

Analisis Unjuk Kerja Aplikasi VoIP Call Android di Jaringan MANET (Mobile Ad Hoc Network)

Analysis Performance VoIP Call Application Android in MANET (Mobile Ad Hoc Network)

Ryan Ari Setyawan¹

¹Jurusan Teknik Informatika, Fakultas Teknik, Universitas Janabadra

¹Jl. Tentara Rakyat Mataram No. 55-57 Yogyakarta 55231, Indonesia

INFORMASI ARTIKEL

Naskah diterima 15 Juni 2015

Direvisi 24 Juni 2015

Disetujui 25 Juni 2015

Keywords:

Android

VoIP

MANET

Kata kunci :

Android

VoIP

MANET

ABSTRACT

This study aims to analyze the performance of VOIP call android application in the MANET (mobile ad hoc network). The results showed that VoIP applications could be implemented in MANET network. The highest delay is produced in indoor testing with distance of 11-15 meters, which is equal to 0.014624811 seconds. Packet loss is generated in the range of 1% -2%, while packet loss standards set by Cisco for VoIP application services are <5%. The jitter is between 0.01 to 0.06 seconds, while the standard set by CISCO is ≤ 30 ms or 0.03 seconds. Throughput generated in the testing process is between 161 kbps-481 kbps.

ABSTRAK

Penelitian ini bertujuan menganalisis kinerja aplikasi VoIP call android di jaringan MANET (mobile ad hoc network). Hasil pengujian menunjukkan bahwa aplikasi VoIP call android dapat digunakan di jaringan MANET. Delay yang dihasilkan paling besar di pengujian indoor dengan jarak 11-15 meter yakni sebesar 0,014624811 seconds. Packet loss yang dihasilkan pada range 1%-2%, sedangkan standar packet loss yang ditetapkan oleh CISCO untuk layanan aplikasi VoIP adalah < 5%. Jitter yang dihasilkan yakni antara 0,01-0,06 seconds sedangkan standar yang ditetapkan oleh CISCO adalah ≤ 30 ms atau 0,03 seconds. Throughput yang dihasilkan pada proses pengujian yakni antar 161 kbps-481 kbps.

1. Pendahuluan

Teknologi *smartphone* saat ini berkembang pesat, memudahkan pengguna dalam melakukan komunikasi, baik secara *real-time* ataupun tidak. Seiring dengan hal tersebut fitur aplikasi *smartphone* yang memudahkan pengguna dalam melakukan komunikasi bermunculan mulai dari *Mobile Yahoo Messenger*, *Kakao Talk*, *We Chat*, *Line*, *Bee Talk* dan yang lainnya. Fitur aplikasi-aplikasi tersebut sangat mudah. Hanya dengan koneksi jaringan internet, para pengguna *smartphone* dapat melakukan komunikasi percakapan yang tanpa batas wilayah, dan dengan biaya yang murah. Sesungguhnya, konsep dari fitur aplikasi tersebut adalah memanfaatkan teknologi jaringan *Voice over Internet Protocol* (VoIP) sebagai akses *internet multimedia system* (IMS). Namun hal tersebut tidak terlepas dari konektivitas yang digunakan. Konektivitas *wireless* merupakan salah satu fitur konektivitas untuk melakukan komunikasi pada *smartphone*. Hal tersebut menunjukkan bahwa pemampatan *wireless adapter* telah ada pada *device* saat ini. Keberadaan pemampatan *wireless adapter* tersebut dapat mendorong dalam membangun jaringan komunikasi lingkup kecil seperti *Mobile Ad Hoc Network* (MANET). MANET adalah jaringan *wireless* yang terdiri dari kumpulan *mobile node* (*mobile station*) yang bersifat dinamik dan spontan, dapat diaplikasikan dimanapun tanpa menggunakan jaringan infrastruktur yang telah ada. Pada jaringan *ad hoc* setiap *node* tidak hanya berfungsi sebagai pengirim dan penerima informasi tetapi juga berfungsi sebagai pendukung jaringan seperti *router*.

¹ Email : ryan@janabadra.ac.id

Karakteristik *node* dalam MANET dapat bebas datang dan meninggalkan jaringan, *node* bebas bergerak atau diam pada posisinya. Setiap *mobile node* memiliki *wireless network interface* dan saling berkomunikasi dengan memanfaatkan media transmisi. Karena daya pancar *node* yang terbatas maka komunikasi antar *node* dapat dilakukan dengan melewati satu dari beberapa *node*.

Karakteristik MANET yang memiliki topologi dinamis dapat bebas bergerak kemana saja dan kapan saja mengakibatkan topologi jaringan berubah secara acak dan cepat pada waktu yang tidak diprediksikan. Batasan *bandwidth* dan kapasitas yang dimiliki tiap jalur berbeda-beda. *Mobile node* menggunakan baterai sebagai sumberdayanya. Daya baterai yang terbatas mengharuskan setiap operasi harus berjalan secara efisien.

Berdasarkan uraian di atas maka perangkat-perangkat seperti *smartphone* dan laptop dapat dijadikan sebagai *node-node* yang dapat saling berkomunikasi satu sama lain untuk membentuk MANET melalui jaringan nirkabel. Namun hal tersebut menimbulkan pertanyaan sebagai rumusan masalah yang dapat diangkat yakni bagaimana pengaruh komunikasi yang terjadi pada aplikasi telepon internet VoIP *call* di MANET.

Node yang digunakan dalam penelitian ini yakni menggunakan dua buah *smartphone* dengan sistem operasi android dan satu unit laptop. Sedangkan aplikasi VoIP *call* yang digunakan yakni *CssipSimple* yang nantinya dianalisis menggunakan *software wireshark* untuk mengambil data yang terjadi pada saat komunikasi. Metode yang digunakan dalam penelitian ini adalah untuk mengukur kualitas layanan suara (*Quality of Service*) yang dihasilkan yakni *throughput*, *delay*, *jitter* dan *packet loss*. Tujuannya menggunakan metode QoS adalah untuk mengetahui pengaruh komunikasi yang terjadi pada VoIP *call* di jaringan *mobile ad hoc network* dan QoS dapat dijadikan sebagai ukuran untuk menentukan baik atau buruknya kinerja suatu jaringan internet..

2. Tinjauan Pustaka

Berbagai studi mengenai jaringan VoIP di MANET yang telah dilakukan seperti desain serta implementasi VoIP *ad hoc* dengan menanamkan p-SIP (*Session Initiation Protocol*) server (Chang, Sung, Chiu, & Lin, 2010). Penelitian tersebut bertujuan merancang server p-sip yang tertanam pada jaringan VoIP *ad hoc*. Metode yang dilakukan untuk mengukur kinerja VoIP digunakan metode QoS, namun karena yang dititikberatkan adalah server p-SIP maka pengukuran kinerja lebih kepada *REGISTER host* dan *INVITE* pada SIP.

Studi berikutnya yakni mengenai paket *delay* VoIP dalam *single hop* di jaringan Ad Hoc IEEE 802.11 (Barcel, Bellalta, Cano, & Sfairopoulou, 2008). Studi tersebut memperkenalkan prediksi model VoIP pada IEEE 802.11 untuk jaringan *ad hoc*. Model tersebut digunakan untuk mengetahui dampak atau prediksi *delay* yang terjadi pada VoIP melalui parameter konfigurasi seperti *codec*, *packetization interval* dan *data rate*.

Studi mengenai komunikasi VoIP di jaringan *wireless ad hoc* melalui *gateway* dilakukan oleh Faloso et.al (Faloso et al., 2007). Studi tersebut memperkenalkan rumus untuk memperkirakan jumlah sesi suara berkelanjutan dalam satu *hop cell*. Keakuratan rumus tersebut dapat diimplementasikan melalui perangkat lunak simulasi jaringan NS2.

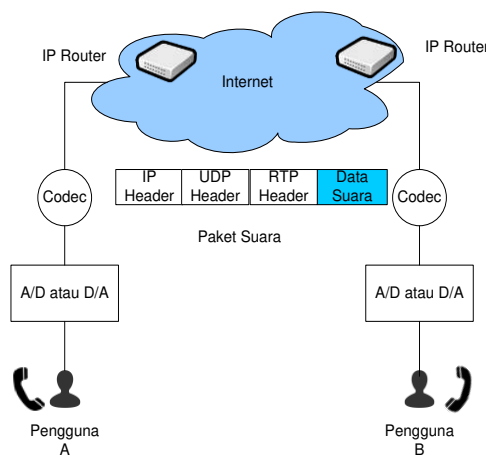
Studi selanjutnya adalah mengenai analisis VoIP melalui jaringan VANET (*vehicular ad hoc networks*). Metode yang digunakan untuk melakukan analisis VoIP pada jaringan VANET menggunakan metode *Mean Opinion Score (MOS)* yakni untuk mengukur kualitas suara yang diterjadi saat komunikasi melalui jaringan VANET (El Brak, Bouhorma, & Boudhir, 2012).

Beberapa studi mengenai jaringan VoIP di MANET telah diuraikan di atas, pada penelitian kali ini penulis ingin melakukan penelitian terhadap komunikasi yang terjadi pada aplikasi VoIP *call* android di *Mobile Ad hoc Networks*. Mengingat fitur aplikasi di android sudah terdapat fitur VoIP *Call*. Hanya dengan mengkoneksikan internet pengguna dapat menggunakan layanan telepon internet tersebut. Metode yang digunakan dalam penelitian ini yakni dengan metode pengukuran kualitas layanan suara. Parameter QoS seperti *delay*, *jitter*, *throughput*, dan *packet loss* dapat dijadikan sebagai ukuran untuk mengetahui kualitas

layanan suara yang dihasilkan saat komunikasi dilakukan. Pengukuran QoS dilakukan nantinya saat komunikasi berlangsung dengan melalui perangkat lunak wireshark, data suara diambil kemudian dianalisis dan dihitung melalui parameter *delay*, *jitter*, *throughput* dan *packet loss*.

2.1 Voice over Internet Protocol

Layanan *Voice over internet Protocol* (VoIP) merupakan teknologi yang merubah data sinyal suara analog menjadi data digital, kinerja VoIP dengan menerapkan *codec* suara untuk kompres suara paket data dan mentransfer data suara terkompresi melalui *internet protocol* (IP). Komunikasi telepon melalui VoIP menjadi lebih murah karena menggunakan frekuensi (*bandwith*) dengan sistem terkompresi yang tingkatnya lebih besar dibanding kompresi seluler. Di *Global System for Mobile Communication* (GSM) suara normal dikodekan dalam 64 kilobit dapat ditekan sampai 13,3 kilobit per detik dengan kualitas suara lebih baik. Namun di dalam VoIP kompresi suara dilakukan dari kanal 13,3 kilobit menjadi 8 kilobit dan nantinya akan lebih kecil lagi (Setyawan, Sulisty, Hantono, Grafika, & Yogyakarta, 2014). Skema dasar kinerja VoIP dapat diperlihatkan pada Gambar 1.

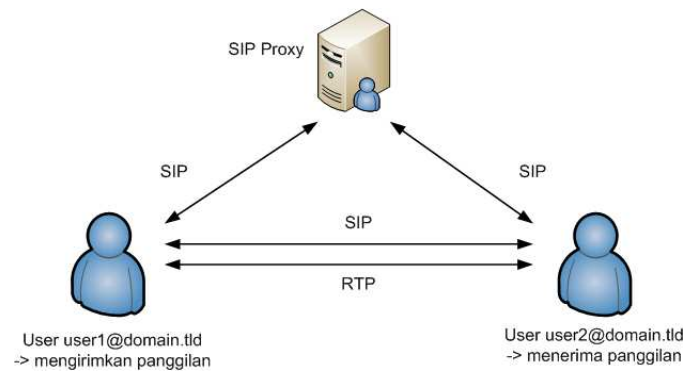


Gambar 1. Skema Kinerja VoIP (Setyawan et al., 2014)

Gambar 1 menjelaskan bahwa pengguna A dan pengguna B dapat melakukan komunikasi melalui layanan telepon internet, dimana data suara akan diproses melalui *analog to digital converter* atau sebaliknya dan dipecah berupa paket-paket data suara yang dikirimkan ke alamat yang dituju yakni pengguna B. Kompresi dilakukan di STDI (Sentral Telepon Digital Indonesia), sehingga suatu kanal kapasitas 64 kilobit dapat digunakan oleh 4-5 suara. Dengan teknologi VoIP kompresi dilakukan sedemikian rupa sehingga tiap kanal tidak lagi 13,3 kilobit tetapi menjadi 8 kilobit.

2.2 Session Initiation Protocol

Salah satu protokol yang digunakan dalam penelitian ini adalah *session initiation protocol* (SIP). Protokol SIP merupakan protokol standar multimedia, produk dari *Internet Engineering Task Force* (IETF) dan telah digunakan menjadi suatu standar penggunaan VoIP. SIP merupakan protokol yang berada pada layer aplikasi untuk mendefinisikan proses awal, pengubahan, dan pengakhiran (pemutusan) suatu sesi komunikasi multimedia. SIP juga dapat dikatakan memiliki karakteristik *client-server*, artinya *request* dilakukan oleh *client* dan ditujukan ke *server*. *Server* mengolah *request* dan memberikan tanggapan terhadap *request* yang dilakukan oleh *client*. *Request* dan tanggapan terhadap *request* tersebut disebut transaksi SIP.



Gambar 2. Skema Kinerja SIP (Setyawan et al., 2014)

Gambar 2. Menjelaskan bahwa SIP bekerja ketika User1@domain.tld memasukkan *username* dan *password* yang sudah didapatkan dari *server* SIP atau *SIP proxy* dan telah terotentikasi. User1@domain.tld dapat melakukan panggilan ke User2@domain.tld, kemudian komunikasi terjadi pada *Real-time-transport protocol* (RTP). Gambar 2 juga memperlihatkan bahwa SIP merupakan suatu protokol *signalling* pada *layer* aplikasi yang berfungsi membangun, memodifikasi, dan mengakhiri suatu sesi multimedia yang melibatkan satu atau beberapa pengguna. Sesi multimedia adalah pertukaran data antar pengguna yang meliputi suara, video atau teks.

SIP tidak menyediakan layanan secara langsung, tetapi menyediakan fondasi yang dapat digunakan oleh protokol aplikasi lainnya untuk memberikan layanan yang lebih lengkap bagi pengguna, misalnya dengan RTP (*Real Time Transport Protocol*) untuk transfer data secara *real-time* dan SDP (*session description protocol*) untuk mendeskripsikan sesi multimedia (Fahdi, Patih, Fitriawan, & Yuniati, 2012). Pembangunan suatu komunikasi multimedia dengan SIP dilakukan melalui beberapa tahap :

- *User location*, menentukan lokasi pengguna yang akan berkomunikasi.
- *User availability*, menentukan tingkat keinginan pihak yang dipanggil untuk terlibat dalam komunikasi.
- *User capability*, menentukan media maupun parameter yang berhubungan dengan media yang akan digunakan untuk komunikasi.
- *Session setup*, 'ringing' pembentukan hubungan antara pihak pemanggil dan pihak yang dipanggil.
- *Session management*, meliputi transfer, modifikasi dan pemutusan sesi.

2.3 Real Time Transport Protocol (RTP)

Real time transport protocol (RTP) merupakan protokol yang mendefinisikan standard format data pengiriman suara dan gambar. RTP dikembangkan pertama kali oleh *audio video transport working group* IETF dan pertama kali dipublikasikan pada tahun 1996 sebagai RFC 1889. RTP tidak memiliki port TCP dan UDP yang standar. Walaupun tidak didefinisikan secara standar, RTP dikonfigurasi menggunakan *range port* 16384-32767.

Pada awalnya RTP didesain untuk protokol yang bersifat *multicast*, namun sekarang ini banyak digunakan di dalam aplikasi yang bersifat *unicast*. RTP digunakan untuk sistem *streaming* media seperti *video conferencing* dan *push to talk system*. Di dalam teknologi VoIP, RTP berfungsi untuk mengatur komunikasi *real time* antara dua pengguna VoIP. RTP menyediakan layanan sebagai berikut :

- a. *Payload type identification*
- b. *Sequence numbering*
- c. *Time stamping*
- d. *Delivery monitoring*

RTP tidak menyediakan mekanisme untuk menentukan lamanya sebuah pengiriman data. Selain itu RTP juga tidak menjamin *Quality of Service* (QoS) dari sebuah pengiriman data. Penanganan terhadap realibilitas dikerjakan oleh mekanisme pada layer yang lebih di atas pada model OSI.

2.4 Mobile Ad Hoc Network (MANET)

Mobile Ad Hoc Networks adalah jaringan nirkabel yang terdiri dari kumpulan *mobile node* yang bersifat dinamik dan spontan, dapat diaplikasikan di manapun tanpa menggunakan jaringan infrastruktur (seluler ataupun PSTN) yang telah ada. Contoh *mobile node* adalah notebook, PDA dan *smartphone*. *Node* bebas datang dan meninggalkan jaringan, *node* juga bebas bergerak atau diam pada posisinya. Setiap *mobile node* memiliki *wireless network interface* dan saling berkomunikasi dengan memanfaatkan media transmisi. Karena media transmisi mempunyai daya pancar yang terbatas, maka komunikasi antar *node* tersebut dilakukan dengan melewati satu dari beberapa *node* (*node* berfungsi sebagai *router* dan *host*) sehingga MANET juga disebut *multihop network*. Berikut ini beberapa karakteristik keuntungan dan kelemahan dari MANET :

- Tidak memerlukan dukungan infrastruktur *backbone* sehingga mudah diimplementasikan dan sangat berguna ketika infrastruktur tidak ada ataupun tidak berfungsi lagi.
- Topologi yang dinamis *node-node* di MANET bebas bergerak kemana saja dan kapan saja. Hal ini mengakibatkan topologi jaringan berubah secara acak dan cepat pada waktu yang tidak diprediksi.
- *Mobile node* yang selalu bergerak (*mobility*) dapat mengakses informasi secara *real time* ketika berhubungan dengan *mobile node* lain, sehingga pertukaran data dan pengambilan keputusan dapat segera dilaksanakan.
- Fleksibel terhadap suatu keperluan tertentu karena jaringan ini memang bersifat sementara.
- Dapat direkonfigurasi dalam beragam topologi, baik untuk jumlah *user* kecil hingga banyak sesuai dengan aplikasi dan instalasi (*scalability*).
- Seringkali terjadi *disconnection*, karena tidak selalu berada dalam area cakupan.
- *Bandwidth* komunikasi yang terbatas.
- *Lifetime* baterai yang singkat.
- Kapasitas *mobile node* yang terbatas dan bervariasi.

2.5 Topologi Mesh MANET

Topologi *Mesh* MANET adalah salah satu jenis jaringan dimana setiap *node* di jaringan tidak hanya menerima atau mengirim data miliknya, tapi juga berfungsi sebagai *relay* untuk *node* yang lain. Dengan kata lain, setiap *node* bekerjasama untuk membangun dan mengirimkan data di jaringan. Manfaatnya antara lain:

- Dapat membangun jaringan yang *reliable*.
- Biaya instalasi sangat murah.
- Adaptif terhadap perubahan kondisi / topologi jaringan.
- Cocok untuk keperluan Militer, wilayah Rural/Perkampungan dan Pegunungan.

Sebuah jaringan *mesh* dapat di rancang menggunakan teknik *flooding* atau menggunakan teknik *routing*. Jika menggunakan teknik *routing*, maka *message* akan di kirim melalui sebuah jalur, dengan cara "loncat" dari satu *node* ke *node* yang lain sampai tujuan tercapai. Untuk menjamin keberadaan *route / path*, maka sebuah mekanisme *routing* harus memungkinkan untuk terjadi sambungan terus menerus dan mengkonfigurasi secara otomatis jika ada jalur yang rusak atau terblokir, menggunakan algoritma "*self-healing*" atau "memperbaiki diri sendiri". Sebuah jaringan *mesh* terjadi dimana semua *node* tersambung ke satu sama lain sebagai sebuah jaringan yang saling tersambung.

Jaringan *mesh* dapat di lihat sebagai salah satu jenis jaringan *ad-hoc*. *Mobile ad hoc network* (MANET) dan jaringan *mesh* sangat erat satu sama lain, akan tetapi MANET harus dapat menyelesaikan masalah yang terjadi karena mobilitas *node*.

Kemampuan *self-healing* memungkinkan sebuah jaringan yang berbasis *routing* untuk tetap bekerja jika salah satu *node* rusak atau sambungan menjadi jelek. Akibatnya, jaringan ini umumnya sangat *reliable*, biasanya ada lebih dari satu sambungan antara sumber dan tujuan di jaringan. Meskipun skenario ini

biasanya digunakan di *wireless*, konsep ini juga dapat digunakan di jaringan kabel maupun interaksi *software*.

2.6 Quality Of Service VoIP

Pembahasan selanjutnya adalah mengenai metode yang digunakan dalam penelitian ini untuk mengukur kualitas kinerja VoIP. Secara umum, ada beberapa parameter-parameter penting yang mempengaruhi *Quality of Service* (QoS) layanan suara pada jaringan VoIP. Parameter ini dijadikan gambaran ukuran kinerja dari suatu jaringan VoIP. Parameter QoS tersebut yakni *delay*, *jitter*, *packet loss* dan *throughput*.

a. Delay

Delay (latency), adalah waktu tunda yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. Dalam perancangan jaringan VoIP, waktu tunda merupakan suatu permasalahan yang harus diperhitungkan karena kualitas suara bagus tidaknya tergantung dari waktu tunda. Besarnya waktu tunda maksimum yang direkomendasikan oleh ITU-T G.711 untuk aplikasi suara adalah 160 ms, sedangkan waktu tunda maksimum dengan kualitas suara yang masih dapat diterima pengguna adalah 250 ms. Waktu tunda *end-to-end* adalah jumlah waktu tunda konversi suara analog ke digital, waktu tunda waktu paketisasi atau bisa disebut juga waktu tunda panjang paket dan waktu tunda jaringan pada saat t (waktu) tertentu. Berbagai sumber jenis penyebab *delay*. Jenis-jenis tersebut diperlihatkan pada Tabel 1.

Tabel 1. Jenis-Jenis Sumber *Delay*

<i>Delay</i>	Keterangan
<i>Processing delay</i>	<i>Delay</i> ini terjadi pada saat proses <i>coding</i> , <i>compression</i> , <i>decompression</i> , dan <i>decoding</i> . <i>Delay</i> ini tergantung pada standar <i>codec</i> yang digunakan.
<i>Packetization delay</i>	<i>Delay</i> yang disebabkan oleh peng-akumulasian bit <i>voice sample</i> ke <i>frame</i> . Seperti contohnya, standar G.711 untuk <i>payload</i> 160 bytes memakan waktu 20ms
<i>Serialization delay</i>	<i>Delay</i> ini terjadi karena adanya waktu yang dibutuhkan untuk pentransmisi paket IP dari sisi <i>originating</i> (pengirim).
<i>Propagation delay</i>	<i>Delay</i> ini terjadi karena perambatan atau perjalanan. Paket IP di media transmisi ke alamat tujuan. Seperti contohnya <i>delay</i> propagasi di dalam Tabel akan memakan waktu 4 sampai 6 μ s per kilometernya.
<i>Queueing delay</i>	<i>Delay</i> ini disebabkan karena waktu tunggu paket selama antrian sampai dilayani.
<i>Component delay</i>	<i>Delay</i> ini disebabkan oleh banyaknya komponen yang digunakan di dalam sistem transmisi.

Sumber : (Setiawan, 2012a)

Sedangkan untuk tingkat kualitas jaringan berdasarkan waktu tunda (*delay*) diperlihatkan pada Tabel 2.

Tabel 2. Kategori Tingkat Kualitas *Delay*

<i>Delay</i>	Kategori
0-150 ms	Dapat diterima untuk kebanyakan aplikasi pengguna
150-300	Masih dapat diterima jika pelaksana telah mengetahui akibat waktu transmisi pada QoS aplikasi pengguna
Lebih dari 300 ms	Tidak dapat diterima untuk perencanaan rancangan jaringan pada umumnya.

Sumber : (Setiawan, 2012a)

Untuk menghitung *delay* yang terjadi digunakan Persamaan 2 (Setiawan, 2012b)(Suryawan, Husni, & Anggraini, 2012).

$$Delay = \frac{Twf}{P} \quad (1)$$

Dimana :

Twf = Time Between first and last packet

P = Jumlah Packet

b. *Throughput*

Throughput yaitu kecepatan (*rate*) transfer data efektif, yang diukur dalam bps. *Header* dalam paket data mengurangi nilai ini. *Throughput* dapat dihitung dengan melihat jumlah paket yang datang terhadap yang dikirim. *Throughput* juga merupakan kemampuan sebenarnya suatu jaringan dalam melakukan pengiriman data. Biasanya *throughput* selalu dikaitkan dengan *bandwidth*. Karena *throughput* memang dapat disebut juga dengan *bandwidth* dalam kondisi yang sebenarnya. *Bandwidth* lebih bersifat pasti sementara *throughput* sifatnya adalah dinamis tergantung trafik yang sedang terjadi. *Throughput* juga diartikan sebagai jumlah *bit* yang diterima dengan sukses per detik melalui sebuah sistem atau media komunikasi dalam selang waktu pengamatan tertentu. Umumnya *throughput* direpresentasikan dalam satuan *bit per second* (bps). Untuk meng-capture nilai *throughput* dapat dilakukan dengan melihat nilai *bit per second* (bps). Untuk menghitung *throughput* yang terjadi digunakan Persamaan 2 (Setiawan, 2012)(Suryawan et al., 2012).

$$Throughput = \frac{JumlahData}{WaktuPengiriman} \quad (2)$$

Dimana :

Average Byte/sec = jumlah data

Time between first & last packet (sec) = waktu

c. *Packet Loss*

Packet loss adalah banyaknya paket yang hilang pada suatu jaringan paket yang disebabkan oleh *Collision*, penuhnya kapasitas jaringan, dan *packet drop* yang disebabkan oleh habisnya *time to live* paket. Persamaan untuk menghitung *packet loss* dapat diperlihatkan pada Persamaan 3.

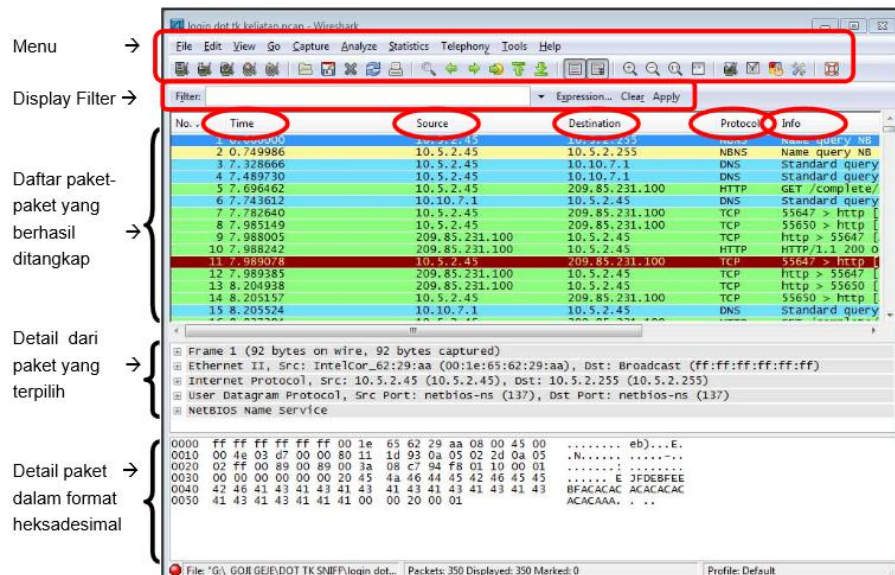
$$Packet Loss = \frac{PaketTotalTercapture - PaketTer kirim}{PaketTotalTercapture} \times 100\% \quad (3)$$

d. *Jitter*

Jitter adalah variasi dari *delay*. *Jitter* disebabkan oleh adanya variasi waktu dalam kedatangan paket. Variasi kedatangan paket ini dapat disebabkan oleh panjang antrian data, lamanya waktu pengolahan data dan lamanya waktu yang dibutuhkan untuk menghimpun paket-paket data yang akhir proses transmisi untuk menjadi satu kesatuan *frame* yang utuh.

2.7 Wireshark

Wireshark merupakan salah satu dari *software* monitoring jaringan yang biasanya banyak digunakan oleh para administrator jaringan untuk mengcapture dan menganalisa kinerja jaringan. Salah satu alasan kenapa *Wireshark* banyak dipilih oleh seorang administrator adalah karena *interface* nya menggunakan *Graphical User Unit* (GUI) atau tampilan grafis.



Gambar 5. Wireshark

Keterangan :

1. Menu : Navigasi antar menu-mnu yang tersedia di wireshark
2. Display filter: Kolom untuk menuliskan sintaks-sintaks untuk memfilter (membatasi) paket-paket apa saja yang bakalan ditampilkan pada list paket.
3. Daftar paket : Tampilan paket-paket yang berhasil ditangkap oleh wirehsark, berutuan mulai dari paket pertama yang ditangkap dan seterusnya.
4. Detail paket : sebuah paket yang tentunya membawa informasi tertentu yang berbeda-beda antar paketnya.
5. Detail heksa : detail paket yang terpilih ditampilkan dalam bentuk heksa terkadang akan lebih mudah untuk mendapatkan informasi.

3. Metode Penelitian

Pada penelitian ini dilakukan pengukuran kualitas suara dengan menggunakan metode QoS. Data suara yang digunakan sebesar 1,830 detik. Jaringan yang digunakan untuk menghubungkan antara satu *node* (*device*) dengan *node* yang lain digunakan jaringan *wireless* LAN. Saat terjadi komunikasi antar satu *node* dengan *node* yang lain, data komunikasi yang terjadi di jaringan diambil menggunakan perangkat lunak wireshark. Kemudian hasil *capture* data jaringan oleh perangkat lunak wireshark tersebut dianalisis melalui pendekatan parameter *delay*, *jitter*, *throughput* dan *packet loss*. Pendekatan tersebut dilakukan untuk mengetahui kualitas layanan suara yang dihasilkan oleh jaringan VoIP di MANET.

3.1 Analisis Sistem

Analisis sistem adalah penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponennya, dengan maksud untuk mendapatkan perangkat lunak dan perangkat keras yang dibutuhkan dalam penelitian ini. Analisis tersebut dilakukan agar dalam proses penelitian ini dapat berjalan seperti yang direncanakan. Dalam penelitian ini dibutuhkan perangkat keras dan perangkat lunak sebagai berikut :

a. Analisis Perangkat Keras (*Hardware*)

Perangkat keras (*hardware*) adalah sistem utama dari sebuah sistem secara fisik, yang terdiri dari komponen-komponen yang saling terkait. Perangkat keras juga dapat dikatakan sebagai *device*. *Device* tersebut nantinya dalam penelitian ini digunakan sebagai *node*. *Device* tersebut adalah :

- 1) Satu unit laptop sebagai salah satu node untuk mengimplementasikan MANET dengan spesifikasi :
 - a) *Network* : WiFi 802.11b/g/n, Ethernet Port
 - b) *Support* WiFi adaptor / WiFi ad hoc

- c) Baterai 5 hours (4-cell battery).
- 2) Dua buah *smartphone* dengan spesifikasi minimum :
 - a) Sistem operasi : Android 4.3 (Jelly Bean)
 - b) *Network* : HSDPA 42.2 Mbps; HSUPA 5.76 Mbps
 - c) Konektivitas : WiFi 802.11/b/g/n
 - d) Support WiFi direct, WiFi *hotspot*/ WiFi ad hoc
 - e) Baterai : LiPo 1170 mAh
- b. Analisis Perangkat Lunak (*Software*)

Perangkat lunak yang dibutuhkan dalam penelitian ini untuk melakukan ujicoba aplikasi VoIP di dalam jaringan MANET adalah sebagai berikut :

 - 1) CsipSimple, merupakan aplikasi *mobile softphone* atau aplikasi telepon internet (VoIP) yang menggunakan *session initiation protocol* (SIP) untuk *device* android. CsipSimple dapat diperoleh dengan mudah di *google play* android.
 - 2) Wireshark, *software* untuk mengambil atau meng-*capture* paket data jaringan internet.
 - 3) Voiprakyat.or.id, server SIP gratis untuk mendapatkan akun SIP seperti *username* dan *password* SIP.

3.2 Pemodelan Sistem

Pemodelan sistem merupakan gambaran secara keseluruhan proses kinerja sistem atau alur jalannya sistem. Pemodelan sistem dalam penelitian ini dapat dilihat pada Gambar 7. Gambar 7 memperlihatkan bahwa hal yang pertama kali dilakukan adalah melakukan instalasi aplikasi VoIP call di *smartphone* android yakni aplikasi CsipSimple. Kemudian melakukan penentuan skenario pengujian dan perancangan jaringan MANET dengan menentukan *node-node* yang akan digunakan. Setelah itu *node-node* yang digunakan diimplementasikan ke dalam topologi *mesh*. Kemudian satu persatu *node* dihubungkan ke jaringan masing-masing. Jika sudah terhubung lakukan percakapan dan data dicapture menggunakan *software* wireshark. Kemudian dianalisis sesuai dengan parameter *delay*, *jitter*, *throughput* dan *packet loss*.

3.3 Perancangan Sistem

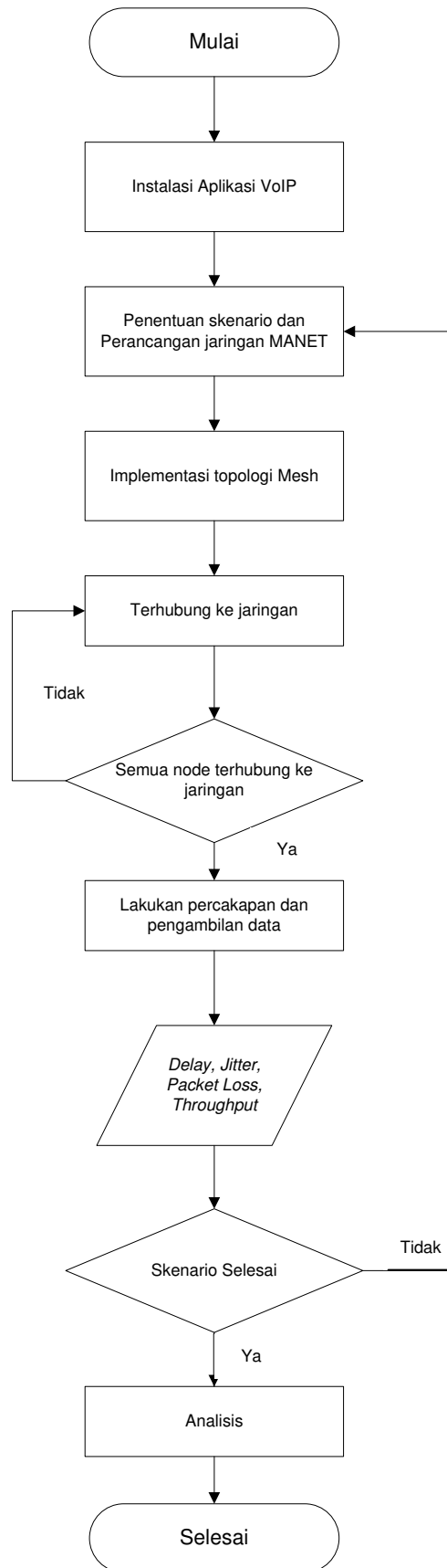
Perancangan sistem dalam penelitian ini menggunakan topologi *mesh* di jaringan MANET. Topologi *mesh* dalam penelitian ini diperlihatkan pada Gambar 8, yang memperlihatkan bahwa *smartphone* A, *smartphone* B dan laptop dijadikan sebagai *node*. Pada *node-node* tersebut terdapat garis putus-putus antara satu *node* dengan *node* yang lain. Garis putus-putus tersebut menunjukkan bahwa komunikasi yang terjadi adalah dapat bergerak bebas antara *node* satu dengan *node* yang lain.

3.4 Implementasi

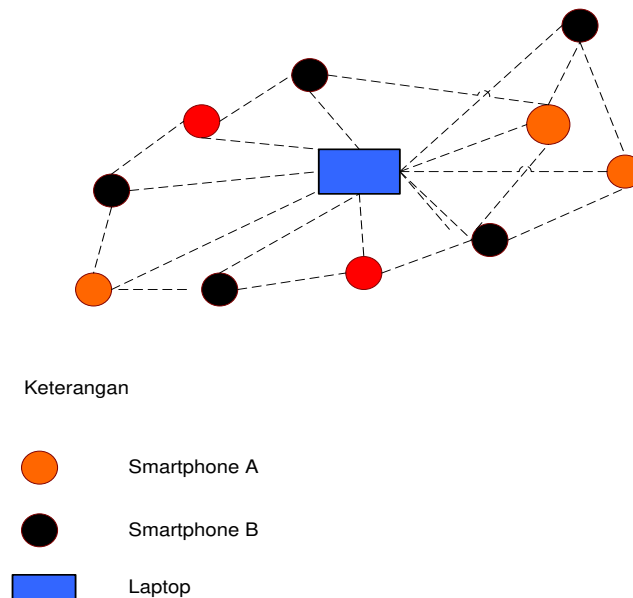
Proses implementasi dalam penelitian ini terbagi dalam beberapa implementasi yakni implementasi aplikasi, implementasi protokol SIP dan implementasi jaringan MANET.

a. Implementasi aplikasi

Implementasi aplikasi VoIP *Call* menggunakan aplikasi CsipSimple, yang sebelumnya telah di *install* di masing-masing *smartphone* android.

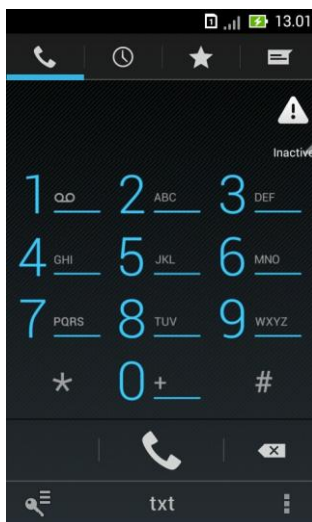


Gambar 7. Diagram Alir Sistem

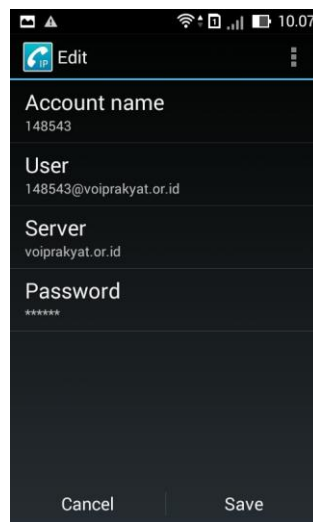


Gambar 8. Topologi Mesh

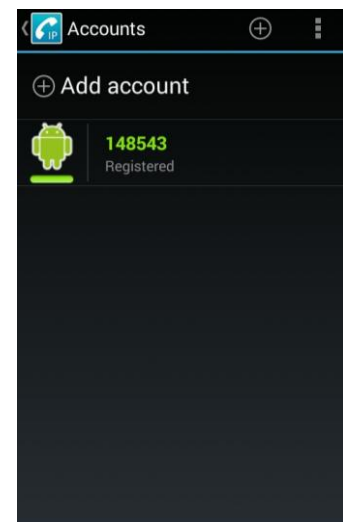
Gambar 9. merupakan gambar aplikasi CsipSimple yang merupakan salah satu aplikasi VoIP *Call* yang tersedia di google play android. Kemudian untuk menggunakan aplikasi tersebut pengguna wajib memasukkan *username* dan *password* yang telah didapatkan dari *server* SIP diperlihatkan pada Gambar 10. Gambar tersebut memperlihatkan pengaturan akun SIP pengguna diwajibkan memasukkan *account name*, *user*, *server name* dan *password*. Apabila sudah memasukkan akun SIP tersebut maka server akan mengotentikasi dan hasilnya seperti pada Gambar 11, yang memperlihatkan akun SIP telah registered artinya bahwa akun SIP yang telah dimasukan ke dalam pengaturan telah terotentikasi dan siap untuk digunakan.



Gambar 9. Aplikasi CsipSimple



Gambar 10. Pengaturan Akun SIP



Gambar 11. Otentikasi Akun SIP

b. Implementasi Protokol SIP

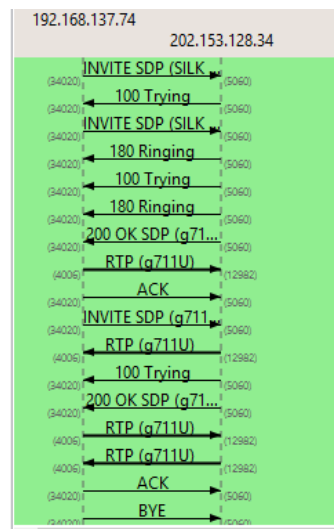
Setelah aplikasi CsipSimple telah terinstall maka proses berikutnya adalah implementasi protokol SIP merupakan proses implementasi penggunaan protokol SIP dalam aplikasi. Protokol SIP nantinya melakukan proses *signalling* sehingga aplikasi CsipSimple satu dengan yang lain dapat melakukan komunikasi.

Pada penelitian ini proses pembuatan akun SIP menggunakan *server* atau *proxy* voiprakyat.or.id, dari proses pembuatan akun SIP tersebut diperoleh *username* (VoIP id) dan *password*. Penelitian ini menggunakan dua akun agar dapat digunakan oleh dua *smartphone* untuk pengujian seperti diperlihatkan pada Tabel 3.

Tabel 3. Akun sip

Konfigurasi	Akun SIP Smartphone A	Akun SIP Smartphone B
<i>Account name</i>	147400	148543
<i>Password</i>	GYJAEP	226DXC
<i>Server</i>	voiprakyat.or.id	voiprakyat.or.id
<i>Port</i>	5060	5060

Pada Tabel 3 Akun SIP A dan Akun SIP B digunakan untuk melakukan konfigurasi di aplikasi. Setelah memasukan akun SIP tersebut maka dapat terotentikasi oleh server SIP. Proses otentikasi akun SIP *smartphone* A diperlihatkan pada Gambar 12. Gambar tersebut memperlihatkan bahwa akun SIP *smartphone* A dengan *account name* 147400@voiprakyat.or.id dengan IP : 192.168.137.74 yang telah terotentikasi (*invite*) di server voiprakyat.or.id dengan IP : 202.153.128.34. Untuk proses otentikasi akun SIP yang lain memiliki otentikasi yang sama.



Gambar 12. Otentikasi Protokol SIP.

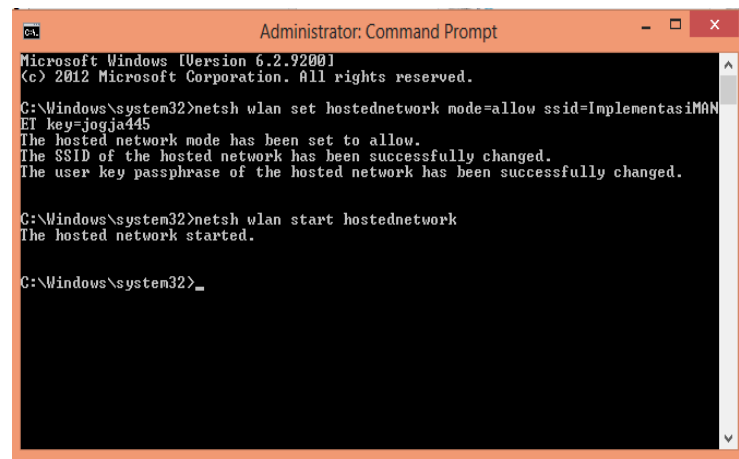
c. Implementasi MANET

Setelah aplikasi dan akun SIP telah dapat digunakan proses selanjutnya adalah implementasi pada jaringan MANET. Proses ini melakukan *wifi ad hoc* agar semua *node* dapat terhubung satu sama lain. Berikut implementasinya :

- 1) Membuat WiFi *ad hoc* di laptop yang nantinya sebagai *node* koordinator melalui *command prompt admin* (windows 8). Prosesnya diperlihatkan pada Gambar 13.

Gambar 13 memperlihatkan proses untuk membuat WiFi *ad hoc* (windows 8) dengan *command prompt admin* dengan cara menuliskan perintah :

```
C:\windows\system32>netsh wlan set hostednetwork mode=allow ssid=ImplementasiMANET key=jogja445
```



Gambar 13. WiFi Ad Hoc

- 2) Kemudian setelah sukses, untuk mengaktifkan wifi *ad hoc* tuliskan perintah :
C:\windows\system32>netsh wlan start hostednetwork
- 3) Secara otomatis maka *wifi ad hoc* dapat dipakai oleh *node-node* lain seperti diperlihatkan pada Gambar 14.



Gambar 14. Implementasi WiFi Ad Hoc

- 4) Selanjutnya koneksikan atau hubungkan semua *node-node* baik *smartphone* A dan *smartphone* B ke WiFi *ad hoc*.

Gambar 15 memperlihatkan bahwa masing-masing *smartphone* sebagai *node* terhubung ke dalam jaringan *ad hoc*.



Gambar 15. Koneksi node ke WiFi *ad hoc*

4. Hasil Penelitian dan Pembahasan

Setelah semua *node-node* telah terhubung semua, proses selanjutnya adalah melakukan pengujian. Proses pengujian dalam penelitian ini dengan melakukan pengujian didalam ruangan (*indoor*) dan luar ruangan (*outdoor*). Pengujian tersebut dilakukan untuk menguji karakteristik dari MANET yang memiliki daya pancar jaringan yang terbatas.

Namun sebelum melakukan pengujian terdapat beberapa skenario pengujian yang harus dilakukan. Tujuannya agar pengujian mendapatkan data yang *fair*. Skenario pengujian tersebut yakni :

- Memastikan bahwa semua *node* telah terhubung ke dalam jaringan *ad hoc*.
- Memasukan *username* dan *password* sesuai dengan akun SIP yang didapatkan dari server SIP voiprakyat.or.id
- Setelah itu menunggu akun SIP masing-masing *smartphone* telah terotentikasi oleh server SIP sampai siap digunakan.
- Salah satu dari *smartphone* tersebut masukan *Caller ID* atau no telepon yang dituju.
- Kirimkan data suara dengan durasi 1,830 detik dan *node-node* bergerak bebas.
- Capture* data dengan *software* wireshark untuk menghitung hasil pengujian.

Selain skenario pengujian, pada telah diuraikan bahwa parameter untuk mengukur kualitas jaringan yang dihasilkan yakni menggunakan parameter *delay*, *jitter*, *packet loss* dan *throughput*. Proses pengujian juga dilakukan diruangan tertutup (*indoor*) dan ruangan terbuka (*outdoor*) dengan jarak antar node 0-5 meter, 6-10 meter serta 11-15 meter. Pembahasan lebih lanjut diuraikan pada subbab berikut. *Bandwith* yang digunakan dalam proses pengujian 2 Mbps.

a. Delay

Delay adalah waktu yang dibutuhkan oleh data untuk sampai pada tujuan. Besarnya *delay* dapat diukur pada saat bit pertama dikirimkan sampai bit terakhir diterima di sisi penerima. Persamaan *delay* dapat dituliskan seperti pada persamaan 1.

Pada proses pengujian di ruangan tertutup (*indoor*) diperoleh hasil dengan mengambil data melalui *software wireshark* dan dihitung menggunakan persamaan 1. Untuk menghitung *delay* komunikasi VoIP melalui *statistics -> summary* di *wireshark* seperti pada Gambar 16 ambil data *between first and last packet* sebagai jumlah waktu paket dan *packet* sebagai total paket yang terkirim.

Traffic	Captured	Displayed	Displayed %	Marked	Marked %
Packets	6487	6405	98,736%	0	0,000%
Between first and last packet	75,910 sec	60,624 sec			
Avg. packets/sec	85,457	105,651			
Avg. packet size	215 bytes	217 bytes			
Bytes	1394618	1386912	99,447%	0	0,000%
Avg. bytes/sec	18372,089	22877,298			
Avg. MBit/sec	0,147	0,183			

Gambar 16. Summary Data VoIP

Pada proses pengujian di dalam ruangan (*indoor*) diperoleh hasil pada Tabel 4. Sedangkan hasil pengujian di ruangan terbuka (*outdoor*) disajikan pada Tabel 5.

Tabel 4. Delay Indoor

Jarak (m)	Jml Total Waktu	Total paket	Delay (s)
0-5	60, 624	6405	0,009465105

Jarak (m)	Jml Total Waktu	Total paket	Delay (s)
6-10	55,121	6740	0,008178191
11-15	144,420	9875	0,014624811

Tabel 5. Delay Outdoor

Jarak (m)	Jml Total Waktu	Total paket	Delay (s)
0-5	43,672	7193	0,0060714585
6-10	73,592	11504	0,0063970791
11-15	58,829	5989	0,0098222842

Hasil pengujian pada Tabel 4 dan Tabel 5 dapat dibandingkan bahwa komunikasi aplikasi telepon VoIP di dalam jaringan MANET dapat berjalan dengan baik meskipun antara satu *node* dengan *node* yang lain dapat bergerak bebas. Perbandingan hasil pengujian menunjukkan bahwa perbandingan antara *delay indoor* dan *outdoor delay* yang paling besar terjadi pada saat komunikasi rata-rata terjadi di dalam ruangan. Hal tersebut terjadi karena pada proses pengujian tempat yang digunakan memiliki penghalang dinding ruangan. Sedangkan untuk pengujian *delay* pada ruangan yang tidak memiliki penghalang dinding ruangan sementara belum dilakukan mengingat tempat pengujian dilakukan di dalam ruangan yang memiliki penghalang dinding ruangan.

b. Packet Loss

Packet loss adalah banyaknya paket yang hilang pada suatu jaringan paket yang disebabkan oleh *Collision*, penuhnya kapasitas jaringan, dan *packet drop* yang disebabkan oleh habisnya *time to live* paket. Persamaan untuk menghitung *packet loss* dapat diperlihatkan pada Persamaan 3.

Sehingga proses pengujian baik *indoor* maupun *outdoor* aplikasi VoIP di jaringan MANET mendapatkan *packet loss* dengan menghitung menggunakan Persamaan 3 diperlihatkan pada Tabel 6 untuk pengujian *indoor* dan Tabel 7 untuk pengujian *outdoor*.

Tabel 6. Pakcet Loss Indoor

Jarak (m)	Total Paket	Paket Terkirim	Packet Loss
0-5	6487	6407	1,23%
6-10	6816	6740	1,11%
11-15	9875	9666	2,11%

Tabel 7. Pakcet Loss Outdoor

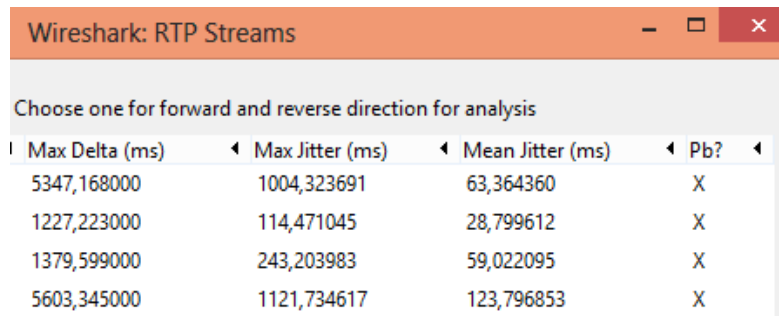
Jarak (m)	Total Paket	Paket Terkirim	Packet Loss
0-5	7320	7193	1,73%
6-10	11732	11506	1,93%
11-15	6100	5991	1,79%

Perbandingan hasil pengujian antara *indoor* dan *outdoor* pada Tabel 7 dan Tabel 8 terlihat bahwa *packet loss* paling besar terjadi pada *indoor*. Namun dari pengujian baik *indoor* maupun *outdoor* rata-rata memiliki *packet loss* yang hampir sama. Hal tersebut memperlihatkan bahwa rata-rata *packet loss* yang dihasil hampir sama. Standar nilai *packet loss* jaringan yang ditentukan CISCO adalah $< 5\%$, sedangkan

nilai rata-rata *packet loss* yang dihasilkan aplikasi VoIP di dalam jaringan MANET masih diberada di bawah 5%. Hal tersebut menunjukkan bahwa jaringan MANET dapat digunakan untuk aplikasi VoIP.

c. Jitter

Jitter adalah variasi dari *delay*. *Jitter* disebabkan oleh adanya variasi waktu dalam kedatangan paket. Variasi kedatangan paket ini dapat disebabkan oleh panjang antrian data, lamanya waktu pengolahan data dan lamanya waktu yang dibutuhkan untuk menghimpun paket-paket data yang akhir proses transmisi untuk menjadi satu kesatuan *frame* yang utuh. *Software wireshark* telah dilengkapi fitur untuk mengetahui *jitter* komunikasi yang terjadi pada aplikasi VoIP di jaringan MANET yakni dengan cara membuka menu tab *Telephony* -> *RTP* -> *Show All Streams*. *RTP* steams diperlihatkan seperti pada Gambar 17.



Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
5347,168000	1004,323691	63,364360	X
1227,223000	114,471045	28,799612	X
1379,599000	243,203983	59,022095	X
5603,345000	1121,734617	123,796853	X

Gambar 17. Capture Jitter Wireshark

Data *jitter* yang dihasil oleh *software wireshark* menggunakan satuan *milliseconds* (ms) namun dalam penelitian ini data tersebut dikonversikan dengan satuan *seconds*.

Tabel 8. Jitter

Jarak (m)	Indoor (s)	Outdoor (s)
0-5	0,01110999	0,01343492
6-10	0,01585744	0,03442137
11-15	0,01720045	0,06336436

Pada Tabel 8 menunjukkan bahwa *jitter* yang dihasilkan dalam *outdoor* memiliki variasi kedatangan paket data lebih besar jika dibandingkan *indoor*. Hal tersebut dapat dilihat dari pengaruh *delay* yang dihasilkan pada *outdoor*. Ketika *jitter outdoor* lebih besar namun *delay outdoor* yang dihasilkan kecil maka jaringan tidak bisa dikatakan jelek karena besarnya *jitter* dapat dikompensasikan dengan nilai *delay* yang kecil.

d. Throughput

Throughput adalah kecepatan *data rate* transfer efektif yang diukur dalam bps. *Throughput* juga mengacu pada banyaknya data yang dapat dikirim dalam satuan waktu. Hal ini sangat bergantung pada ketersediaan *bandwidth* pada jaringan. Persamaan *throughput* diperlihatkan pada persamaan 2.

Hasil pengujian baik *indoor* maupun *outdoor* aplikasi VoIP di dalam jaringan MANET diperoleh seperti pada Tabel 9 dan Tabel 10. Hasil tersebut dihitung dengan menggunakan Persamaan 3.

Tabel 9. Throughput Indoor

Jarak (m)	Jumlah data yang dikirim	Waktu pengiriman data	Throughput (kbps)
0-5	26444,071	55,121	479,745
6-10	22877,298	60,624	377,363
11-15	18766,999	111,040	169,011

Tabel 10. Throughput Outdoor

Jarak (m)	Jumlah data yang dikirim	Waktu pengiriman data	Throughput (kbps)
0-5	27701,293	57,474	481,979
6-10	33675,771	73,592	457,600
11-15	22030,722	58,829	374,487

Hasil pengujian pada Tabel 9 dan Tabel 10 menunjukkan *throughput* yang dihasilkan antara *indoor* dan *outdoor* hampir sama. Namun *throughput* yang paling kecil dihasilkan pengujian *indoor* dengan jarak 11-15 meter. Pengaruh dinding ruangan menjadi faktor penyebab pengujian *indoor* tersebut menghasilkan *throughput* yang kecil. Hal tersebut menunjukkan bahwa *throughput* yang dihasilkan baik *outdoor* maupun *indoor* aplikasi VoIP masih layak di jaringan MANET, karena *throughput* merupakan parameter untuk mengukur kemampuan sebenarnya suatu jaringan dalam melakukan pengiriman data.

5. Kesimpulan

Berdasarkan dari hasil implementasi, pengujian dan analisis aplikasi VoIP di jaringan MANET dapat disimpulkan bahwa aplikasi *VoIP call* dapat berjalan di dalam jaringan *mobile ad hoc network* (MANET). *Delay* paling besar yang dihasilkan adalah 0,014624811 *seconds* pada pengujian di dalam ruangan (*indoor*) dengan jarak 11-10 meter. Berdasarkan standar ITU-T, kualitas VoIP dapat dikatakan baik jika berada *range delay* 0-150 ms. Dengan demikian, dapat disimpulkan bahwa kualitas komunikasi yang dihasilkan aplikasi VoIP di dalam jaringan MANET memiliki kualitas yang baik. *Packet loss* yang dihasilkan pada pengujian baik *indoor* maupun *outdoor* berada pada *range* 1% - 2%. CISCO menetapkan bahwa standard packet loss yang untuk layanan aplikasi VoIP adalah < 5%. Hal tersebut menunjukkan bahwa packet loss yang dihasilkan masih berada pada batas yang ditentukan. *Jitter* yang dihasilkan berada pada rentang antara 0,01 – 0,06 *seconds*. Padahal, standard yang ditetapkan oleh CISCO adalah ≤ 30 ms atau $\leq 0,03$ *seconds*. Walaupun dalam rata-rata pengujian memenuhi standard tersebut, pengujian di *outdoor* dengan jarak 10-11 meter memiliki *jitter* yang nilainya melampaui standard yakni sebesar 0,06336436 *seconds*. *Throughput* yang dihasilkan pada proses pengujian adalah antara 161 kbps – 481 kbps. Hal ini menunjukkan bahwa kemampuan jaringan MANET untuk layanan telepon VoIP masih sangat baik dalam melakukan pengiriman data. Faktor yang mempengaruhi kurang baiknya komunikasi VoIP di jaringan MANET adalah dinding pembatas di dalam ruangan. Karena, berdasarkan hasil pengujian rata-rata baik parameter *delay*, *jitter*, *packet loss* maupun *throughput*, yang menghasilkan nilai paling kecil adalah pengujian di dalam ruangan. *Hidden node* dan propagasi sinyal juga merupakan penyebab faktor komunikasi aplikasi VoIP kurang baik. Hal tersebut ditunjukkan dengan hasil pengujian di dalam ruangan dengan jarak 11-15 meter yang memiliki kualitas suara kurang baik. Faktor lain yang mempengaruhi performa aplikasi VoIP di MANET akan dieksplorasi pada penelitian selanjutnya.

6. Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada semua pihak yang telah membantu penulis dalam menyelesaikan karya ilmiah ini, terutama Universitas Janabadra.

Daftar Pustaka

- Barcel, J., Bellalta, B., Cano, C., & Sfairopoulou, A. (2008). VOIP Packet Delay in Single-Hop Ad-Hoc IEEE 802.11 networks. In *Wireless on Demand Network Systems and Services* (pp. 77–80). IEEE.
- Chang, L. H., Sung, C. H., Chiu, S. Y., & Lin, Y. W. (2010). Design and realization of ad-hoc VoIP with embedded p-SIP server. *Journal of Systems and Software*, 83(12), 2536–2555. doi:10.1016/j.jss.2010.07.053
- El Brak, S., Bouhorma, M., & Boudhir, a. a. (2012). VoIP over VANETs (VoVAN): A QoS measurements analysis of inter-vehicular voice communication in urban scenario. *2012 5th International Conference on New Technologies, Mobility and Security - Proceedings of NTMS 2012 Conference and Workshops*. doi:10.1109/NTMS.2012.6208691

- Fahdi, D., Patih, J., Fitriawan, H., & Yuniati, Y. (2012). Analisa Perancangan Server Voip (Voice Internet Protocol) Dengan Opensource Asterisk Dan Vpn (Virtual Private Network) Sebagai Pengaman Jaringan Antar Client, *1*(1), 42–48.
- Fasolo, E., Maguolo, F., Zanella, A., Zorzi, M., Ruffino, S., Stupar, P., & Italia, T. (2007). VoIP Communications in Wireless Ad-hoc Network with Gateways, 69–74.
- Setiawan, E. B. (2012a). ANALISA QUALITY OF SERVICES (QoS) VOICE OVER INTERNET PROTOCOL (VoIP) DENGAN PROTOKOL H . 323 DAN SESSION INITIAL PROTOCOL (SIP). *Jurnal Ilmiah Komputer Dan Informatika (KOMPUTA)*, *1*(2).
- Setiawan, E. B. (2012b). Analisa Quality Of Services (Qos) Voice Over Internet Protocol (Voip) Dengan Protokol H. 323 Dan Session Initial Protocol (Sip). *UNIKOM, Bandung*.
- Setyawan, R. A., Sulisty, S., Hantono, B. S., Grafika, J., & Yogyakarta, N. (2014). Review : Algoritma Kriptografi Untuk Pengembangan Aplikasi Telepon Anti Sadap di Android, 7–8.
- Suryawan, K. D., Husni, M., & Anggraini, E. L. (2012). Analisis Layanan Kinerja Jaringan VoIP Pada Protokol SRTP Dan VPN. *ANALISIS LAYANAN KINERJA JARINGAN VoIP PADA PROTOKOL SRTP DAN VPN*.