

# Analisa Kinerja Jaringan *Provider* untuk Aplikasi *Video Chatting* (Studi Kasus di daerah Marpoyan)

Jeri Dwi Danur\*, Febrizal\*\*

\*Alumni Teknik Elektro Universitas Riau, \*\*Jurusan Teknik Elektro Universitas Riau  
Kampus Bina Widya Km 12,5 Simpang Baru Panam Pekanbaru, 28293  
Jurusan Teknik Elektro Universitas Riau  
E-mail : jeri2danur@gmail.com

## ABSTRACT

*Video chat is a communication that allows users to chat and can see the other person's face directly. Video chatting requires a good network to transmit audio and video data in a short time. This paper examines the performance of three mobile network provider (Provider A and Provider B Provider C) in a video chat in the area Marpoyan and the effect of time when doing video chat on the performance of each provider. The method used is the method using packet sniffing software wireshark to see Quality of Service (QoS) generated by each provider. QoS parameters analyzed such as delay, jitter, packet loss and throughput. Analysis results of data from experiments conducted showed that the best time to do video chat for three provider is at early morning (at 00:00 to 06:00), while the worst time to do video chat on Provider A and Provider B at daytime (12:00 to 18:00) and Provider C at night time (18:00 to 00:00).*

**Keyword:** *Video chatting, QoS, Packet sniffing*

## I. Pendahuluan

Dengan perkembangan ilmu pengetahuan dan teknologi, perkembangan telekomunikasi juga berubah pesat, yang awalnya komunikasi hanya menggunakan tanda, suara bahkan sekarang bisa mencapai komunikasi data, gambar bahkan *video*. Salah satu diantara teknologi tersebut adalah *video chatting*. *Video chatting* dapat menyalurkan gambar dan suara penggunanya pada saat berkomunikasi dalam bentuk *video real time*. *Video chatting* dapat dilakukan menggunakan Laptop atau *PC (Personal Computer)* yang lengkap dan memiliki *webcam, speaker, microphone* dan aplikasi penunjang *video chatting*. Salah satu aplikasi *video chatting* yang cukup dikenal dimasyarakat adalah Line. Line adalah *software* aplikasi komunikasi yang berbasis *IP (Internet Protocol)*, terhubung melalui internet, dan menghubungkan sesama pengguna line. Pada saat melakukan *video*

*chatting* beban trafik pada jaringan internet akan bertambah besar, untuk melihat beban trafik di internet membutuhkan *software* monitoring yang handal, salah satunya wireshark. Wireshark adalah sebuah aplikasi yang mampu *capture* paket-paket data yang terjadi selama proses komunikasi *point to point*. Setelah paket-paket tersebut *capture* maka hasil yang diperoleh dikalkulasikan untuk menganalisa parameter *QoS (Quality of Service)* antara lain *delay, packet loss* dan *throughput*. Marpoyan merupakan tempat dimana pengguna jasa telekomunikasi cukup banyak, karena di Marpoyan terdapat salah satu universitas swasta yang menjadi salah satu pilihan tempat kuliah bagi warga Pekanbaru dan dari luar Pekanbaru. Disekitar universitas tersebut juga terdapat kos-kosan tempat tinggal mahasiswa yang membuat pengguna layanan telekomunikasi sangat padat didaerah tersebut. Untuk itu diwilayah Marpoyan sangat dibutuhkan *provider* yang

handal, yang mampu mengatasi banyaknya penggunaan jaringan. Pada tulisan ini membahas tentang analisa kinerja jaringan *provider* untuk aplikasi *video chatting* yang dilakukan di daerah marpoyan. Parameter yang akan dianalisa adalah *delay*, *jitter*, *packetloss* dan *throughput*.

## II. Dasar Teori

### 2.1. Video chatting

*Video chat* adalah salah satu media telekomunikasi yang memberikan kemudahan kepada pengguna untuk dapat melihat wajah lawan bicara dalam *chatting* dengan menggunakan laptop atau PC yang memiliki perangkat webcam, loudspeaker, monitor dan microphone. *Video chatting* juga membutuhkan jaringan internet sebagai media transmisinya.

### 2.2. Protokol Transfer Data VoIP

Dalam aplikasi *VoIP* (Voice over Internet Protocol) terdapat beberapa protokol yang ikut berperan dalam proses transfer paket data, antara lain:

#### 1. TCP (Transmission Control Protocol)

*TCP* merupakan protokol yang bersifat connection oriented, artinya menjaga reabilitas hubungan komunikasi end-to-end. Pada *TCP* terdapat sinyal *ACK* (*ACKnowledgement*) sebagai balasan positif dari penerima kepada pengirim paket bahwa paket yang dikirim sudah diterima. Jika *ACK* tidak diterima dalam rentang waktu tertentu maka paket akan dikirim ulang. Pada transfer data *VoIP*, *TCP* digunakan pada proses signalling.

#### 2. UDP (User Datagram Protocol)

*UDP* merupakan protokol yang lebih sederhana dibandingkan *TCP*. *UDP* digunakan untuk situasi yang tidak mementingkan mekanisme reabilitas. *UDP* lebih mementingkan kecepatan transfer

data tanpa memperhatikan adanya paket yang hilang walaupun mencapai 50% dari jumlah paket yang harus dikirim.

#### 3. STUN (Simple Transversal UDP through NAT)

*STUN* adalah sebuah protokol ringan yang memungkinkan aplikasi menemukan tipe NAT (Network Address Translator) dan firewall antara private dan public internet. Protokol *STUN* Memiliki cara kerja yang sama dengan *UDP*. Beberapa aplikasi *video chatting* menggunakan *STUN* untuk mengirimkan paket audio streaming dan *VoIP*.

### 2.3. QoS (Quality of Service)

*QoS* merupakan kemampuan suatu network menyediakan service lebih baik untuk user dalam membagi bandwidth sesuai kebutuhan data dan voice yang digunakan. Parameter *QoS*, antara lain:

#### 1. Delay (Waktu tunda)

*Delay* merupakan akumulasi dari berbagai waktu tunda dari ujung ke ujung pada jaringan internet. *Delay* menjadi acuan waktu transmisi paket dari pengirim hingga ke penerima. Berikut merupakan standar *delay* menurut ITU-T G.114.

Tabel 2.1. Standar kualitas *Delay*.

Delay (ms)	Kualitas
0 - 150	Baik
150 - 400	Cukup, masih dapat diterima
> 400	Buruk

Untuk menghitung nilai *delay*, dapat menggunakan persamaan sebagai berikut:

$$Delay = \frac{\text{waktu antar paket}}{\text{jumlah paket}} \quad (2.1)$$

#### 2. Jitter (Variasi waktu tunda)

*Jitter* merupakan perbedaan selang waktu kedatangan antar paket. *Jitter* dapat disebabkan oleh terjadinya kongesti, kurangnya kapasitas jaringan, variasi ukuran paket serta ketidak urutan paket. Berikut

merupakan standar *jitter* menurut ITU-T G.114.

Tabel 2.2. Standar kualitas *Jitter*.

<i>Jitter</i> (ms)	Kualitas
0 - 20	Baik
20 - 50	Dapat diterima
> 50	Tidak dapat diterima

Untuk menghitung besarnya nilai *jitter* dapat menggunakan persamaan berikut :

$$Jitter = \frac{Total\ Variasi\ Delay}{Jumlah\ Paket - 1} \quad (2.2)$$

Total variasi *delay* merupakan jumlah dari selisih tiap nilai *delay*, dapat dihitung menggunakan persamaan berikut :

$$Total\ Variasi\ Delay = |delay2 - delay1| + |delay3 - delay2...|delayn - delay(n-1) \quad (2.3)$$

### 3. Packet loss (Paket hilang)

*Packet loss* merupakan penyebab utama pelemahan transfer *VoIP*. *Packet loss* terjadi karena pembuangan paket dalam jaringan (network loss) atau pembuangan paket di gateway samapai kedatangan terakhir (late loss). Berikut merupakan standar *packet loss* menurut ITU-T G.114.

Tabel 2.3. Standar kualitas *Packet loss*.

<i>Packet Loss</i> (%)	Kualitas
0 - 1 %	Baik
1 - 5 %	Dapat diterima
> 10 %	Tidak dapat diterima

Untuk menghitung nilai *packet loss* dapat menggunakan persamaan berikut:

$$Packet\ loss = \frac{(Total\ Packet - Paket\ Terkirim)}{Paket\ total\ tercapture} \times 100\% \quad (2.4)$$

### 4. Throughput

*Throughput* merupakan kecepatan transfer efektif yang diukur dalam bit per second. *Throughput* merupakan total jumlah bit paket

selama transfer dibagi dengan durasi selang waktu transfer tersebut. Untuk menghitung nilai *throughput* dapat menggunakan persamaan berikut :

$$Throughput = \frac{panjang\ paket\ diterima}{waktu\ antar\ paket} \quad (2.5)$$

### 2.4. Wireshark

Wireshark adalah salah satu *software* penyaring paket yang digunakan untuk menganalisa sebuah *trafik* pada jaringan. Wireshark dapat melihat *trafik* yang menuju sebuah alamat *interface*, tidak hanya itu bahkan wireshark dapat melihat semua *trafik broadcast* dan juga *multicast*. Metode yang digunakan wireshark adalah *packet sniffing* yang mana wireshark memiliki kemampuan untuk *capture* semua paket yang dikirim dan diterima sebuah *interface* dalam jaringan kemudian di *decodekan* untuk di analisa.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.00000000	203.104.172.226	10.53.13.181	STUN	204	ChannelData TURN Message
2	0.00303100	203.104.172.226	10.53.13.181	STUN	1168	ChannelData TURN Message
3	0.00341200	203.104.172.226	10.53.13.181	STUN	1168	ChannelData TURN Message
4	0.03300800	203.104.172.226	10.53.13.181	STUN	1168	ChannelData TURN Message
5	0.04332100	203.104.172.226	10.53.13.181	STUN	1168	ChannelData TURN Message
6	0.04363300	203.104.172.226	10.53.13.181	STUN	740	ChannelData TURN Message
7	0.04374600	203.104.172.226	10.53.13.181	STUN	191	ChannelData TURN Message
8	0.04499400	203.104.172.226	10.53.13.181	STUN	208	ChannelData TURN Message
9	0.05936600	10.53.13.181	203.104.172.226	STUN	244	ChannelData TURN Message
10	0.09368600	203.104.172.226	10.53.13.181	STUN	1168	ChannelData TURN Message
11	0.12192300	10.53.13.181	203.104.172.226	STUN	258	ChannelData TURN Message
12	0.12326700	203.104.172.226	10.53.13.181	STUN	1168	ChannelData TURN Message
13	0.12364400	203.104.172.226	10.53.13.181	STUN	1168	ChannelData TURN Message
14	0.13327900	203.104.172.226	10.53.13.181	STUN	1168	ChannelData TURN Message
15	0.13361600	203.104.172.226	10.53.13.181	STUN	792	ChannelData TURN Message
16	0.13412600	10.53.13.181	203.104.172.226	STUN	1168	ChannelData TURN Message
17	0.13499400	203.104.172.226	10.53.13.181	STUN	183	ChannelData TURN Message
18	0.13601900	10.53.13.181	203.104.172.226	STUN	1168	ChannelData TURN Message
19	0.13641400	10.53.13.181	203.104.172.226	STUN	1168	ChannelData TURN Message
20	0.13656800	10.53.13.181	203.104.172.226	STUN	527	ChannelData TURN Message
21	0.14595900	10.53.13.181	203.104.172.226	STUN	112	ChannelData TURN Message
22	0.15302100	203.104.172.226	10.53.13.181	STUN	178	ChannelData TURN Message
23	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
24	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
25	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
26	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
27	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
28	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
29	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
30	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
31	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
32	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
33	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
34	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
35	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
36	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
37	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
38	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
39	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
40	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
41	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
42	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
43	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
44	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
45	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
46	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
47	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
48	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
49	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
50	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
51	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
52	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
53	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
54	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
55	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
56	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
57	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
58	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
59	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
60	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
61	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
62	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
63	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
64	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
65	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
66	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
67	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
68	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
69	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
70	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
71	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
72	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
73	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
74	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
75	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
76	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
77	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
78	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
79	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
80	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
81	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
82	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
83	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
84	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
85	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
86	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
87	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
88	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
89	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
90	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
91	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
92	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
93	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
94	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
95	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
96	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
97	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
98	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
99	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message
100	0.15355500	10.53.13.181	203.104.172.226	STUN	222	ChannelData TURN Message

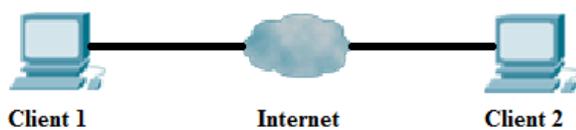
Gambar 2.1. Tampilan *capture* paket pada wireshark

Gambar 2.1 menunjukkan informasi dari berbagai paket yang diperoleh oleh *software* wireshark. Informasi tersebut berupa sumber paket, tujuan paket, tipe paket, panjang bit dan waktu *capture* paket.

### III. Metode Penelitian

Penelitian dilakukan disebuah tempat di daerah Marpoyan menggunakan tiga *provider* berbeda yaitu *provider A*, *provider B* dan *provider C*. Waktu penelitian dilakukan

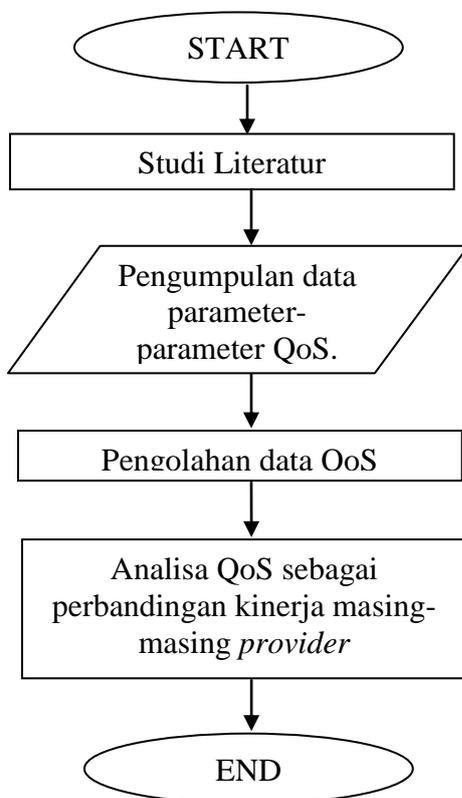
dalam pembagian waktu yaitu dini hari (00.00-06.00), pagi (06.00-12.00), siang (12.00-18.00) dan malam (18.00-00.00) dengan lamanya waktu melakukan *video chatting* maksimal 15 menit. Dalam masing-masing pembagian waktu dilakukan 3 kali pengambilan untuk setiap *provider* yang kemudian data yang diperoleh akan dirata-ratakan. Pengujian menggunakan 2 client dengan model pengujian seperti pada gambar 3.1.



Gambar 3.1 Model Pengujian

### 3.1. Flowchart penelitian

Dalam melakukan analisa kinerja jaringan *provider* telekomunikasi seluler untuk aplikasi *video chatting* di daerah Marpoyan, diperlukan diagram alir yang berisi tahapan-tahapan untuk membantu proses penelitian. Diagram alir tersebut di tunjukkan seperti pada gambar 3.2.



Gambar 3.2. Flowchart penelitian

Penelitian ini dimulai dengan mengumpulkan literatur dan teori yang mendukung hal-hal yang akan diamati dalam penelitian. Kemudian penelitian ini dilanjutkan dengan pengumpulan data-data yang dilakukan menggunakan software wireshark, lalu data yang sudah diperoleh akan di kalkulasikan dengan persamaan matematika dengan menggunakan Ms. Excell untuk mendapatkan nilai parameter QoS berupa delay, jitter, packet loss dan throughput. Kemudian hasil parameter QoS akan dianalisa dan dibandingkan dengan standar ITU-T G.114.

## IV. Hasil dan Pembahasan

### 4.1. Delay

Perhitungan *delay* dilakukan untuk mendapatkan nilai waktu tunda rata-rata antar paket. Untuk mendapatkan nilai *delay*, terlebih dahulu tentukan banyaknya paket (jumlah paket) yang terjadi selang masa pengambilan data (waktu *video chatting*). Untuk menghitung *delay Provider A* pada pukul 00.00 – 06.00 secara manual dapat dilakukan sesuai persamaan 2.1 .

$$Delay = \frac{900.008397}{70638,3} = 0.01274 \text{ s}$$

Berdasarkan perhitungan *delay* secara matematis seperti pada *delay Provider A* pada pukul 00.00 – 06.00, maka perhitungan *delay* untuk *Provider A*, *Provider B* dan *Provider C* pada rentang waktu yang lainnya dapat dihitung dengan persamaan yang sama. Berikut merupakan hasil perhitungan *delay* yang ditunjukkan tabel 4.1.

Tabel 4.1. Hasil Perhitungan *Delay*

Waktu	<i>Provider</i>	Waktu <i>Video chatting</i> (s)	Jumlah Paket	<i>Delay</i> (s)
Dini hari (00.00) - 06.00)	<i>Provider A</i>	900,008397	70638,3	0,01274
	<i>Provider B</i>	900,011248	68190,3	0,0132
	<i>Provider C</i>	900,014125	76345	0,01179

Waktu	Provider	Waktu Video chatting (s)	Jumlah Paket	Delay (s)
Pagi (06.00 - 12.00)	Provider A	900,012398	68695	0,0131
	Provider B	900,013707	54961	0,01638
	Provider C	900,017889	66277,3	0,01358
Siang (12.00 - 18.00)	Provider A	900,012967	62884,3	0,01431
	Provider B	57,2844077	2351,33	0,02436
	Provider C	395,06758	27519,7	0,01436
Malam (18.00 - 00.00)	Provider A	707,682589	51964,7	0,01362
	Provider B	130,380141	7935,33	0,01643
	Provider C	120,191216	6116,33	0,01965

Pada tabel 4.1 dapat dilihat nilai *delay* terbaik (terkecil) untuk masing-masing *provider* diperoleh pada waktu dinihari (00.00-06.00) yang mana pada waktu tersebut aktifitas telekomunikasi belum padat (bukan jam sibuk). Nilai *delay* terburuk (terbesar) untuk *provider A* dan *provider B* diperoleh pada waktu siang (12.00-18.00) sedangkan *provider C* mendapatkan nilai terburuk pada waktu malam (18.00-00.00) yang mana pada waktu siang merupakan pertengahan jam sibuk dan waktu malam merupakan akhir dari jam sibuk.

#### 4.2. Jitter

Perhitungan *jitter* dilakukan untuk melihat nilai selang waktu tunda rata-rata. Untuk mendapatkan nilai *jitter*, dibutuhkan jumlah paket dan nilai total variasi *delay*. Untuk menghitung nilai *jitter* *Provider A* pada pukul 00.00 – 06.00 secara manual dapat dilakukan sesuai dengan persamaan 2.2 .

$$Jitter = \frac{1006,9135}{70638,3 - 1} = 0,0142547 s$$

Berdasarkan perhitungan *jitter* secara matematis seperti pada *jitter* *Provider A* pada pukul 00.00 – 06.00, maka perhitungan *jitter* untuk *Provider A*, *Provider B* dan *Provider C*

pada rentang waktu yang lainnya dapat dihitung dengan persamaan yang sama. Berikut merupakan hasil perhitungan *jitter* yang ditunjukkan tabel 4.2.

Tabel 4.2. Hasil Perhitungan *Jitter*

Waktu	Provider	Total Variasi Delay (s)	Jumlah Paket	Jitter (s)
Dini hari (00.00 - 06.00)	Provider A	1006,9135	70638,3	0,0142547
	Provider B	987,0942	68190,3	0,0144758
	Provider C	1040,3676	76345	0,0136274
Pagi (06.00 - 12.00)	Provider A	1007,267	68695	0,0146631
	Provider B	893,3499	54961	0,0162545
	Provider C	1010,2594	66277,3	0,0152431
Siang (12.00 - 18.00)	Provider A	922,2575	62884,3	0,0146662
	Provider B	53,53157	2351,33	0,0227762
	Provider C	430,5386	27519,7	0,0156453
Malam (18.00 - 00.00)	Provider A	787,4505	51964,7	0,0151539
	Provider B	140,2314	7935,33	0,017674
	Provider C	163,1272	6116,33	0,0266751

Pada tabel 4.2 dapat dilihat nilai *jitter* terbaik (terkecil) untuk masing-masing *provider* diperoleh pada waktu dinihari (00.00-06.00) yang mana pada waktu tersebut aktifitas telekomunikasi belum padat (bukan jam sibuk). Nilai *jitter* terburuk (terbesar) untuk *provider A* dan *provider C* diperoleh pada waktu malam (18.00-00.00) sedangkan *provider B* mendapatkan nilai terburuk pada waktu siang (12.00-18.00) yang mana pada waktu siang merupakan pertengahan jam sibuk dan waktu malam merupakan akhir dari jam sibuk.

#### 4.3. Packet loss

Perhitungan *packet loss* dilakukan untuk mengetahui persentasi jumlah paket yang hilang selama terjadinya aktivitas *video chatting*. *software* wireshark menghitung nilai

*packet loss* secara langsung, *packet loss* yang diambil hanya berasal dari *software* tanpa melewati perhitungan ulang. Berikut merupakan tabel pengukuran wireshark terhadap *packet loss*.

Tabel 4.3. Tabel Pengukuran *Packet loss*

Waktu	Provider	Jumlah Paket	Packet loss (%)
Dini Hari (00.00 - 06.00)	Provider A	70638,33	0
	Provider B	68190,33	0
	Provider C	76345	0
Pagi (06.00 - 12.00)	Provider A	68695	0
	Provider B	54961	0
	Provider C	66277,33	0
Siang (12.00 - 18.00)	Provider A	62884,33	0
	Provider B	2351,333	0
	Provider C	27519,67	0
Malam (18.00 - 00.00)	Provider A	51964,67	0
	Provider B	7935,333	0
	Provider C	6116,333	0

Pada tabel 4.3 dapat dilihat nilai *packet loss* dari masing-masing *provider* sebesar 0%, sehingga nilai *packet loss* untuk masing-masing *provider* dapat dikategorikan baik.

#### 4.4. Throughput

Perhitungan *throughput* dilakukan untuk melihat seberapa efektifnya nilai transfer data yang terjadi setiap detik. Untuk mendapatkan nilai *throughput* dibutuhkan total panjang paket dan lamanya waktu *video chatting*. Untuk menghitung nilai *throughput* *Provider A* pada pukul 00.00 – 06.00 secara manual dapat dilakukan sesuai persamaan 2.5:

$$\text{Throughput} = \frac{45044955}{900,008397} = 50049,483 \text{ Byte/s}$$

Berdasarkan perhitungan *throughput* secara matematis seperti pada *throughput* *Provider A* pada pukul 00.00 – 06.00, maka perhitungan *throughput* untuk *Provider A*, *Provider B* dan *Provider C* pada rentang waktu yang lainnya dapat dihitung dengan persamaan yang sama. Berikut merupakan hasil perhitungan *delay* yang ditunjukkan tabel 4.4.

Tabel 4.4. Hasil Perhitungan *Throughput*

Waktu	Provider	Waktu Video chatting (s)	Panjang Data (Byte)	Throughput (Byte/s)
Dini Hari (00.00 - 06.00)	Provider A	900,008397	45044955	50049,483
	Provider B	900,011248	42758151	47508,4632
	Provider C	900,014125	52039234	57820,4633
Pagi (06.00 - 12.00)	Provider A	900,012398	42134047	46814,9632
	Provider B	900,013707	28098859	31220,479
	Provider C	900,017889	39816703	44239,9018
Siang (12.00 - 18.00)	Provider A	900,012967	36047003	40051,6488
	Provider B	57,2844077	1317073,7	22991,8354
	Provider C	395,06758	15631519	39566,6964
Malam (18.00 - 00.00)	Provider A	707,682589	30942842	43724,1815
	Provider B	130,380141	4689071,7	35964,6157
	Provider C	120,191216	3939681,3	32778,4464

Pada tabel 4.4 dapat dilihat nilai *throughput* terbaik (terbesar) untuk masing-masing *provider* diperoleh pada waktu dinihari (00.00-06.00) yang mana pada waktu tersebut aktifitas telekomunikasi belum padat (bukan jam sibuk). Nilai *throughput* terburuk (terkecil) untuk *provider A* dan *provider B* diperoleh pada waktu siang (12.00-18.00) sedangkan *provider C* mendapatkan nilai terburuk pada waktu malam (18.00-00.00) yang mana pada waktu siang merupakan pertengahan jam sibuk dan waktu malam merupakan akhir dari jam sibuk.

## V. Kesimpulan dan Saran

### 5.1. Kesimpulan

Adapun kesimpulan yang dapat diambil dari hasil penelitian ini adalah sebagai berikut :

1. Secara keseluruhan hasil pengukuran kinerja terbaik untuk masing-masing *provider* sebagai berikut:
  - a. *Provider A* mendapatkan hasil pengukuran terbaik pada saat dinihari dengan *delay* 0,01274 detik, *jitter* 0,01425 detik, *packet loss* 0 % dan *throughput* 50049,5 Bytes/s.
  - b. *Provider B* mendapatkan hasil pengukuran terbaik pada saat dinihari dengan *delay* 0,0132 detik, *jitter* 0,01447 detik, *packet loss* 0 % dan *throughput* 47508,5 Bytes/s.
  - c. *Provider C* mendapatkan hasil pengukuran terbaik pada saat dinihari dengan *delay* 0,01179 detik, *jitter* 0,01363 detik, *packet loss* 0 % dan *throughput* 57820,5 Bytes/s.
2. Secara keseluruhan hasil pengukuran kinerja terburuk untuk masing-masing *provider* sebagai berikut:
  - a. *Provider A* mendapatkan hasil pengukuran terburuk pada saat siang dengan *delay* 0,01431 detik, *jitter* 0,01467 detik, *packet loss* 0 % dan *throughput* 40051,6 Bytes/s.
  - b. *Provider B* mendapatkan hasil pengukuran terburuk pada saat siang dengan *delay* 0,02436 detik, *jitter* 0,02277 detik, *packet loss* 0 % dan *throughput* 22991,8 Bytes/s.
  - c. *Provider C* mendapatkan hasil pengukuran terburuk pada saat malam dengan *delay* 0,01965 detik, *jitter* 0,02667 detik, *packet loss* 0 % dan *throughput* 32778,4 Bytes/s.
3. Faktor yang mempengaruhi kinerja jaringan *provider* seluler adalah aktivitas penggunaan jaringan seluler pada waktu-

waktu tertentu (jam sibuk dan jam tidak sibuk).

### 5.2. Saran

1. Skripsi ini dapat dilanjutkan dengan menggunakan skenario pengukuran di beberapa lokasi berbeda ataupun menggunakan aplikasi *video chatting* yang berbeda.
2. Jam tidak sibuk adalah waktu terbaik untuk melakukan aktivitas telekomunikasi dengan beban trafik yang besar.

## DAFTAR PUSTAKA

- Cahyadi Seto A, Santoso Imam, Dan Zahra Ajub A. 2013 . “*Analisis Quality Of Service (QoS) Pada Jaringan Local Session Initial Protocol (SIP) Menggunakan GNS3*”. Universitas Diponegoro, Semarang.
- Inderjit Kaur, Harkarandeep Kaur, Dan Er. Gurjot Singh. 2014 . “*Analysing Various Packet Sniffing Tools*”. International Journal Of Electrical Electronics And Computer Science Engineering, India.
- Peraturan Menteri Komunikasi dan Informatika Republik Indonesia No. 16 Tahun 2013 Tentang Kualitas Layanan Jasa Teleponi Dasar Pada Jaringan Bergerak Seluler.
- Pratiwi Putri E, Isnawati Anggun F, Dan Hikmaturokhman Alfin. 2014 . “*Analisis Qos Pada Jaringan Multi Protocol Label Switching(MPLS) Studi Kasus Pelabuhan Indonesia III Cabang Tanjung Intan Cilacap*”. Akademi Telekomunikasi Sandhi Putra , Purwokerto.
- Samudra Rudy P. 2014 . “*Analisa Perbandingan QoS (Quality of Service) VoIP (Voice over Internet Protocol) Pada Jaringan OSPF (Open Shortest Path First) dan RIP (Routing Information Protocol)*”. Universitas Dian Nuswantoro, Semarang.
- Suhendra Made . 2010 . “*Analisis Performansi Live Streaming Dengan Menggunakan*

- Jaringan HSDPA*". Institut Teknologi Sepuluh November, Surabaya.
- Sutanto Tarcisius W C . 2011. "*Pembangunan Aplikasi Text Chatting Dan Video Chatting Berbasis Web*". Universitas Atma Jaya, Yogyakarta.
- Syafitri Dwita A. 2007 . "*Analisis Waktu Tunda Satu Arah Pada Panggilan VoIP Antara Jaringan UMTS Dan PSTN*". Universitas Sumatera Utara, Medan.
- Yuvandra Rayhan Dan Zulfin M. 2013 . "*Analisis Kinerja Trafik Video Chatting Client-Client Dengan Aplikasi Wireshark*". Universitas Sumatera Utara, Medan.