

ANALISIS PERFORMANSI RTSP LIVE STREAMING SERVER BERBASIS RASPBERRY PI UNTUK VIDEO SURVEILLANCE SYSTEM

PERFORMANCE ANALYSIS RTSP LIVE STREAMING SERVER BASE ON RASPBERRY PI FOR VIDEO SURVEILLANCE SYSTEM

Muhamad Hamdan Rifai ¹, Budhi Irawan, S.Si, M.T.², Randy Erfa Saputra, S.T, M.T.³

^{1,3}Prodi S1 Sistem Komputer, Fakultas Teknik Elektro, Universitas Telkom

¹hamdan@students.telkomuniversity.ac.id,

²budhiirawan@telkomuniversity.ac.id, ³resaputra@telkomuniversity.ac.id

Abstrak

Live streaming merupakan teknologi untuk menjalankan *file video* atau *audio* secara langsung dengan sebuah *server* yang menjalankan layanan *live streaming*. *Server* yang menyediakan layanan *live streaming* akan menggunakan protokol *streaming* sebagai pengontrol dan pengirim paket data secara *real-time* kepada *client* yang menjalankan layanan *live streaming* dari *server*. Salah satu protokol *real-time* yang digunakan untuk *live streaming server* yaitu *Real Time Streaming Protocol* atau disebut RTSP, protokol tersebut berfungsi untuk berkomunikasi dengan *client* yang menjalankan *live streaming*.

Dalam Tugas Akhir ini, penulis melakukan analisis protokol RTSP yang diimplementasikan pada sebuah *raspberry pi* sebagai *live streaming server* untuk *video surveillance system* yang terintegrasi dengan *camera* pengawas. *Surveillance system* merupakan teknologi yang dipakai untuk meningkatkan pengawasan pada tempat atau lokasi yang lepas dari jangkauan penglihatan kita atau tempat tersebut sedang ditinggalkan sehingga tidak ada yang bisa mengawasi secara langsung contohnya rumah atau tempat kerja. Protokol RTSP tersebut dianalisis performansinya untuk penggunaan pada *video surveillance system*, serta analisis penggunaan *raspberry pi* sebagai *live streaming server* yang terintegrasi dengan *camera*.

Performansi protokol RTSP khususnya pada *throughput* jaringannya cenderung lebih besar dari protokol RTMP yaitu dengan nilai *throughput* terbesar 3,83 Mbps, sehingga nilai delay RTSP lebih besar dari RTMP. Dengan demikian bahwa protokol RTSP memiliki kelebihan dalam kualitas *output video* tetapi untuk performansi lebih baik RTMP karena nilai *delay* nya lebih kecil sehingga pengiriman data lebih cepat. Kehandalan *raspberry pi* sebagai *live streaming server* memiliki stabilitas yang baik saat menjalankan *video streaming* pada jaringan *video surveillance system* yang diakses oleh *user* atau *client*.

Kata kunci : *Live streaming server, RTSP, raspberry pi, video surveillance system*

Abstract

Live streaming is a technology for running a video or audio file directly with a server that is running a live streaming service. Server that provides live streaming using a streaming protocol as a controller and the sender of the packet data in real-time to a client that is running a live streaming service on the server. One of the real-time protocol used for live streaming server that is Real Time Streaming Protocol, or RTSP called, the protocol used to communicate with a client that is running live streaming.

In this final project, the author will do the analysis RTSP protocol implemented on a raspberry pi as a live streaming server for video surveillance systems integrated with camera supervisor. Surveillance system is a technology used to improve surveillance on the place or location out of reach of our vision or where it is being abandoned so that no one can supervise directly for example your home or workplace. The RTSP protocol will be analyzed its performance for use in video surveillance systems, as well as analysis of the use of raspberry pi as a live streaming server that is integrated with the camera.

RTSP protocol performance, especially on network throughput tends to be greater than the RTMP protocol. The greatest value of throughput in RTSP is 3,83 Mbps, so that the delay value is greater than RTMP. The RTSP protocol has advantages in the quality of the video output but for better performance RTMP for delay value is smaller so sending data faster. Reliability raspberry pi as a live streaming server has good stability when running video streaming on a network video surveillance system that is accessed by the user or client.

Keywords: *Live streaming server, RTSP, raspberry pi, video surveillance system*

1. Pendahuluan

Perkembangan teknologi *video surveillance system* atau *camera* pengawas berkembang sangat pesat dan sudah menjadi alat yang sangat penting dan pokok, karena kebutuhan masyarakat sekarang ini yang menginginkan keamanan yang *extra* pada tempat yang tidak dalam area pengawasan pemilik tempat tersebut [7]. Seiring dengan kebutuhan *video surveillance system* yang semakin meningkat setiap kurun waktu tertentu maka diperlukan juga tingkat kualitas layanan, performansi dan kehandalan *video streaming* itu sendiri yang bertugas menjalankan *video surveillance system*. Maka dari itu diperlukan protokol yang benar-benar handal dalam memproses *video streaming* pada *video surveillance system*, protokol yang digunakan untuk *video streaming* yaitu RTSP [2], protokol tersebut nantinya yang akan mengontrol dan mengirimkan paket *data streaming* dari *camera* pengawas ke penerima yaitu sebagai *user video surveillance system* yang akan menjalankan *video streaming* dari *camera* yang terintegrasi dengan perangkat *camera* pengawas.

Dilihat dari permasalahan diatas diperlukan analisis pada protokol RTSP sebagai protokol yang diimplementasikan pada *live streaming server* yang berupa *embedded system* yang terintegrasi pada *video surveillance system* dengan *camera* pengawas, sehingga client dapat mengetahui seberapa handalnya performansi protokol RTSP untuk implementasi *surveillance system* dibandingkan dengan protokol *streaming* lainnya seperti protokol *Real Time Messaging Protocol* atau disebut RTMP.

Pada penelitian ini akan membahas tentang permasalahan yaitu :

1. Bagaimana merancang dan mengimplementasikan protokol RTSP sebagai *live streaming server* menggunakan *embedded system* untuk *video surveillance system*?
2. Bagaimana cara pengukuran dan analisis performansi protokol RTSP sebagai *live streaming server* untuk *video surveillance system* dilihat berdasarkan pengukuran QoS jaringan?
3. Bagaimana cara mengetahui kehandalan kinerja *embedded system* sebagai *live streaming server* untuk *video surveillance system* dilihat berdasarkan utilitas pemakaian CPU ?

Selanjutnya penelitian ini bertujuan untuk melakukan perancangan dan mengimplementasikan protokol RTSP sebagai *live streaming server* menggunakan raspberry pi sebagai *embedded system* yang berfungsi sebagai server, selanjutnya mengetahui hasil analisis performansi protokol RTSP sebagai *live streaming server* pada *video surveillance system* dan mengetahui kehandalan *embedded system* dalam menjalankan *live streaming server* untuk keperluan *video surveillance system*.

2. Dasar Teori dan Perancangan Sistem

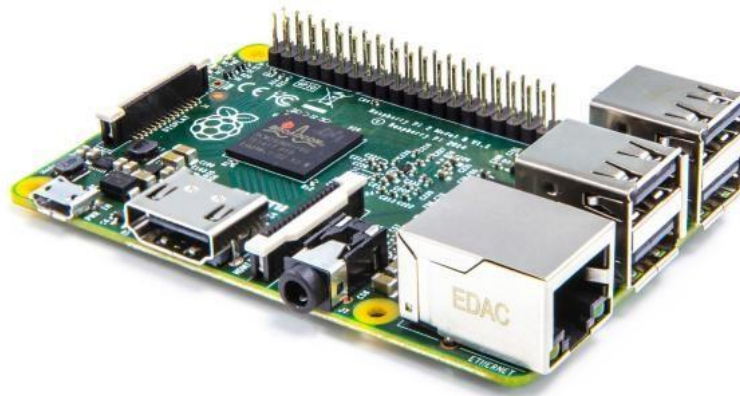
2.1 Protokol RTSP

Real Time Streaming Protocol atau RTSP merupakan protokol yang ada pada level aplikasi yang berfungsi untuk mengontrol pengiriman data dengan cara *real-time*. RTSP menggunakan dan menyediakan kerangka kerja yang *extensible* untuk mengontrol pengiriman data *on-demand* secara *real-time* seperti *audio* dan *video* [1]. Data yang berasal dari pengirim dapat mencakup *live* data dan data yang disimpan. Protokol ini berfungsi untuk mengendalikan beberapa pengiriman data seperti UDP, *multicast* UDP dan TCP. RTSP juga berfungsi sebagai penyedia mekanisme yang berdasarkan protokol *Real-Time Protocol* atau RTP [1]. RTP adalah kombinasi antara protokol RTP dan *Real-Time Transport Control Protocol* atau RTCP untuk mentransmisikan *real-time data streaming* multimedia antara *single-point* dan *multi-point*. RTP paket sangatlah banyak kegunaannya dalam mentransmisikan *data streaming*, yang termasuk kedalam header RTP diantaranya *Payload Type*, *Time Stamp*, *Sequence Number* dan *Synchronization Source Identifier* [3].

RTSP merupakan protokol yang digunakan pada layanan *streaming*, teknologi *streaming* itu sendiri merupakan sebuah teknologi yang mampu mengompresi atau menyusutkan ukuran *file audio* dan *video* agar mudah ditransfer melalui jaringan data. Aplikasi layanan *streaming* dibagi menjadi dua, yaitu *on-demand* dan *live streaming*. Layanan *streaming on-demand* seperti halnya musik atau *video* yang dijalankan langsung dari media penyimpanan *server* yang diakses melalui *client*, sedangkan *live streaming* seperti halnya *video* atau *audio* yang disiarkan secara *broadcast* dan langsung pada saat itu juga [8] [9].

2.2 Raspberry Pi

Raspberry pi merupakan sebuah karya dari yayasan nirlaba asal Inggris yaitu Raspberry Pi Foundation yang awalnya membuat perangkat ini untuk menarik minat anak-anak agar dapat menjadi seorang pengembang atau developer di bidang hardware maupun software [13]. Raspberry pi dirilis dengan lisensi Open-Source Hardware yang berarti rancangan perangkat kerasnya dapat dipelajari, dimodifikasi, didistribusi dan dijual pada publik [14]. Pada tugas akhir ini akan menggunakan raspberry pi 2 model B yang memiliki spesifikasi yaitu dengan CPU 900 MHz quad-core ARM Cortex-A7, Ethernet 100 Mbps, 4 slot USB 2.0, video out HDMI 1.4 dan card slot menggunakan micro SD [13].



Gambar 1 Raspberry Pi 2 Model B

2.3. Quality of Service (QoS)

Pada saat ini komunikasi dapat dilakukan melalui jaringan internet, tetapi bukan berarti komunikasi data tidak memiliki hambatan. Masalah rendahnya kualitas komunikasi berhubungan erat dengan permasalahan jaringan IP yaitu berkaitan dengan rendahnya bandwidth dari jaringan tersebut. Sehingga dapat diartikan kita sebagai pengguna jaringan harus mengetahui permasalahan kualitas komunikasi dan mendapatkan solusi agar mendapat QoS yang lebih baik [8]. QoS itu sendiri merupakan kemampuan suatu jaringan untuk memberikan layanan yang lebih baik pada trafik data tertentu pada berbagai jenis platform teknologi yang di terapkan. Parameter QoS diantaranya throughput, delay, jitter dan packet loss [8].

a. Throughput

Merupakan kecepatan (*rate*) transfer data yang efektif yang diukur dalam *bit persecond* (bps). *Header* dalam paket data dapat mengurangi nilai *throughput*. *Bandwidth* yang disediakan tidak semua dipakai pada jaringan. Besarnya *bandwidth* yang dipakai dalam jaringan yang berjalan saat itu juga adalah *throughput* [8] [11].

$$\Sigma$$

b. Delay

Delay merupakan total waktu yang dibutuhkan paket data untuk menempuh jarak dari asal (*source*) ke penerima atau tujuan (*destination*). *Delay* dari pengirim ke penerima pada dasarnya tersusun oleh *hardware latency*, *delay* akses, dan *delay* transmisi. *Delay* yang paling sering terjadi oleh *traffic* yang lewat adalah *delay* transmisi [8] [11].

$$\Sigma$$

c. Jitter

Jitter merupakan variasi kedatangan paket data, 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 [8] [11].

$$\frac{\Sigma}{\Sigma}$$

d. Packet Loss

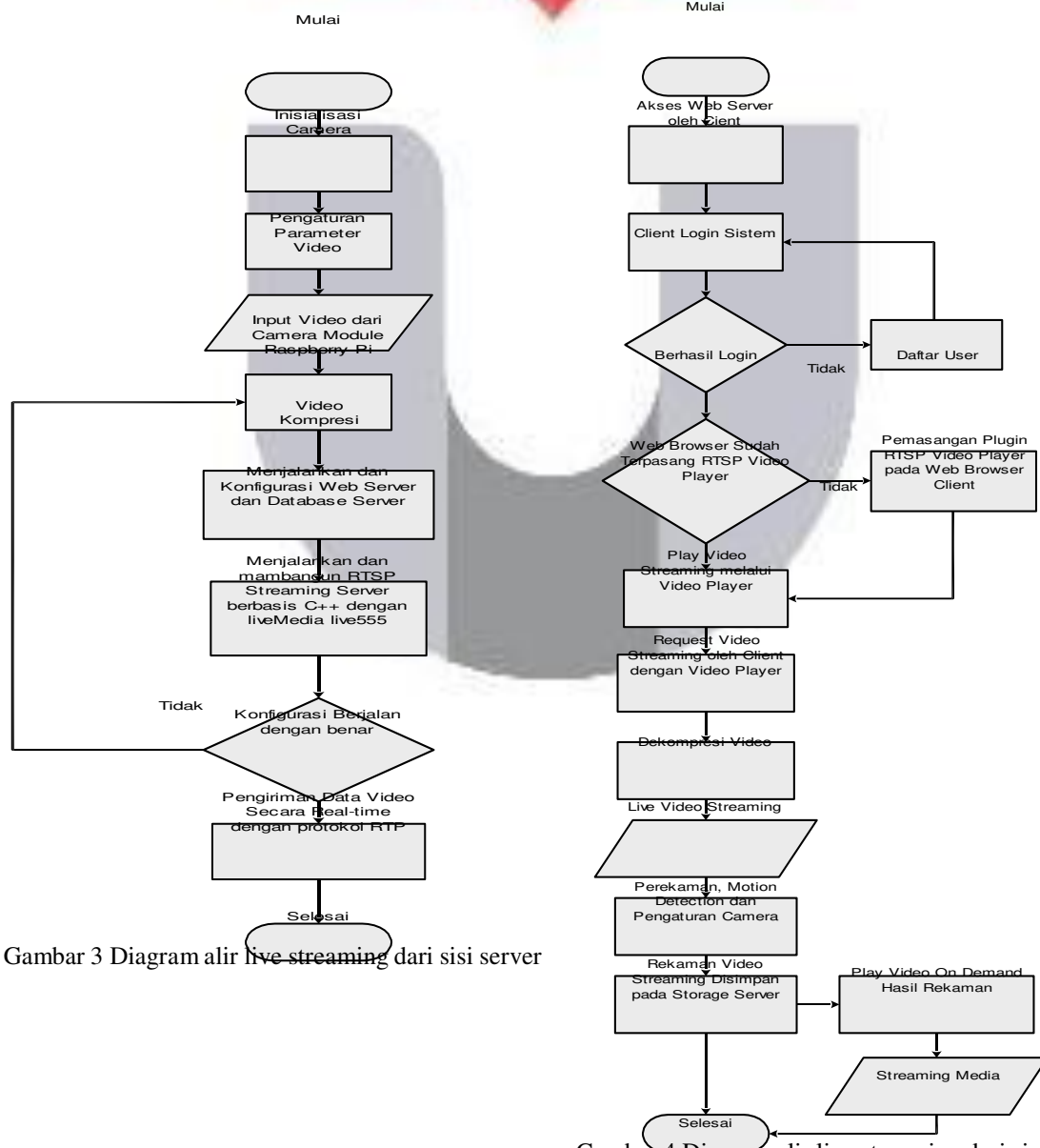
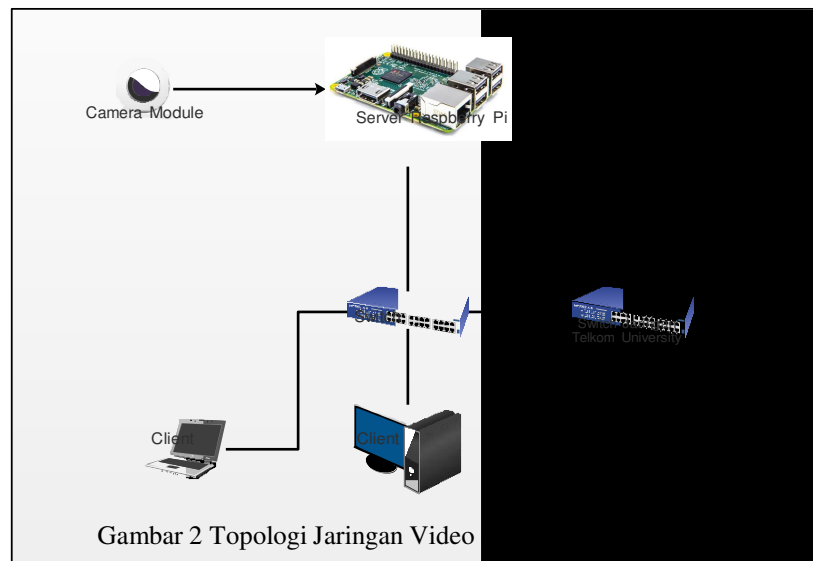
Packet loss adalah kegagalan transmisi paket data yang akan mencapai tujuannya atau jumlah paket data yang hilang. Kegagalan paket data tersebut untuk mencapai tujuannya dapat disebabkan oleh penurunan *signal* dalam media jaringan, melebihi batas saturasi jaringan, paket data yang *corrupt* yang menolak untuk transit dan bisa juga karena kesalahan *hardware* jaringan [8] [11].

$$\frac{\Sigma}{\Sigma}$$

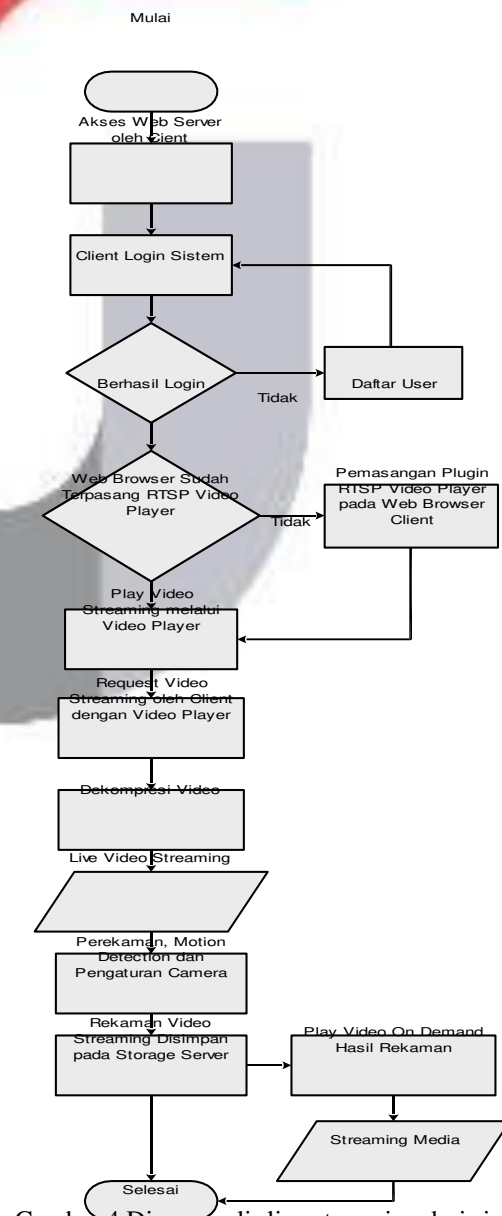
2.4. Video Streaming Server

Video streaming server adalah sebuah web server atau aplikasi server yang didalamnya dapat digunakan untuk menjalankan video secara real-time. Video streaming merupakan teknologi yang dapat memungkinkan untuk mentransmisikan data berupa video. Pada teknologi video streaming juga mampu mengompresi atau menyusutkan ukuran file video agar lebih mudah dikirimkan pada jaringan internet [12].

2.5. Desain Sistem



Gambar 3 Diagram alir live streaming dari sisi server



Gambar 4 Diagram alir live streaming dari sisi client

3. Pengujian dan Analisis

3.1. Pengujian Performansi QoS

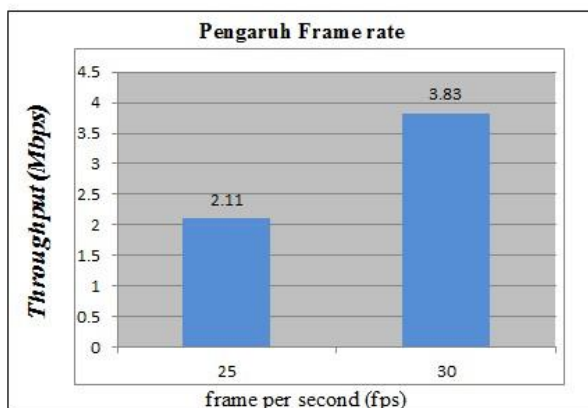
Pada pengujian pengukuran performansi QoS protokol RTSP live streaming ini akan diuji dengan berdasarkan beberapa skenario yang telah ditentukan yaitu :

- Skenario Pengujian 1 (Pengaruh frame rate dan bitrate)**
Pada pengujian ini akan menggunakan parameter video live streaming dengan resolusi video 1280x720 (HD) dan video codec H.264 dengan nilai frame rate yang berbeda yaitu 25 fps dan 30 fps. Sedangkan untuk nilai variasi bitrate yaitu 64 Kbps, 128 Kbps dan 256 Kbps.
- Skenario Pengujian 2 (Pengaruh Background Traffic pada bitrate)**
Pada pengujian ini akan menggunakan parameter video live streaming dengan resolusi video 1280x720 (HD) dan video codec H.264 dengan nilai variasi bitrate yaitu 64 Kbps, 128 Kbps dan 256 Kbps pada background traffic 20 Mbps, 30 Mbps dan 40 Mbps.
- Skenario Pengujian 3 (Pengaruh Background Traffic pada frame rate)**
Pada pengujian ini akan menggunakan parameter video live streaming dengan resolusi video 1280x720 (HD) dan video codec H.264 serta server diakses oleh client dengan nilai frame rate yang berbeda yaitu 25 fps dan 30 fps pada background traffic 20 Mbps, 30 Mbps dan 40 Mbps.

3.1.1. Pengukuran Throughput

Tabel 1 Nilai Throughput Skenario 1

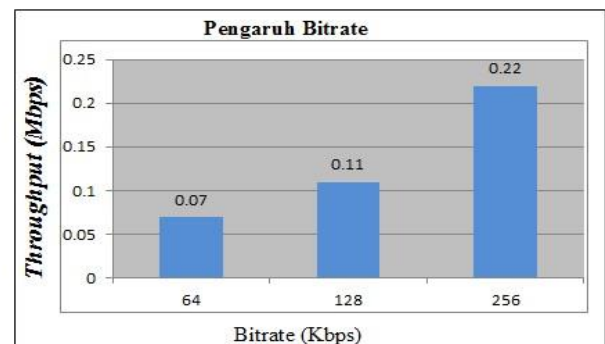
Frame rate (fps)	Throughput (Mbps)
25	2,11
30	3,83



Gambar 5 Grafik Throughput Skenario 1

Tabel 2 Nilai Throughput Skenario 1

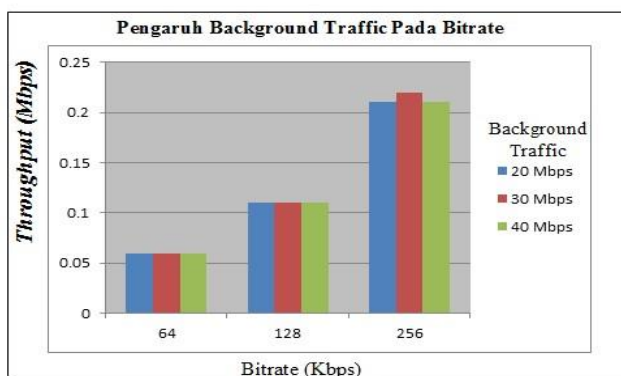
Bitrate (Kbps)	Throughput (Mbps)
64	0,07
128	0,11
256	0,22



Gambar 6 Grafik Throughput Skenario 1

Tabel 3 Nilai Throughput Skenario 2

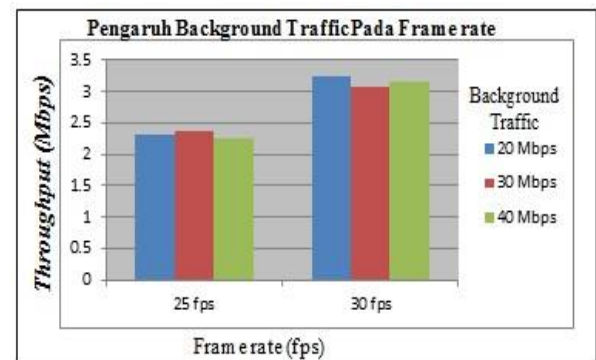
Throughput (Mbps)	Background Traffic (Mbps)		
	20	30	40
64 Kbps	0,06	0,06	0,06
128 Kbps	0,11	0,11	0,11
256 Kbps	0,21	0,22	0,21



Gambar 7 Grafik Throughput Skenario 2

Tabel 4 Nilai Throughput Skenario 3

Throughput (Mbps)	Background Traffic (Mbps)		
	20	30	40
25 fps	2,30	2,37	2,27
30 fps	3,25	3,07	3,15

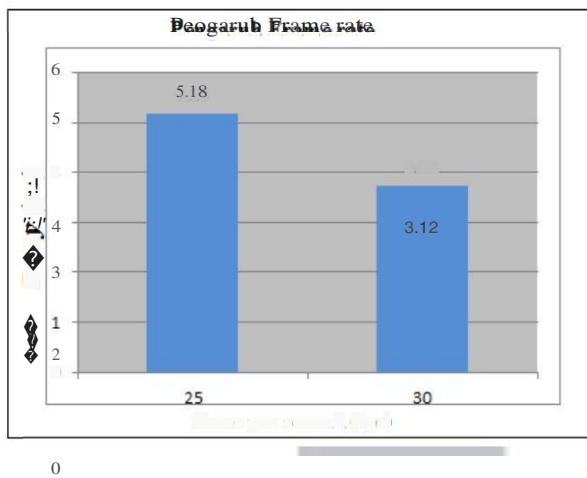


Gambar 8 Grafik Throughput Skenario 3

3.1.2. Pengukuran Delay

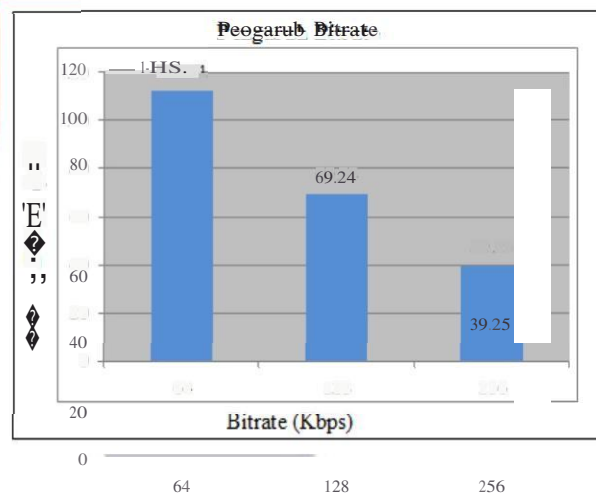
Tabel 5 Nilai Delay Skenario 1

Frame rate (fps)	Delay (ms)
25	5,18
30	3,72



Tabel 6 Nilai Delay Skenario 1

Bitrate (Kbps)	Delay (ms)
64	112,51
128	69,24
256	39,25



Gambar 9 Grafik Delay Skenario 1

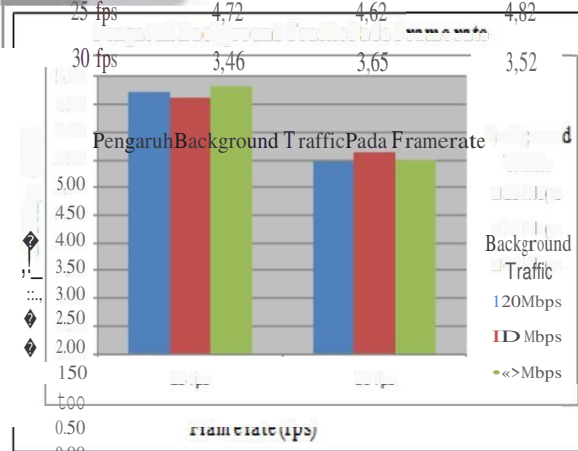
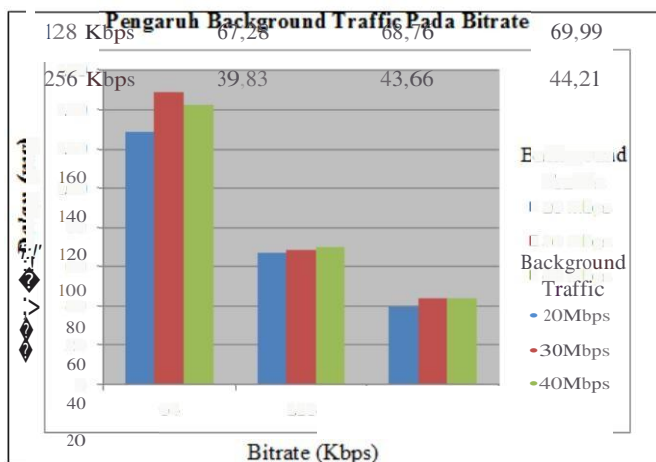
Tabel 7 Nilai Delay Skenario 2

	Background Traffic (Mbps)		
Delay (ms)	20	30	40
64 Kbps	128,40	149,33	142,50

Gambar 10 Grafik Delay Skenario 1

Tabel 8 Nilai Delay Skenario 3

	Background Traffic (Mbps)		
Delay (ms)	20	30	40



3.1.3. Pengukuran Jitter

Gambar 11 Grafik Delay Skenario 2

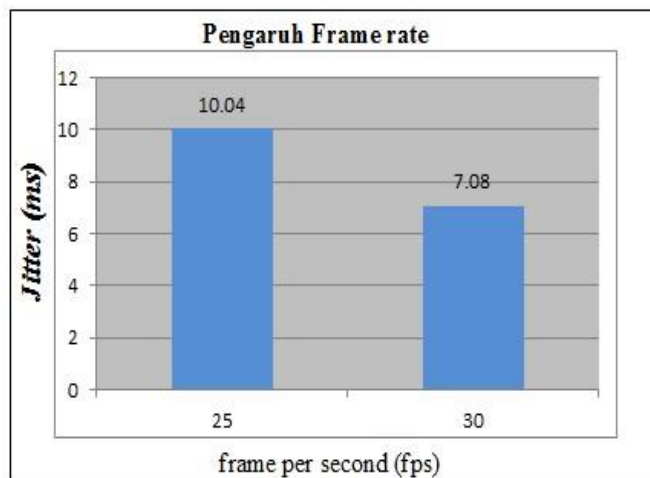
Tabel 9 Nilai Jitter Skenario 1

Frame rate (fps)	Jitter (ms)
25	10,04
30	7,08

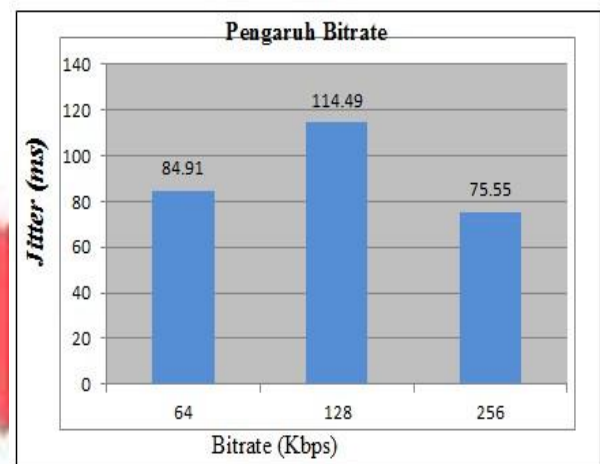
Gambar 12 Grafik Delay Skenario 3

Tabel 10 Nilai Jitter Skenario 1

Bitrate (Kbps)	Jitter (ms)
64	84,91
128	114,49
256	75,55



Gambar 13 Grafik Jitter Skenario 1



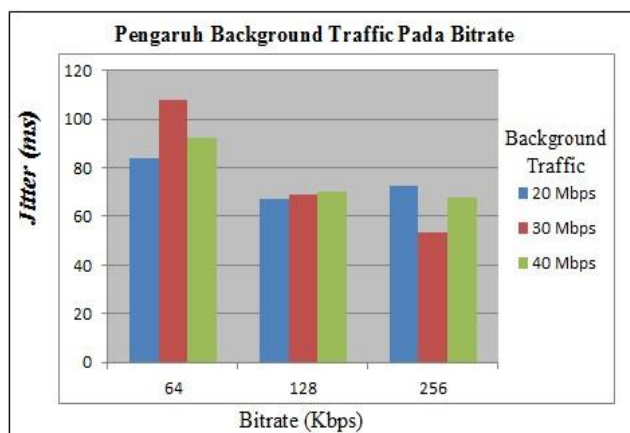
Gambar 14 Grafik Jitter Skenario 1

Tabel 11 Nilai Jitter Skenario 2

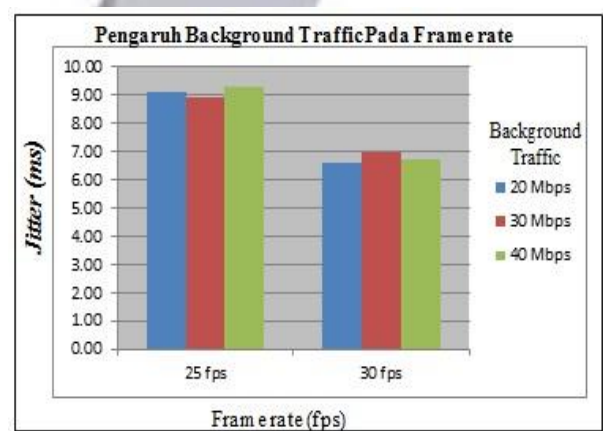
Jitter (ms)	Background Traffic (Mbps)		
	20	30	40
64 Kbps	83,88	108,06	92,45
128 Kbps	67,28	68,76	69,99
256 Kbps	72,66	53,50	67,62

Tabel 12 Nilai Jitter Skenario 3

Jitter (ms)	Background Traffic (Mbps)		
	20	30	40
25 fps	9,12	8,92	9,31
30 fps	6,59	6,98	6,76



Gambar 15 Grafik Jitter Skenario 2



Gambar 16 Grafik Jitter Skenario 3

3.2. Analisis Hasil Pengujian QoS

Pada ketiga skenario pengujian diatas nilai throughput memiliki nilai yang selalu naik apabila nilai frame rate atau bitrate semakin besar, dikarenakan pengiriman bit semakin besar apabila nilai frame rate atau bitrate semakin besar sehingga jumlah data semakin banyak maka throughput jaringan juga akan mengalami kenaikan nilai secara signifikan. Pada ketiga pengujian diatas didapatkan nilai throughput tertinggi yaitu 3,83 Mbps dan terendah 0,06 Mbps.

Pada ketiga skenario pengukuran pengujian diatas semua nilai delay lebih kecil dari 150 ms yang merupakan kategori delay sangat baik menurut standar ITU-T G.1010 yang menyebutkan bahwa delay untuk komunikasi multimedia yang direkomendasikan yaitu < 150 ms. Dengan nilai delay tertinggi yaitu 149,33 ms dan nilai delay terendah yaitu 3,46 ms. Untuk jitter pada pengukuran pengujian diatas yang nilainya paling tinggi yaitu 114,49 ms dan yang paling rendah yaitu 6,59 ms.

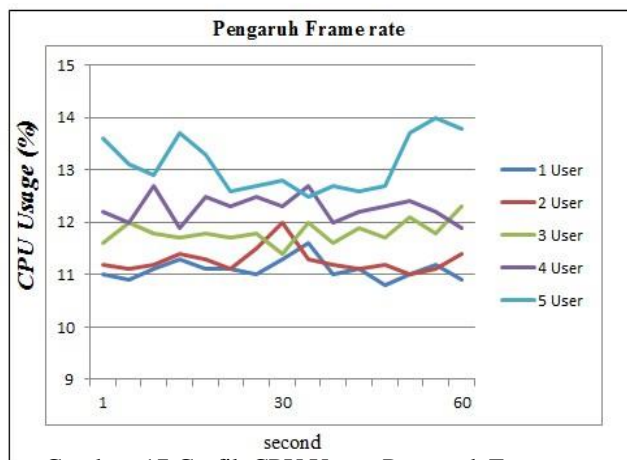
Pada ketiga skenario pengukuran pengujian diatas tidak dicantumkan hasil pengukuran packet loss karena protokol video live streaming RTSP menggunakan transport protokol TCP yang merupakan protokol yang bersifat connection oriented sehingga paket yang dikirimkan akan dijamin sampai ke penerima maka nilai packet loss rata-rata keseluruhan ketiga skenario pengujian diatas adalah 0,00%.

Untuk pengaruh *background traffic* pada keseluruhan skenario pengujian tidak terlalu berpengaruh pada nilai *throughput*, *delay* dan *jitter*, dikarenakan *background traffic* saat mengirimkan paket data pada jaringan *video surveillance system* cenderung tidak stabil, sehingga nilai *background traffic* yang telah dibangkitkan belum tentu paket

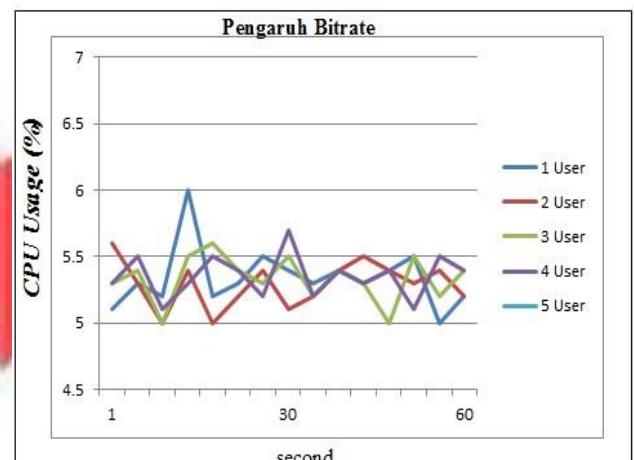
datanya terkirim semua pada jaringan. Hal ini disebabkan karena *background traffic* yang berupa protokol UDP memiliki karakteristik *connectionless* sehingga tidak ada pembangunan koneksi terlebih dahulu saat pengiriman paket data. Pada hasil keseluruhan skenario pengujian dipengaruhi juga oleh beberapa faktor yaitu intensitas cahaya, perubahan warna dan banyaknya pergerakan objek saat camera mengambil gambar yang menyebabkan paket data yang berupa bit mengalami perbedaan dikarenakan video streaming yang dijalankan secara live.

Performansi dari perbandingan protokol dilihat dari RTSP dan protokol streaming lain yaitu RTMP dilihat dari nilai *throughput* jaringannya maka nilai *throughput* RTSP lebih besar dibandingkan dengan nilai *throughput* RTMP dikarenakan pada proses pengiriman data RTSP hanya mengirimkan *message* yang berupa *request message* dan *response message*. *Request message* yaitu berupa pesan dari client ke server sedangkan *response message* berupa pesan dari server ke client, sedangkan protokol RTMP memiliki beberapa tahap untuk membangun koneksi pengiriman data yaitu berupa *Handshaking*, *create connection*, *create stream* dan *Play* yang akan lebih menjamin pengiriman bit. Selain itu pada protokol RTMP pengiriman data berupa *message* yang dibagi kedalam beberapa potongan-potongan sebelum di transmisikan. Oleh karena itu RTSP memiliki nilai *message* yang lebih kecil dalam membangun koneksi yang menyebabkan aliran data menjadi lebih cepat sehingga nilai *throughput* menjadi lebih besar. Selain itu protokol RTSP lebih stabil dalam pengiriman data.

3.3. Pengujian Performansi Raspberry Pi



Gambar 17 Grafik CPU Usage Pengaruh Frame rate



Gambar 18 Grafik CPU Usage Pengaruh bitrate

Raspberry pi sebagai live streaming server memiliki stabilitas yang baik saat menjalankan live streaming yang diakses oleh client video surveillance system.

4. Kesimpulan

1. *Delay* pada keseluruhan skenario pengujian sudah memenuhi standar ITU-T G.1010 dengan nilai lebih kecil dari 150 ms yang merupakan kategori baik. Untuk nilai *jitter* pada keseluruhan skenario pengujian yang nilainya paling tinggi yaitu 114,49 ms dan yang paling rendah yaitu 6,59 ms.
2. Pengaruh *background traffic* tidak terlalu signifikan pada jaringan, sehingga *background traffic* tidak mempengaruhi nilai penurunan QoS pada jaringan *video surveillance system*.
3. *Raspberry pi* sebagai *embedded system* yang menjalankan *live streaming server* memiliki stabilitas yang baik saat menjalankan dan mengelola *video streaming* yang diakses oleh *client video surveillance system*.
4. Pada protokol RTSP, *throughput* jaringan cenderung lebih besar dari protokol RTMP, sehingga nilai *delay* RTSP lebih besar dari RTMP. Dengan demikian bahwa protokol RTSP memiliki kelebihan dalam kualitas *output* video tetapi untuk performansi lebih baik RTMP.

5. Saran

Saran yang dapat diberikan untuk penelitian lebih lanjut tentang topik ini diantaranya adalah penggunaan *embedded system* selain *raspberry pi*, pengembangan aplikasi live streaming pada perangkat mobile.

Daftar Pustaka:

- [1] H. Schulzrinne, R. Lanphier, and A. Rao. *Real Time Streaming Protocol (RTSP)*. RFC 2326, *Internet Engineering Task Force (IETF)*. 1998. Page 1
- [2] YJ Lee, OG Min, and HY Kim. *Performance Evaluation Technique of the RTSP based Streaming Server*. *Fourth Annual ACIS International Conference on Computer and Information Science (ICIS'05)*. 2005. Volume 00. Page 414-417

- [3] Yong Liu, Bin Du, Shuling Wang, Haibo Yang, and Xingwei Wang. *Design and Implementation of Performance Testing Utility for RTSP Streaming Media Server*. 2010 First International Conference on Pervasive Computing, Signal Processing and Application. 2010. Page 193-196
- [4] Dian Chu, Chun-hua Jiang, Zong-bo Hao, and Wei Jiang. *The Design and Implementation of Video Surveillance System Based on H.264, SIP, RTP/RTCP and RTSP*. 2013 Sixth International Symposium on Computational Intelligence and Design. 2013. Page 39-43
- [5] Prasad, Sanjana., P., Mahalakshmi., Clement, A. John., Swathi, R. *Smart Surveillance Monitoring System Using Raspberry Pi and PIR Sensor*. International Journal of Computer Science and Information Technologies. Volume 5. 2014. Page 7107-7109.
- [6] Fernando, Erick. *Automasi Smart Home dengan Raspberry Pi dan Smartphone Android*. Jurnal Konferensi Nasional Ilmu Komputer (KONIK). 2014. ISSN:2338-2899.
- [7] Adriansyah, Andi., Rizki G.M., Mirzanu., Yulita. *Rancang Bangun dan Analisa CCTV Online Berbasis Raspberry Pi*. Jurnal SINERGI. Volume.18(2). 2014. Halaman 105-110.
- [8] I Dwi, Anggelina., M., R. Ruman., Wahidah, Ida. *Analisis Kualitas Layanan Video Live Streaming Pada Jaringan Lokal Universitas Telkom*. Jurnal Pusat Penelitian dan Pengembangan Sumber Daya dan Perangkat Pos dan Informatika. Volume 12. 2014. ISSN: 16930991.
- [9] Diharso, Adi Setio. *Analisa Perbandingan Performansi Video Streaming Antara Darwin Streaming Server dengan Red5*. Jurnal komputer dan komunikasi. 2013.
- [10] Wei, Chongyu., Zhang, Honglin. *Applications of a Streaming Video Server in a Mobile Phone Live Streaming System*. Journal of Software Engineering and Applications. Volume 7. 2014. Page 975-982.
- [11] Mehta, Vikram., Dr. Gupta, Neena. *Performance Analysis of QOS Parameter for Wimax Networks*. International Journal of Engineering Innovative Technology (IJEIT). Volume 1. 2012. Page 105-110
- [12] Kristanto, Endi Dwi. *Video Streaming Menggunakan VLC*. Jurnal Ilmu Komputer. 2013. Halaman 1-7.
- [13] Velthuis, P.J.E. *Small Data Center Using Raspberry Pi 2 for Video Streaming*. International Journal of Engineering Sciences. 2016
- [14] Dawood, Rahmad., Qiana, Said Fairuz., Muchallil, Sayed. *Kelayakan Raspberry Pi Sebagai Web Server: Perbandingan Kinerja Nginx, Apache, Lighttpd pada platform Raspberry Pi*. Jurnal Rekayasa Elektronika. 2014. Halaman 25-29.
- [15] Prasetya, Bayu Aditiya. *Pengaruh Video Bitrate dan Background Traffic Terhadap Kinerja Video Streaming Pada Jaringan Wireless LAN*. Jurnal Departemen Ilmu Komputer IPB. 2008. Halaman 1-10.
- [16] Motion Vector
http://www.webopedia.com/TERM/M/motion_vector.html
- [17] Halsall, Fred. 2001. *Multimedia Communications*. Harlow, England: Addison-Wesley.