



# Analisis Kualitas Layanan *Video Live Streaming* pada Jaringan Lokal Universitas Telkom

## *Quality of Service Analysis for Live Streaming Video Services on Telkom University Local Network*

Anggelina I Diwi<sup>1</sup>, R Rumani M<sup>2</sup>, Ida Wahidah<sup>3</sup>

<sup>1,2,3</sup>Fakultas Teknik Elektro, Universitas Telkom

<sup>1,2,3</sup>Jl. Telekomunikasi No. 1, Bandung 40257

<sup>1</sup>anggelina\_flash@yahoo.co.id , <sup>2</sup>rumani@telkomuniversity.ac.id, <sup>3</sup>idw@ittelkom.ac.id

Naskah diterima: 17 Juli 2014 ; Direvisi: 4 September 2014; Disetujui: 15 September 2014

**Abstract**— Streaming is one kind of technology that allows file to be operated immediately without waiting for download finished and continuous without any interruption. To apply video streaming into the network, it needs first to calculate the available bandwidth that supports data transmission. Bandwidth is the important parameter for streaming in the network. The bigger available bandwidth, the better video quality presented. The objective of this research is to know the bandwidth necessity for video live streaming service; the method used in this research is measuring the network performance directly from the field, Telkom University local network. Implementation of streaming server-client media in this research make use a different in file (video) based on the amount of frame (fps). This streaming video scenario is measured in different background traffic , to see its influence in network QoS parameter. Testing of Quality of Service ( QoS ) performance in video streaming implementation uses software network analyzer Wireshark. The result shows that video with frame rate bigger than 15 fps, gives bigger jitter and throughput.

**Keywords**— Bandwidth, Video Service, Live Streaming, LAN

**Abstrak**— *Streaming* adalah salah satu bentuk teknologi yang memperkenalkan file digunakan secara langsung tanpa menunggu selesainya unggahan (*download*) dan berlangsung secara kontinyu tanpa interupsi. Untuk mengaplikasikan *video streaming* kedalam jaringan, diperlukan pertama-tama untuk mengkalkulasi *bandwidth* yang tersedia, untuk mendukung transmisi data. *Bandwidth* merupakan parameter penting untuk *streaming* di dalam jaringan. Makin besar *bandwidth* yang tersedia, makin baik kualitas video yang ditampilkan. Penelitian ini bertujuan untuk mengetahui kebutuhan *bandwidth* untuk layanan *video live streaming*; metode yang digunakan di dalam penelitian ini adalah dengan mengadakan pengukuran unjuk kerja jaringan secara langsung di lapangan, yaitu LAN di lingkungan Universitas Telkom. Implementasi media *streaming server-client* di dalam penelitian ini menggunakan *file* video yang berbeda, berdasarkan jumlah frame yang dikirim (fps). Skenario *video streaming* ini dilakukan dengan menggunakan latar belakang trafik yang bervariasi, untuk melihat pengaruhnya

terhadap parameter QoS jaringan. Pengujian performansi *Quality of Service (QoS)* dalam implementasi *video live streaming* ini menggunakan *software network analyzer* Wireshark. Hasil penelitian menunjukkan, bahwa video dengan laju *frame* lebih besar dari 15 fps, memberikan *jitter* dan *throughput* yang besar pula.

**Kata kunci**— Lebar pita, Layanan Video, *Live Streaming*, LAN.

### I. PENDAHULUAN

Saat ini, trend perkembangan aplikasi layanan multimedia telah menggabungkan komunikasi *video* dan *voice*. Aplikasi tradisional pada jaringan TCP/IP, umumnya hanya melibatkan komunikasi antara dua *host*. Namun dalam perkembangannya, pengguna jaringan dapat melakukan komunikasi yang melibatkan lebih dari dua pihak secara bersamaan. *VideoLAN Streaming* merupakan salah satu aplikasi jaringan komputer yang bersifat multimedia, *real-time*, interaktif dan merupakan salah satu solusi kebutuhan *multipoint*.

Dalam melakukan perencanaan sebelum menggelar fasilitas *video* melalui suatu jaringan, kita perlu memperhitungkan kebutuhan *bandwidth* untuk sukses-nya sebuah konferensi *video*.

Secara umum ada dua (2) kebutuhan *bandwidth* yang perlu dipenuhi, yaitu:

- Kebutuhan *bandwidth* untuk mengirimkan sinyal gambar / *video*.
- Kebutuhan *bandwidth* untuk mengirimkan sinyal suara / *audio*.

Diantara kedua kebutuhan di atas, kebutuhan *bandwidth* pengiriman *video* menjadi sangat penting karena akan memakan sebagian besar *bandwidth* komunikasi yang ada. Tidak mengherankan jika teknik-teknik untuk melakukan kompresi data menjadi sangat strategis yang memungkinkan penghematan *bandwidth* telekomunikasi. Contoh penggunaan

teknologi yang turut mengkonsumsi *bandwidth* dalam jumlah besar adalah *video streaming* dan *voice over IP (VoIP)*.

Pada aplikasi *live streaming* masalah yang ada bertambah dengan adanya proses *capturing* dan *live decoding* pada sisi *server*. Selain masalah pada *server*, masalah terbesar yang dihadapi dari teknologi ini adalah keterbatasan *bandwidth* sedangkan proses komunikasi menggunakan *digital video* ini menghabiskan *resource* yang cukup besar. Semakin besar *bandwidth* yang tersedia, semakin bagus kualitas *video* yang ditampilkan. Aplikasi *streaming* media yang membutuhkan *bitrate* cukup tinggi menyebabkan beban jaringan bertambah sehingga menyebabkan *service* yang diberikan tidak dapat berjalan dengan baik.

Urgensi dari penelitian ini adalah untuk mengetahui seberapa besar pengaruh laju *frame* di dalam jaringan, dikaitkan dengan *bandwidth* yang tersedia, terhadap parameter QoS jaringan.

## II. TINJAUAN PUSTAKA

### A. Teknologi Video Streaming

#### 1) Pengertian

Apostolopoulos et.al (Apostolopoulos et.al, 2002), mengatakan, dalam dunia internet, *streaming* merupakan sebuah teknologi yang mampu mengkompresi atau menyusutkan ukuran *file audio* dan *video* agar mudah ditransfer melalui jaringan internet. Penransferan *file audio* dan *video* tersebut dilakukan secara terus menerus. Dari sudut pandang prosesnya, *streaming* berarti sebuah teknologi pengiriman *file* dari *server* ke *client* melalui jaringan *packet based*. *Streaming* merupakan sebuah metode untuk membuat *audio*, *video*, dan multimedia yang lain yang tersedia untuk *real-time* pada tipe jaringan yang berbeda. Data pada *file streaming* di bagi-bagi ke dalam beberapa paket kecil yang dikirim ke sebuah aliran secara terus menerus ke perangkat *end-user* atau *mobile phone*.

Aplikasi dalam layanan *streaming* di bagi menjadi dua, yaitu “*on-demand*” dan “*live*”. Layanan *streaming on-demand* contohnya adalah musik dan *video*. Sedangkan layanan *streaming* yang *live* contohnya adalah acara radio atau acara televisi yang disiarkan secara *broadcast* pada saat itu juga.

Menurut (Jaromil, 2002), ide dasar dari *video streaming* adalah untuk membagi-bagi *video* asli menjadi beberapa *paket* yang kemudian dikirim secara berurutan, dan memungkinkan *receiver* melakukan *decode* dan *playback* *video* berdasarkan *packet* tersebut tanpa harus menunggu seluruh *video* terkirim.

Menurut (Austerbery, 2013), ada beberapa masalah dasar dalam *video streaming*, yaitu:

#### a) Bandwidth

Jika pengirim mengirimkan data lebih cepat dibandingkan dengan ketersediaan *bandwidth* yang ada, maka akan terjadi *congestion* (kemacetan) pada jaringan, *packet loss*, dan kualitas *video* yang jelek. Tapi apabila pengirim mengirimkan paket data lebih lambat dari *bandwidth* yang tersedia, maka kualitas *video* yang sampai ke penerima akan kurang optimal.

#### b) Jitter

*Jitter* merupakan masalah yang menyebabkan penerima harus menerima/*decode*/menampilkan *frame* dengan *rate* yang konstan, dan setiap *frame* yang terlambat datang akan menyulitkan rekonstruksi *video* yang diterima.

#### c) Loss rate

Pada jaringan *wireless*, *loss rate* dapat disebabkan oleh *bit error* dan *burst error*. *Loss rate* ini dapat menimbulkan kerusakan pada kualitas *video* hasil rekonstruksi. Untuk mengatasi *loss rate* ini, sistem *video streaming* dapat didesain dengan fasilitas *error rate control*.

#### 2) Real Time Encoding dan Pre-encoded ( Stored Video )

*Video* dapat dicapture dan diencode untuk kebutuhan komunikasi *real time* atau dapat juga dipre-encoded dan disimpan dalam format *DVD* atau *Video-CD* untuk dijalankan pada saat dibutuhkan. Contoh aplikasi yang membutuhkan *real time encoding* adalah *videophone*, *video conferencing* dan *interactive games*. Sedangkan aplikasi *video pre-encoded* antara lain *DVD*, *Video-CD*, dan yang dikenal dengan penyimpanan secara lokal atau *Video On Demand (VOD)*. Penyimpanannya dilakukan secara *remote* di *server*.

#### 3) Transfer Video via File Download dan Transfer Video via Streaming

Sebuah *file video* dapat ditampilkan di *user* dengan menggunakan dua metode transfer *file*. Pertama, dengan men-download keseluruhan *file video* dan yang kedua, dengan melakukan proses *streaming*. Kedua metode ini memiliki kekurangan dan kelebihan masing-masing.

*File video* yang diambil dari *server* dengan cara *download*, tidak dapat ditampilkan sebelum seluruh *file video* tersebut selesai tersalin ke *buffer*. Untuk itu, diperlukan media penyimpanan yang cukup besar dan waktu yang diperlukan untuk proses *download* cukup lama karena *file video* biasanya berukuran besar. Ide dasar dari *video streaming* ini adalah membagi beberapa paket *video* menjadi beberapa bagian, mentransmisikan paket data tersebut, kemudian penerima (*receiver*) dapat mendecode dan memainkan potongan *video* tanpa harus menunggu seluruh *file* terkirim ke mesin penerima.

#### 4) Streaming Protocol

Teknik *streaming* tingkat aplikasi untuk mendapatkan hasil terbaik dari layanan *best effort* :

- a) *Buffer* disisi *client*
  - b) Menggunakan *UDP* bukan *TCP*
  - c) Menggunakan teknik pengkodean multimedia
- Aplikasi layanan *streaming* saat ini banyak menggunakan *RTSP (Real Time Streaming Protocol)*. Adapun cara kerjanya adalah sebagai berikut :
- a) *Metafile* berkomunikasi melalui *web browser*
  - b) *Browser* menjalankan *player*
  - c) *Player* membuat kondisi kendali *RTSP* dan koneksi data ke *server streaming*

*Real Time Protokol ( RTP )* merupakan struktur paket standar untuk membawa data *audio* dan *video*. *RTP* berfungsi untuk transmisi *real time* data seperti *audio*, *video*, multimedia secara *end-to-end*. *RTP* mendukung transmisi *unicast*, *broadcast* dan *multicast*. *RTP* hanyalah sebagai protokol *transport* dan hal ini tidak menjamin *quality of service* untuk layanannya.

*Streaming server* memotong media data untuk masing-masing *track* menjadi paket *RTP* yang kemudian ditransmisikan melalui *UDP*. Masing-masing paket *RTP* ditransmisikan dengan beberapa informasi yaitu :

- a) *RTP header*
- b) *RTP payload header*

c) RTP payload data

Pada RTP header memuat informasi tentang versi RTP, sequence number, timestamp, synchronization Source Identifier (SSRC). Sequence number digunakan untuk inialisasi paket yang dikirimkan dan bisa dipakai untuk penghitungan packet loss. Timestamp menunjukkan waktu paket yang dipakai untuk sinkronisasi dan penghitungan jitter. SSRC mengindikasikan keunikan dan keaslian paket dalam suatu session. Kunci keunggulan RTP adalah kemampuan interoperabilitas : jika dua aplikasi Internet Phone menggunakan RTP, maka tidak akan terjadi masalah.

B. Frame Rate Video

Video merupakan gabungan gambar-gambar mati yang dibaca berurutan dalam suatu waktu dengan kecepatan tertentu. Gambar-gambar yang digabung tersebut dinamakan frame dan kecepatan pembacaan gambar disebut dengan frame rate, dengan satuan fps (frame per second). Karena dimainkan dalam kecepatan yang tinggi maka akan tercipta ilusi gerak yang halus. Inilah prinsip dasar film, video dan animasi. Semakin besar nilai frame rate maka akan semakin halus pergerakan yang ditampilkan. Untuk menghasilkan gambar pergerakan yang halus, diperlukan frame rate minimal 10 fps. Film-film yang kita lihat di bioskop adalah film yang diproyeksikan dengan frame rate sebesar 24 fps, sedangkan video yang kita lihat di televise kira-kira memiliki frame rate sebesar 30 fps (tepatnya 29.97 fps).

Menurut (Sarosa., et.al, 2000), umumnya terdapat 3 format standar yang digunakan di seluruh dunia yaitu :

TABEL 1. FORMAT VIDEO STANDAR

Format Standar	Negara	Frame rate
NTSC	USA, Jepang, Kanada, Meksiko dan Korea	29,97
PAL	Indonesia, Inggris, Australia, Eropa dan Cina	25
SECAM	Perancis, Timur Tengah dan Afrika	25

Sumber : (Sarosa et.al, 2000)

C. Video Coding

Kompresi adalah proses merapikan data menjadi jumlah bit yang lebih sedikit. Kompresi video (video coding) adalah proses merapikan atau mempersingkat urutan dari serangkaian video digital jumlah bit yang lebih sedikit. Kompresi video digunakan dalam dunia digital karena kebutuhan penyimpanan dan bandwidth yang besar oleh data video. Tujuan utama dari kompresi video adalah menurunkan kebutuhan bitrate dengan cara memanfaatkan redundansi dari sinyal video.

D. Quality Of Service ( QoS )

Komunikasi saat ini dapat dilakukan melalui jaringan data /internet. Namun bukan berarti komunikasi data tidak memiliki hambatan. Masalah rendahnya kualitas komunikasi berhubungan erat dengan permasalahan jaringan IP yaitu rendahnya bandwidth dari jaringan itu sendiri. Ini berarti kita harus mengetahui masalah kualitas komunikasi dan kemungkinan solusinya supaya mendapat QoS yang lebih bagus.

(Croll, et. al.,1999), menyatakan bahwa QoS adalah kemampuan suatu jaringan untuk memberikan layanan yang lebih baik pada trafik data tertentu pada berbagai jenis

platform teknologi. Masalah utama dari Qos adalah delay, jitter, packet loss, dan throughput.

1) Delay

Delay adalah total waktu yang dibutuhkan paket untuk menempuh jarak dari asal ( pengirim ) ke tujuan ( penerima ). Delay dari pengirim ke penerima pada dasarnya tersusun atas hardware latency, delay akses, dan delay transmisi. Delay yang paling sering dialami oleh trafik yang lewat adalah delay transmisi

2) Jitter

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

3) Packet Loss

Packet loss didefinisikan sebagai kegagalan transmisi paket mencapai tujuannya atau jumlah paket yang hilang. Kegagalan paket tersebut mencapai tujuan, dapat disebabkan oleh beberapa kemungkinan, diantaranya yaitu:

- a) Terjadinya overload trafik didalam jaringan,
- b) Tabrakan (congestion) dalam jaringan,
- c) Error yang terjadi pada media fisik,
- d) Kegagalan yang terjadi pada sisi penerima antara lain bisa disebabkan karena overflow yang terjadi pada buffer.

Di dalam implementasi jaringan, nilai packet loss ini diharapkan mempunyai nilai yang minimum.

4) Throughput

Throughput adalah kecepatan (rate) transfer data efektif, yang diukur dalam bps. Header dalam paket data mengurangi nilai ini. Throughput berhubungan dengan bandwidth yang tersedia di jaringan. Bandwidth yang disediakan tidak semua dipakai oleh aplikasi di jaringan. Besarnya bandwidth yang dipakai aplikasi tersebut adalah throughput. Throughput dihitung dengan mengetahui terlebih dahulu jumlah paket data yang diterima, lalu dibagi dengan lama pengamatan dalam satuan waktu.

E. Kebutuhan Bandwidth Video

Ada dua (2) buah standar kompresi video yang umum digunakan dalam pengiriman video melalui saluran komunikasi yang sempit, yaitu:

- a) H.261 – biasanya menggunakan kanal ISDN dengan kecepatan p x 64Kbps, dimana p adalah 1, 2, 3, ..., 30.
- b) H.263 – di arahkan untuk mengirimkan gambar video berkecepatan rendah mulai dari 20-30Kbps ke atas.

Beberapa hal yang perlu di perhatian adalah:

- a) Jika kita menggunakan video hitam-putih akan memakan bandwidth lebih kecil daripada jika kita melakukan konferensi menggunakan video berwarna.
- b) Jika kita menggunakan kecepatan pengiriman frame per second (fps) video yang rendah, akan memakan bandwidth yang rendah dibandingkan frame per second (fps) yang tinggi.

Video yang cukup baik biasanya dikirim dengan kecepatan frame per second (fps) sekitar 30 fps. Jika dikirimkan tanpa kompresi, sebuah video dengan 30fps akan mengambil bandwidth kira-kira 9Mbps, amat sangat besar untuk ukuran kanal komunikasi data.

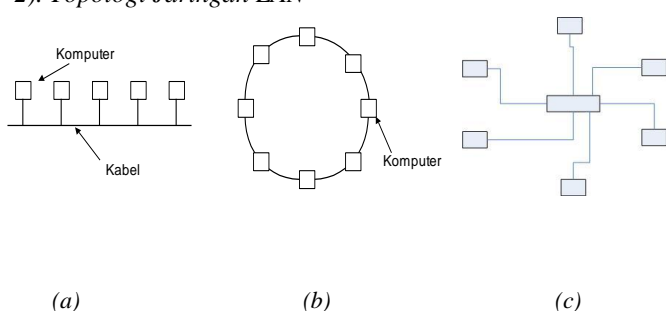
## F. Local Area Network (LAN)

### 1). Pengertian LAN

Menurut Regan (Regan, 2003), *Local Area Network* (LAN) merupakan jaringan milik pribadi di dalam sebuah gedung atau kampus yang berukuran sampai beberapa kilometer. LAN seringkali digunakan untuk menghubungkan komputer-komputer pribadi dan *workstation* dalam kantor perusahaan atau pabrik-pabrik untuk memakai bersama *resource* (misalnya, *printer*, *scanner*) dan saling bertukar informasi. LAN dapat dibedakan dari jenis jaringan lainnya berdasarkan tiga karakteristik, yaitu ukuran, teknologi transmisi dan topologinya.

LAN mempunyai ukuran yang terbatas, yang berarti bahwa waktu transmisi pada keadaan terburuknya terbatas dan dapat diketahui sebelumnya. Dengan mengetahui keterbatasannya, menyebabkan adanya kemungkinan untuk menggunakan jenis desain tertentu. Hal ini juga memudahkan manajemen jaringan.

### 2). Topologi Jaringan LAN



Gambar 1. Jenis Jaringan LAN. (a) Bus (b) Ring (c) Star

Pada jaringan *Bus* (yaitu kabel linier), pada suatu saat sebuah mesin bertindak sebagai *master* dan diijinkan untuk mengirim paket. Mesin-mesin lainnya perlu menahan diri untuk tidak mengirimkan apapun. Maka untuk mencegah terjadinya konflik, ketika dua mesin atau lebih ingin mengirimkan secara bersamaan, maka mekanisme pengatur diperlukan. Mekanisme pengatur dapat berbentuk tersentralisasi atau terdistribusi. IEEE 802.3 yang populer disebut Ethernet merupakan jaringan *multicast* bus dengan pengendali terdesentralisasi yang beroperasi pada kecepatan 10 s/d 100 Mbps.

Sistem *multicast* yang lain adalah *Ring*, pada topologi ini setiap bit dikirim ke daerah sekitarnya tanpa menunggu paket lengkap diterima. Biasanya setiap bit mengelilingi *ring* dalam waktu yang dibutuhkan untuk mengirimkan beberapa bit, bahkan seringkali sebelum paket lengkap dikirim seluruhnya. Seperti sistem *multicast* lainnya, beberapa aturan harus dipenuhi untuk mengendalikan akses simultan ke *ring*. Demikian juga dengan sistem *Star*, semua *host* yang terdapat dalam jaringan akan mendapatkan paket apabila *server* mengirimkan paket secara *multicast*.

### 3). Komponen LAN

Beberapa komponen dasar yang biasanya membentuk suatu LAN adalah sebagai berikut (Forouzan, 2002):

#### a) Workstation

*Workstation* merupakan *node* atau *host* yang berupa suatu sistem komputer. Sistem komputer ini dapat berupa PC atau

dapat pula berupa suatu komputer yang besar seperti sistem *minicomputer*, bahkan suatu *mainframe*.

#### b) Server

Merupakan perangkat keras (*hardware*) yang berfungsi untuk melayani jaringan dan *workstation* yang terhubung pada jaringan tersebut.

#### c) Link (hubungan)

Hubungan *workstation* dan *server* dalam LAN dikenal sebagai media transmisi yang umumnya berupa kabel. Adapun beberapa contoh dari *link* adalah kabel *Twisted Pair*, kabel *Coaxial* dan kabel *Fiber Optik*.

#### d) Network Interface Card (NIC)

Suatu *workstation* tidak dihubungkan secara langsung dengan kabel jaringan ataupun *tranceiver cable*, tetapi melalui suatu rangkaian elektronika yang dirancang khusus untuk menangani network protocol yang dikenal dengan *Network Interface Card* (NIC).

#### e) Network Software

Tanpa adanya *software* jaringan maka jaringan tersebut tidak akan bekerja sebagaimana yang dikehendaki. *Software* ini juga yang memungkinkan sistem komputer yang satu berkomunikasi dengan sistem komputer yang lain.

Selain komponen-komponen utama diatas, LAN juga memiliki peralatan pendukung lainnya yaitu :

- Repeater*, merupakan perangkat yang berfungsi meneruskan dan memperkuat sinyal dan bekerja pada lapisan fisik OSI *layer*.
- Hub*, merupakan perangkat yang berfungsi sebagai pembagi saluran. *Hub* tertentu juga menguatkan sinyal yang dikenal dengan *hub* aktif. Sedangkan *hub* yang tidak menguatkan sinyal dikenal sebagai *hub* pasif.
- Switch*, merupakan perangkat yang berfungsi juga untuk merutekan trafik dengan menggunakan *MAC address*.
- Router*, merupakan komponen dasar yang berfungsi merutekan paket IP dari pengirim menuju ke penerima. Perangkat ini bekerja pada *layer 3* untuk keperluan ruting.

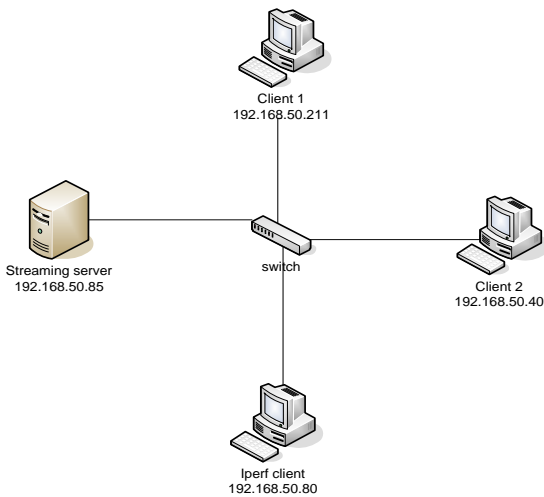
## G. Transmisi Multicast

Supandi (Supandi, 2006), mengatakan, bahwa transmisi *multicast* merupakan teknologi *bandwidth-conserving* yang dapat mengurangi trafik jaringan dengan mengirimkan sebuah *stream* tunggal secara simultan ke beberapa *client* atau *host*. Untuk aplikasi yang membutuhkan *bandwidth* besar seperti *streaming*, metode *multicast* dapat menjadi suatu solusi untuk mengatasi keterbatasan *bandwidth* dan ketersediaan jaringan terhadap paket *stream* yang cenderung besar dan kontinyu. Kunci utama keunggulan komunikasi *multicast* dibandingkan dengan komunikasi lainnya adalah kemampuan komunikasi *multicast* untuk menghindari terjadinya pengiriman banyak data yang sama pada *link* yang sama.

## III. METODE PENELITIAN

### A. Arsitektur Sistem

Arsitektur system yang digunakan dalam penelitian ini adalah seperti gambar dibawah ini :



Gambar 2. Arsitektur Sistem

**B. Skenario Implementasi**

**1) Skenario 1 (pengaruh frame rate)**

Dilakukan pengukuran untuk *video on demand* dengan masukan video yang memiliki *frame rate* yang berbeda yaitu 15 fps dan 30 fps. Pengukuran dilakukan pada *background traffic* yang berbeda yaitu 50 Mbps, 75 Mbps, dan 100 Mbps.

**2) Skenario 2 (pengaruh bitrate)**

Dilakukan pengukuran untuk *video live streaming* yang dicapture dari *webcam*. *Codec* yang digunakan adalah *mp4g* dengan variasi *bitrate* 16 kbps, 64 kbps, 128 kbps, dan 256 kbps. Proses *live streaming* ini diukur pada *background traffic* 50 Mbps dan 75 Mbps.

**3) Skenario 3 (pengaruh background traffic)**

Dilakukan pengukuran untuk *video live streaming* yang dicapture dari *webcam*. *Codec* yang digunakan adalah *mp4v* dengan *bitrate* 16 kbps dan 256 kbps. Proses *live streaming* ini diukur pada *background traffic* 50 Mbps, 75 Mbps, dan 100 Mbps.

**IV. HASIL PENELITIAN DAN PEMBAHASAN**

**A. Analisa Performansi Streaming Server**

**1) Pengukuran Delta**

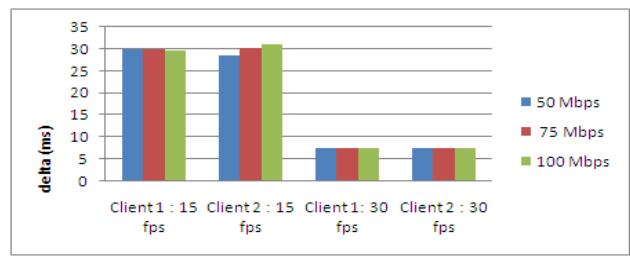
**a) Tujuan Pengukuran Delta**

*Delta* adalah *delay* yang terjadi antar kedatangan paket. Pengukuran ini bertujuan untuk melakukan evaluasi *delay network* pada hubungan antara *client* dan *server* selama *streaming video* berlangsung.

**b) Hasil Pengukuran Delta**

TABEL 2. NILAI DELTA SKENARIO 1

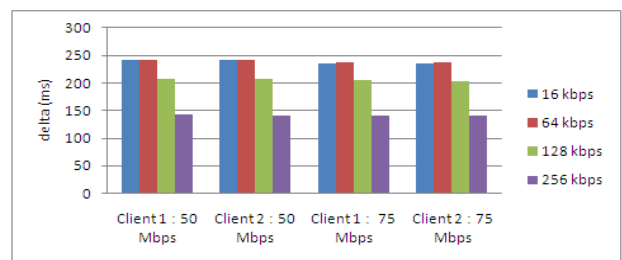
Delta (ms)	Background traffic (Mbps)		
	50	75	100
Client 1 : 15 fps	29.9813	29.9854	29.4990
Client 2 : 15 fps	28.4806	30.2598	30.9264
Client 1 : 30 fps	7.2741	7.2719	7.2726
Client 2 : 30 fps	7.2729	7.2739	7.2726



Gambar 3. Grafik Nilai Delta Scenario 1

TABEL 3. NILAI DELTA SKENARIO 2

	Bitrate			
	16 kbps	64 kbps	128 kbps	256 kbps
Client 1 : 50 Mbps	242.6319	243.7934	209.0715	143.1458
Client 2 : 50 Mbps	243.7159	243.7732	209.4590	141.6415
Client 1 : 75 Mbps	236.8888	239.2334	206.6371	141.2084
Client 2 : 75 Mbps	236.8246	239.8142	205.2264	141.955



Gambar 4. Grafik Nilai Delta Skenario 2

TABEL 4. NILAI DELTA SKENARIO 3

Delta (ms)	Background traffic		
	50 Mbps	75 Mbps	100 Mbps
Client 1 : 16 kbps	242.6319	236.8888	243.7592
Client 2 : 16 kbps	243.7159	236.8246	243.8768
Client 1 : 256 kbps	143.1458	141.2084	142.5284
Client 2 : 256 kbps	141.6415	141.955	142.0482



Gambar 5. Grafik Nilai Delta Skenario 3

**c) Analisis Hasil Pengukuran**

Pada percobaan 1, diperoleh nilai delta yang semakin menurun untuk *frame rate* yang makin besar. Nilai *delta* terbesar 30.2127 ms dihasilkan pada *streaming video* 15 fps pada *background traffic* 100 Mbps. Sedangkan nilai *delta* terkecil 7.2729 ms diperoleh pada *streaming video* 30 fps dengan *background traffic* 75 Mbps. Penurunan nilai *delta* itu disebabkan karena makin besar *frame rate* maka *bitrate* semakin besar, jumlah *bit* yang dikirimkan tiap detik makin bertambah, sehingga *delay* antar paket yang terjadi semakin kecil. Demikian juga pada percobaan 2, diperoleh bahwa

dengan bertambahnya *bitrate* maka nilai *delta* yang dihasilkan semakin kecil. Nilai *delta* terbesar diperoleh pada *streaming* dengan *bitrate* 16 kbps (243.818) ms pada *background traffic* 100 Mbps. Sedangkan nilai *delta* terkecil diperoleh pada *streaming* dengan *bitrate* 256 kbps 141.5817 ms pada *background traffic* 75 Mbps. Pada percobaan 3, dengan mengubah besar *background traffic* diperoleh bahwa nilai *delta* terbesar diperoleh pada *streaming* dengan *bitrate* yang lebih kecil 16 kbps (243.818 ms) pada *background traffic* 100 mbps. Sedangkan nilai *delta* terkecil diperoleh pada *streaming* dengan *bitrate* yang lebih besar 256 kbps (141.5817 ms) pada *background traffic* 75 Mbps. Jadi makin besar *background traffic*, nilai *delta* makin besar. Demikian sebaliknya, makin kecil *background traffic*, nilai *delta* makin kecil.

2) Pengukuran Jitter

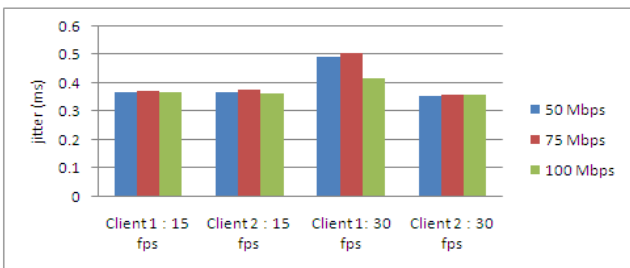
a) Tujuan Pengukuran Jitter

*Jitter* adalah variasi waktu kedatangan paket. *Jitter* diukur antara paket sekarang dengan paket yang datang sebelumnya.

b) Hasil Pengukuran Jitter

TABEL 5. NILAI JITTER SKENARIO 1

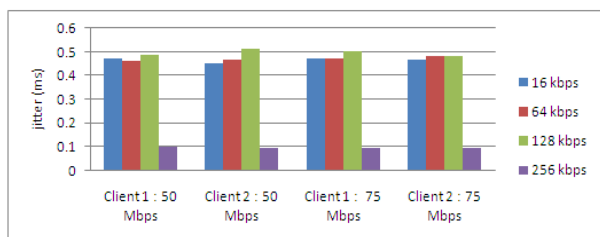
	Background traffic		
	50 Mbps	75 Mbps	100 Mbps
Client 1 : 15 fps	0.3684	0.3724	0.3671
Client 2 : 15 fps	0.36838	0.3746	0.3649
Client 1 : 30 fps	0.49328	0.5053	0.41695
Client 2 : 30 fps	0.3540	0.3598	0.3584



Gambar 6. Grafik Nilai Jitter Scenario 1

TABEL 6. NILAI JITTER SKENARIO 2

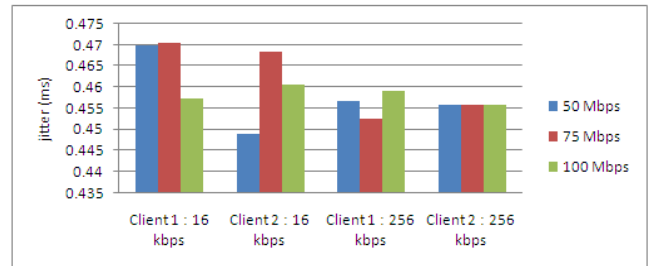
	Bitrate			
	16 kbps	64 kbps	128 kbps	256 kbps
Client 1 : 50 Mbps	0.46977	0.4638	0.4878	0.0952
Client 2 : 50 Mbps	0.44872	0.4685	0.5116	0.0938
Client 1 : 75 Mbps	0.4706	0.47268	0.50489	0.0941
Client 2 : 75 Mbps	0.4684	0.48155	0.48276	0.0918



Gambar 7. Grafik Nilai Jitter Scenario 2

TABEL 7. NILAI JITTER SKENARIO 3

	Background traffic		
	50 Mbps	75 Mbps	100 Mbps
Client 1 : 16 kbps	0.4698	0.4706	0.4572
Client 2 : 16 kbps	0.4487	0.4684	0.4606
Client 1 : 256 kbps	0.4568	0.45255	0.45899
Client 2 : 256 kbps	0.4559	0.45564	0.45573



Gambar 8. Grafik Nilai Jitter Skenario 3

c) Analisis Hasil Pengukuran

Pada percobaan 1, diperoleh nilai *jitter* terbesar pada *frame rate* 30 fps (75 Mbps) yaitu 0.4325 ms. Sedangkan nilai terkecilnya diperoleh pada *frame rate* 15 fps (0.366 ms) dengan *background traffic* 100 Mbps. Pada percobaan 2, nilai *jitter* terkecil diperoleh pada *bitrate* 256 kbps yaitu sebesar 0.0929 ms. Sedangkan nilai terbesarnya diperoleh pada penggunaan *bitrate* 128 kbps yaitu 0.4997 ms. Pada percobaan 3, diperoleh bahwa *background traffic* sedikit mempengaruhi nilai *jitter*. Hal ini dapat dilihat pada tabel hasil pengukuran dimana pada penggunaan *bitrate* yang kecil dan *bitrate* yang besar, tidak menunjukkan perbedaan hasil yang signifikan.

Dari semua skenario yang telah dijalankan, nilai *jitter* yang dihasilkan kurang lebih sama dan nilai *jitter* tersebut masih dalam batas standar yang ditentukan ITU ( nilai *jitter* untuk *streaming* tidak boleh melebihi 30 ms). *Jitter* yang makin besar akan mengakibatkan keterlambatan paket data berikutnya sehingga paket itu akan dibuang dan akan menyebabkan adanya *packet loss*. Dalam layanan *streaming multimedia*, dibutuhkan nilai *jitter* yang kecil dan cenderung stabil. Hal ini akan memberikan laju kedatangan paket yang bersifat kontinyu sehingga paket-paket yang datang ke dalam *buffer* tidak berlebih/berkurang.

3) Pengukuran Packet loss

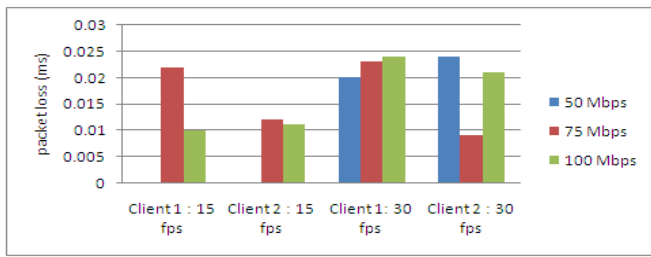
a) Tujuan Pengukuran Packet Loss

*Packet Loss* adalah jumlah paket hilang yang terjadi saat pengiriman paket. Pengukuran ini bertujuan untuk melakukan evaluasi *jitter* pada hubungan antara *client* dan *server* selama *streaming video* berlangsung, sehingga dapat disimpulkan seberapa baik kualitas *video streaming* yang didapatkan.

b) Hasil Pengukuran Packet Loss

TABEL 8. NILAI PACKET LOSS SKENARIO 1

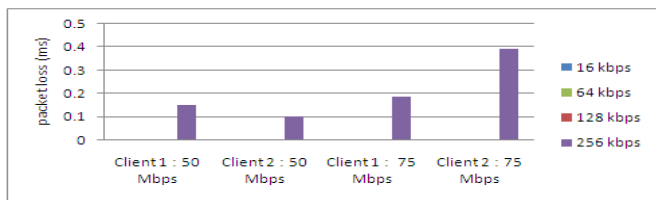
	Background traffic (Mbps)		
	50 Mbps	75 Mbps	100 Mbps
Client 1 : 15 fps	0	0.022	0.01
Client 2 : 15 fps	0	0.012	0.011
Client 1 : 30 fps	0.02	0.023	0.024
Client 2 : 30 fps	0.024	0.009	0.021



Gambar 9. Grafik Nilai Packet Loss Skenario 1

TABEL 9. NILAI PACKET LOSS SKENARIO 2

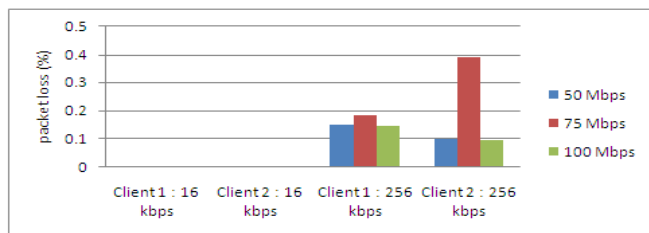
	Bitrate			
	16 kbps	64 kbps	128 kbps	256 kbps
Client 1 : 50 Mbps	0	0	0	0.149
Client 2 : 50 Mbps	0	0	0	0.1
Client 1 : 75 Mbps	0	0	0	0.185
Client 2 : 75 Mbps	0	0	0	0.392



Gambar 10. Grafik Nilai Packet Loss Skenario 3

TABEL 10. NILAI PACKET LOSS SKENARIO 3

	Background traffic		
	50 Mbps	75 Mbps	100 Mbps
Client 1 : 16 kbps	0	0	0
Client 2 : 16 kbps	0	0	0
Client 1 : 256 kbps	0.149	0.185	0.144
Client 2 : 256 kbps	0.1	0.392	0.096



Gambar 11. Grafik Nilai Packet Loss Skenario 3

c) Analisis Hasil Pengukuran

Pada percobaan 1, diperoleh bahwa semakin besar frame rate maka packet loss yang dihasilkan pun akan semakin besar. Pada streaming video dengan frame rate 30 fps didapatkan packet loss sebesar 0.025 % pada background traffic 100 Mbps. Sedangkan streaming video untuk frame rate 15 fps diperoleh packet loss yang kecil. Dari hal ini terlihat pula bahwa perubahan terhadap penambahan background traffic dapat mempengaruhi nilai packet loss, dimana semakin besar background traffic maka packet loss pun semakin besar. Pengaruh background traffic pun terlihat pada percobaan 2. Semakin besar bitrate yang digunakan maka packet loss yang dihasilkan pun semakin besar. Packet loss terbesar dihasilkan pada penggunaan bitrate 256 kbps

(0.2885 %). Sedangkan untuk bitrate yang kecil, dihasilkan packet loss 0 %

Secara keseluruhan terlihat bahwa perubahan frame rate, bitrate dan background traffic yang semakin besar akan menghasilkan packet loss yang besar juga. Namun dari hasil pengukuran yang didapatkan, nilai packet loss tersebut masih dalam batas standar dari ITU-T, yaitu dibawah 10 %.

4). Pengukuran Throughput

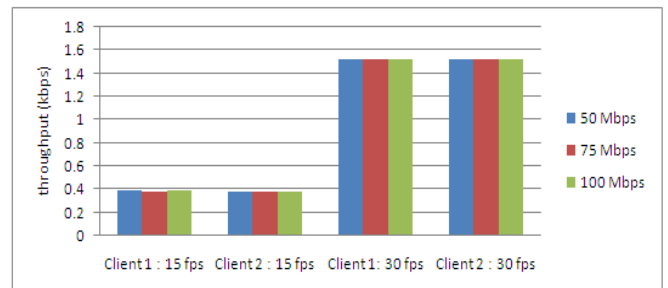
a) Tujuan Pengukuran Throughput

Throughput adalah besar paket yang sampai dengan sukses dari server menuju client dalam selang periode waktu tertentu. Pengukuran ini bertujuan untuk mengetahui besar throughput pada hubungan antara server dan client sesuai dengan efek dari masing-masing skenario yaitu pengaruh bitrate, pengaruh background traffic, dan pengaruh jumlah client.

b) Hasil Pengukuran Throughput

TABEL 11. NILAI THROUGHPUT SKENARIO 1

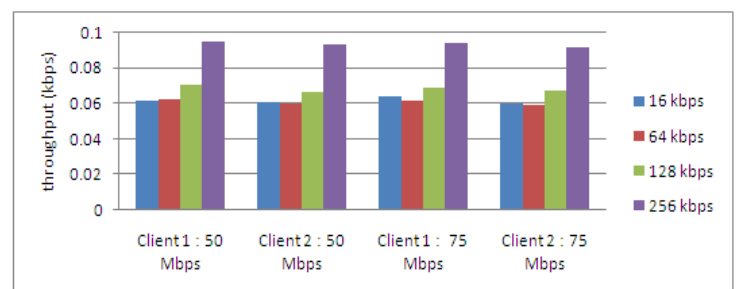
	Background traffic (Mbps)		
	50 Mbps	75 Mbps	100 Mbps
Client 1 : 15 fps	0.3871	0.3838	0.3941
Client 2 : 15 fps	0.378	0.3764	0.3796
Client 1 : 30 fps	1.5288	1.5212	1.5273
Client 2 : 30 fps	1.5248	1.5232	1.521



Gambar 12. Grafik Throughput Skenario 1

TABEL 12. NILAI THROUGHPUT SKENARIO 2

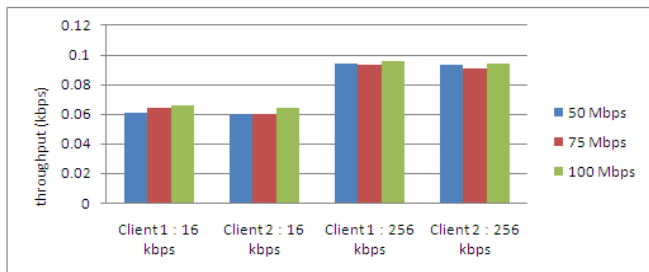
	Bitrate			
	16 kbps	64 kbps	128 kbps	256 kbps
Client 1 : 50 Mbps	0.0614	0.0619	0.0705	0.0952
Client 2 : 50 Mbps	0.0607	0.0601	0.0665	0.0938
Client 1 : 75 Mbps	0.0643	0.0613	0.0685	0.0941
Client 2 : 75 Mbps	0.06	0.059	0.0673	0.0918



Gambar 13. Grafik Throughput Skenario 2

TABEL 13. NILAI THROUGHPUT SKENARIO 3

	Background traffic		
	50 Mbps	75 Mbps	100 Mbps
Client 1 : 16 kbps	0.0614	0.0643	0.0664
Client 2 : 16 kbps	0.0607	0.06	0.0649
Client 1 : 256 kbps	0.0952	0.0941	0.0969
Client 2 : 256 kbps	0.0938	0.0918	0.095



Gambar 14. Grafik Throughput Skenario 3

c) Analisis Hasil Pengukuran

Berdasarkan pengukuran yang telah dilakukan, terdapat pengaruh antara *frame rate*, perubahan besar *bitrate* dan penambahan *background traffic* terhadap nilai *throughput* yang diperoleh. Pada skenario 1 (dimana dilakukan perubahan *frame rate*), menunjukkan bahwa semakin besar *frame rate* maka nilai *throughput* yang dihasilkan pun semakin besar. Pada *streaming* dengan *frame rate* 15 fps diperoleh nilai *throughput* terkecil yaitu 0.3825 Mbps. Sedangkan nilai *throughput* terbesar pada *frame rate* 30 fps yaitu 1.5268 Mbps. Pada percobaan 2, diperoleh bahwa semakin besar *bitrate*, maka *throughput* yang dihasilkan pun semakin besar. Nilai *throughput* terbesar diperoleh pada *bitrate* 256 kbps (0.0929 Mbps). Sedangkan nilai terkecilnya diperoleh pada penggunaan *bitrate* 64 kbps (0.0605 Mbps). Pada percobaan 3 diperoleh nilai *throughput* terbesar pada *background* 100 Mbps dan nilai terkecilnya adalah 0.061 kbps ( pada *background traffic* 50 Mbps).

Semakin besar *background traffic* maka pengiriman data di jaringan dapat terganggu atau dapat menurunkan performansi. Berdasarkan pemahaman ini maka diperoleh perbandingan yang terbalik antara *background traffic* dan nilai *throughput* yang dihasilkan. Namun dari hasil pengukuran pada skenario 3, yang terjadi adalah semakin besar *background traffic* maka nilai *throughput* yang dihasilkan semakin besar meskipun hasil yang diperoleh antara *client 1* dan *client 2* tidak menunjukkan perbedaan yang signifikan. Penurunan nilai *throughput* dapat menyebabkan kualitas gambar pada layanan yang diterima di *client* menjadi kurang baik.

B. Analisa Kebutuhan Bandwidth

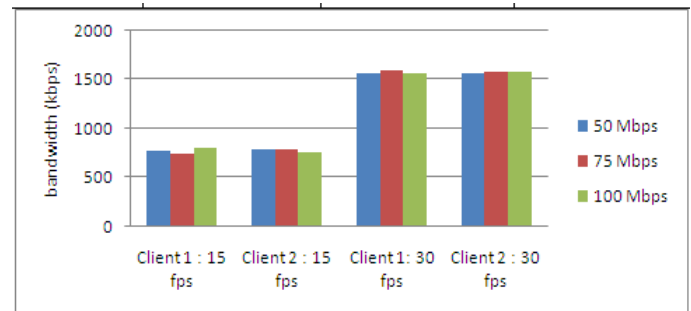
1). Tujuan Pengukuran Bandwidth

*Bandwidth* digambarkan sebagai ukuran kecepatan aliran data yang menyatakan banyaknya informasi yang dapat mengalir dari suatu tempat ke tempat lain dalam suatu waktu tertentu. Tujuan pengukuran *bandwidth* adalah untuk mengetahui *bandwidth* maksimum yang diperlukan aplikasi *live streaming* untuk masing-masing skenario.

2). Hasil Pengukuran Bandwidth

TABEL 14. NILAI BANDWIDTH SKENARIO 1

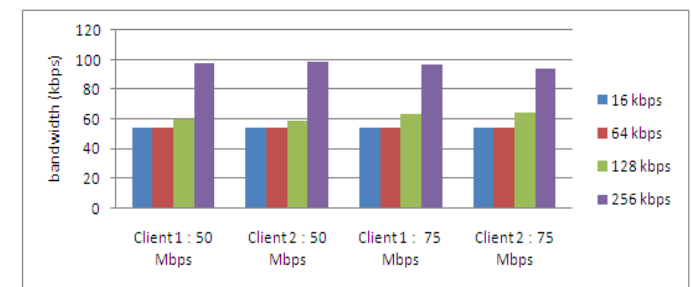
	Background traffic (Mbps)		
	50 Mbps	75 Mbps	100 Mbps
Client 1 : 15 fps	770.112	737.052	791.903
Client 2 : 15 fps	780.757	784.6	753.935
Client 1 : 30 fps	1560.68	1591.402	1556.688
Client 2 : 30 fps	1567.534	1583.807	1578.385



Gambar 15. Grafik Nilai Bandwidth Skenario 1

TABEL 15. NILAI BANDWIDTH SKENARIO 2

	Bitrate			
	16 kbps	64 kbps	128 kbps	256 kbps
Client 1 : 50 Mbps	54.24	54.24	59.665	97.63
Client 2 : 50 Mbps	54.24	54.24	58.58	98.71
Client 1 : 75 Mbps	54.24	54.24	62.92	96.545
Client 2 : 75 Mbps	54.24	54.24	64.005	94.375

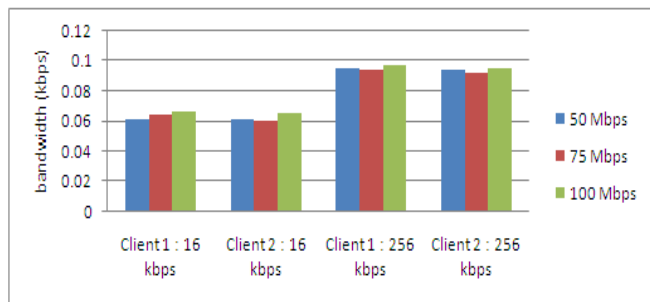


Gambar 16. Grafik Nilai Bandwidth Skenario 2

TABEL 16. NILAI BANDWIDTH SKENARIO 3

	Background traffic (Mbps)		
	50 Mbps	75 Mbps	100 Mbps
Client 1 : 16 kbps	54.24	54.24	54.24
Client 2 : 16 kbps	54.24	54.24	54.24
Client 1 : 256 kbps	97.63	96.545	92.205
Client 2 : 256 kbps	98.71	94.375	91.12





Gambar 17. Grafik Nilai Bandwidth Skenario 3

### 3). Analisis Hasil Pengukuran

Pada percobaan 1, *streaming* dengan *frame rate* 15 fps membutuhkan *bandwidth* maksimal 775.4345 kbps sedangkan *frame rate* 30 fps membutuhkan *bandwidth* maksimal 1587.6045 kbps. Hasil ini menunjukkan bahwa semakin besar *frame rate* maka *bandwidth* maksimum yang dibutuhkan semakin besar juga. Pada percobaan 2, diperoleh nilai *bandwidth* maksimum yang makin besar untuk penambahan *bitrate*. Pada *bitrate codec* 16 kbps diperoleh *bandwidth* maksimum rata-rata sebesar 54.24 kbps. Sedangkan untuk *bitrate codec* 256 kbps, dibutuhkan *bandwidth* sebesar 98.17 kbps. Pada percobaan 3, diperoleh bahwa semakin besar *background traffic* maka *bandwidth* maksimum yang dibutuhkan semakin besar.

Secara keseluruhan, analisis kebutuhan *bandwidth* maksimum untuk masing-masing skenario menunjukkan bahwa makin besar *bitrate*, *frame rate*, dan *background traffic* maka *bandwidth* maksimum yang dibutuhkan semakin besar juga. *Bandwidth* maksimum yang dibutuhkan oleh masing-masing *client* pada ketiga skenario perbedaan yang tidak signifikan. Sekali lagi hal ini dipengaruhi oleh metode transmisi yang digunakan yaitu *multicast* dimana tiap *client* mendapatkan *stream* yang sama dari *server* sehingga *bandwidth* yang dibutuhkan adalah sama untuk tiap *client*.

## V. SIMPULAN DAN SARAN

### A. Kesimpulan

- Dari pengukuran performansi *streaming server* :
  - Delta (Inter-arrival-packet Delay)*; untuk *frame rate* 15 fps diperoleh *delta* sebesar 30.2127 ms sedangkan untuk 30 fps diperoleh *delta* sebesar 7.2729 ms. Artinya nilai *delta* akan semakin kecil jika *frame rate* semakin besar. Penambahan *bit rate* menyebabkan nilai *delta* semakin kecil. Sedangkan semakin besar *background traffic* memberikan nilai *delta* yang semakin besar juga
  - Jitter* ; untuk *frame rate* yang semakin besar maka nilai *jitter* yang diperoleh semakin besar juga (0.366 ms untuk 15 fps dan 0.4325 ms untuk 30 fps). Penambahan *bitrate* menyebabkan nilai *jitter* yang semakin kecil (*bitrate* 128 kbps yaitu 0.4997 ms *bitrate* 256 kbps yaitu sebesar 0.0929 ms).
  - Packet Loss* ; untuk *frame rate* 30 fps memiliki nilai maksimal sebesar 0.025 % sedangkan 15 fps mendekati 0 %. Penambahan *bitrate* menyebabkan *packet loss* yang terjadi semakin besar (256 kbps 0.2885 % sedangkan 16 kbps 0 %). *Background traffic* yang semakin besar menyebabkan nilai *packet loss* semakin

besar. Namun nilai *ini* masih dapat diterima sebab standar *packet loss* tidak boleh melebihi 10 %.

- Throughput* ; semakin besar *frame rate* maka nilai *throughput* yang dihasilkan pun semakin besar. Nilai *throughput* terbesar diperoleh pada *bitrate* 256 kbps (0.0929 Mbps). Sedangkan nilai terkecilnya diperoleh pada penggunaan *bitrate* 64 kbps (0.0605 Mbps). Nilai *throughput* terbesar pada *background* 100 Mbps dan nilai terkecilnya adalah 0.061 kbps ( pada *background traffic* 50 Mbps).

- Semakin besar *background traffic* maka pengiriman data di jaringan dapat terganggu atau dapat menurunkan performansi. Berdasarkan pemahaman ini maka diperoleh perbandingan yang terbalik antara *background traffic* dan nilai *throughput* yang dihasilkan. Namun dari hasil pengukuran, yang terjadi adalah semakin besar *background traffic* maka nilai *throughput* yang dihasilkan semakin besar meskipun hasil yang diperoleh antara *client* 1 dan *client* 2 tidak menunjukkan perbedaan yang signifikan.
- Aplikasi *streaming* media dapat digunakan dengan baik dalam jaringan internal di Universitas Telkom . Hal ini ditunjukkan dengan tingkat performansi *streaming server* yang memenuhi persyaratan rekomendasi ITU, yaitu nilai *delta* 0 – 150 ms yang berarti *acceptable for most user application*, *jitter* yang diperoleh kurang dari 30 ms yang berarti masih dapat diterima dengan baik, *packet loss* di bawah 10 % yang berarti masih termasuk dalam rekomendasi.
- Penambahan *background traffic* yang makin besar dapat membuat QoS menjadi memburuk. Namun pemberian *background traffic* tidak membuat penurunan QoS yang signifikan, karena *background traffic* yang tidak stabil dimana pada saat diberi trafik maksimal justru *background traffic* yang mengalami penurunan pembebanan jaringan.

### B. Saran

Dipandang perlu adanya penelitian lebih lanjut akan kebutuhan *bandwidth* untuk berbagai jaringan yang belum dilakukan dalam penelitian ini, termasuk *Metro Area Network* (MAN), *Wide Area Network* (WAN), baik yang menggunakan kabel metalik, optik maupun tanpa kabel (*Wireless*).

## DAFTAR PUSTAKA

- Apostolopoulos, John G. Wai-tian Tan. Susie J. Wee, Susie J., (2002). *Video Streaming : Concepts, Algorithms and Systems*. Mobile and Media Systems Laboratory, HP Laboratories, Palo Alto, California, USA
- Austerbery, David., (2013). *The Technology of Video & Audio Streaming* , 2nd ed., Focal Press, Burlington, Massachusetts, USA.
- Croll, Alistair., Eric Packman. (1999). *Managing Bandwidth : Deploying QoS in Enterprise Networks*, 1st ed., Prentice Hall PTR Inc., USA.
- Forouzan, Bechrouz A., (2002). *Local Area Networks*, 1st ed., McGraw Hill, USA.

Jaromil., (2002). *Video streaming a research on current technologies*. Nederlands Institute voor Mediakunst, The Netherlands.

Regan, Patrick E., (2003). *Local Area Networks*, 1st ed., Prentice Hall, Inc., USA.

Sarosa, Moehammad. Anggoro, Sigit. (2000). *Jaringan Komputer : Data Link, Network & Issue.*, Teknik Sistem Komputer, Elektroteknik, ITB, Bandung.

Supandi, Dede. (2006). *Instalasi dan Konfigurasi Jaringan Komputer* Edisi Revisi. Penerbit Informatika, Bandung