan diterima lebih besar daripada *Wired Fast Ethernet*, namun lebih sering terjadi paket yang hilang daripada jaringan *Wired Fast Ethernet*. Hal ini terjadi pada hasil simulasi dan analisa yang telah dilakukan dimana nilai *average throughput* terbaik dengan nilai *average* 13050,6 *bit/sec*, adapun *delay* terbaik sebesar 0,000253 *sec*.

Berdasarkan pengaruh *congestion control* pada algoritma TCP *New Reno*, jika terjadi multi paket drop maka algoritma ini akan bertahan di fase *fast recovery*. Sehingga nilai *throughput* yang dihasilkan tetap tinggi. Kemudian, Algoritma TCP *New Reno* akan memperkecil *delay* dari yang seharusnya agar lebih cepat terdeteksi adanya *packet loss* dan cepat untuk dikirimkan kembali kepada *client* (*fast retransmit*).

5. SARAN

Beberapa saran untuk perkembangan bagi penelitian selanjutnya seperti yaitu penelitian selanjutnya dapat dilakukan dengan menggunakan layanan HTTP, *Email*, *Remote login* dan *DB Query*. Penelitian selanjutnya dapat dilakukan dengan menggunakan topologi lain, seperti topologi *tree*, *mesh*, maupun *bus* pada jaringan *wired*. Penelitian selanjutnya dapat dilakukan dengan menggunakan jumlah *transfer data* sebesar 100% dari besar *bandwidth*.

REFERENSI

- [1] T. Mahardian Bangkit Sugiri, "Analisis Perbandingan Unjuk Kerja TCP Reno dan TCP Vegas pada Jaringan Kabel," 2016.
- [2] A. Siswanto, "Evaluasi Kinerja Wireless 802.11N untuk E Learning," *ITJRD*, vol. 1, no. 2, hlm. 13–25, Feb 2017, doi: 10.25299/itjrd.2017.vol1(2).557.
- [3] K. R. A. Manibuy, "Analisis Perbandingan Unjuk Kerja TCP Tahoe dan TCP New Reno pada Jaringan Wired dan Wireless," 2017.
- [4] A. Samuel Luceano Basylea Siahaan, "Analisis Perbandingan TCP Window Size Menggunakan Algoritma TCP Reno pada Jaringan Kabel," 2020.
- [5] H. Idwan, T. Y. Arif, dan R. Munadi, "Analisis Round Trip Time (RTT) Terhadap Kinerja Jaringan Wireless TCP New Reno," hlm. 9, 2018.

- [6] "OPNET Network Simulator," *Opnet Projects*. <http://opnetprojects.com/opnet-network-simulator/>.
- [7] M. K. Hanif, S. M. Aamir, R. Talib, dan Y. Saeed, "Analysis of Network Traffic Congestion Control over TCP Protocol," hlm. 10, 2017.
- [8] H. Riyadi, "Kabel Jaringan: Pengertian, Fungsi, Kelebihan dan Kekurangannya," *Nesabamedia*, Sep 14, 2017. <https://www.nesabamedia.com/pengertian-kabel-jaringan/>
- [9] A. T. Harfad, S. R. Akbar, dan A. Bhawiyuga, "Analisa Kinerja Algoritma TCP Congestion Control Cubic, Reno, Vegas Dan Westwood+," hlm. 10.
- [10] Y. H. Putra, "Simulasi dan Analisa Pengaruh TCP Windowing pada Transport Layer Terhadap Peningkatan Kinerja Jaringan Berbasis Virtual Private Network (VPN) Menggunakan Simulator OPNET," hlm. 87, 2010.
- [11] Anonym, "Mikrotik.ID : TCP/IP (Bagian - 2) : Protokol." http://mikrotik.co.id/artikel_lihat.php?id=62 (diakses Des 22, 2019).
- [12] G. A. Ryandy, "Analisis Perbandingan Unjuk Kerja Algoritma TCP New Reno dengan TCP Westwood+ Untuk Mereduksi Kongesti pada Jaringan WLAN."
- [13] Y. T. Nugroho, "Analisis Unjuk Kerja TCP pada Finite Buffer dan Infinite Buffer," 2018.
- [14] M. Ibnu R. Febriansah, "Analisis Bottleneck dan Bufferbloat pada AQM Droptail, RED dan SFQ di Komunikasi Data TCP Newreno," *JR*, vol. 2, no. 9, Agu 2020, doi: 10.22219/repositor.v2i9.748.
- [15] M. Taruk dan H. J. Setyadi, "Analisis Mekanisme Penanganan Kemacetan (Congestion Control) pada Algoritma Varian Protokol TCP," hlm. 5, 2016.
- [16] E. Prasetyo, A. Hamzah, dan E. Sutanta, "Analisa Quality of Service (QOS) Kinerja Point to Point Protocol Over Ethernet (PPPOE) dan Point to Point Tunneling Protocol (PPTP)," vol. 4, no. 1, hlm. 10, 2016.
- [17] D. Ruwaida dan D. Kurnia, "Rancang Bangun File Transfer Protocol (FTP) dengan Pengaman Open SSL pada Jaringan VPN Mikrotik di SMKS Dwiwarna," *CESS (Journal of Computer Engineering System and Science)*, vol. 3, 2018.
- [18] F. Rofii, F.- Hunaini, dan S. Sholawati, "Kinerja Jaringan Komunikasi Nirkabel Berbasis Xbee pada Topologi Bus, Star dan Mesh," *ELKOMIKA*, vol. 6, no. 3, hlm. 393, Okt 2018, doi: 10.26760/elkomika.v6i3.393.
- [19] E. F. Cahyadi, R. D. Cahyani, dan A. Hikmaturokhman, "Analisa Karakteristik Teori Antrian pada Jaringan IP Multimedia Subsystem (IMS) Menggunakan OPNET Modeler 14.5," hlm. 9, 2015.

