ANALISIS KEAMANAN DAN PERFORMANSI VOIP (*VOICE OVER INTERNET PROTOCOL*) MENGGUNAKAN VPN (*VIRTUAL PRIVATE NETWORK*) PADA MEDIA *WIRED* DAN *WIRELESS*

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK TELEKOMUNIKASI

Ditujukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik

ARIF PUTRA ADHITAMA

NIM. 135060301111038

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2018

ANALISIS KEAMANAN DAN PERFORMANSI VOIP (*VOICE OVER INTERNET PROTOCOL*) MENGGUNAKAN VPN (*VIRTUAL PRIVATE NETWORK*) PADA MEDIA *WIRED* DAN *WIRELESS*

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK TELEKOMUNIKASI

Ditujukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik

ARIF PUTRA ADHITAMA

NIM. 135060301111038

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2018

LEMBAR PENGESAHAN

ANALISIS KEAMANAN DAN PERFORMANSI VOIP (*VOICE OVER INTERNET PROTOCOL*) MENGGUNAKAN VPN (*VIRTUAL PRIVATE NETWORK*) PADA MEDIA *WIRED* DAN *WIRELESS*

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK TELEKOMUNIKASI

Ditujukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik

ARIF PUTRA ADHITAMA
NIM. 135060301111038

Dosen Pembimbing I	**Dosen Pembimbing II**
Ir. Wahyu Adi Priyono, M.T.	**Ali Mustofa, S.T., M.T.**
NIP. 19600518 198802 1 001	NIP. 19710601 200003 1 001

Mengetahui,

Ketua Jurusan Teknik Elektro

Hadi Suyono, S.T., M.T., Ph.D.
NIP. 19730520 200801 1 013

JUDUL SKRIPSI :

ANALISIS KEAMANAN DAN PERFORMANSI VOIP (*VOICE OVER INTERNET PROTOCOL*) MENGGUNAKAN VPN (*VIRTUAL PRIVATE NETWORK*) PADA MEDIA *WIRED* DAN *WIRELESS*

Nama Mahasiswa : Arif Putra Adhitama
NIM : 135060301111038
Program Studi : Teknik Elektro
Konsentrasi : Teknik Telekomunikasi

KOMISI PEMBIMBING :

Ketua : Ir. Wahyu Adi Priyono, M.T. ..……

Anggota : Ali Mustofa, S.T., M.T. …….

TIM DOSES PENGUJI :

Dosen Penguji 1 : Ir. Sigit Kusmaryanto, M.Eng.Sc. ……..

Dosen Penguji 2 : Rudy Yuwono, S.T., M.Sc. ……..

Dosen Penguji 3 : Ir. Erfan Achmad Dahlan, M.T. ……..

Tanggal Ujian : 29 juni 2018
SK Penguji : 1285/UN10.F07/SK/2018

JUDUL SKRIPSI :

ANALISIS KEAMANAN DAN PERFORMANSI VOIP (*VOICE OVER INTERNET PROTOCOL*) MENGGUNAKAN VPN (*VIRTUAL PRIVATE NETWORK*) PADA MEDIA *WIRED* DAN *WIRELESS*

Nama Mahasiswa : Arif Putra Adhitama
NIM : 135060301111038
Program Studi : Teknik Elektro
Konsentrasi : Teknik Telekomunikasi

KOMISI PEMBIMBING :

Ketua : Ir. Wahyu Adi Priyono, M.T. ..……

Anggota : Ali Mustofa, S.T., M.T. …….

TIM DOSES PENGUJI :

Dosen Penguji 1 : Ir. Sigit Kusmaryanto, M.Eng.Sc. ……..

Dosen Penguji 2 : Rudy Yuwono, S.T., M.Sc. ……..

Dosen Penguji 3 : Ir. Erfan Achmad Dahlan, M.T. ……..

Tanggal Ujian : 29 juni 2018
SK Penguji : 1285/UN10.F07/SK/2018

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya dan berdasarkan hasil penelusuran berbagai karya ilmiah, gagasan dan masalah ilmiah yang diteliti dan diulas di dalam Naskah Skripsi ini adalah asli dari pemikiran saya. Tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu Perguruan Tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam naskah Skripsi ini dapat dibuktikan terdapat unsur-unsur jiplakan, saya bersedia Skripsi dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, pasal 25 ayat 2 dan pasal 70).

Malang, 29 Juni 2018

Mahasiswa,

ARIF PUTRA ADHITAMA

NIM. 135060301111038

Teriring Ucapan Terima Kasih kepada:

Ayahanda dan Ibunda tercinta

PENGANTAR

Bismillahirrohmanirrohim. Alhamdulillah, puji syukur kepada Allah SWT karena hanya dengan rahmat, ridho, dan kasih sayang-Nya penulis dapat menyelesaikan skripsi yang berjudul ”Analisis Keamanan Dan Performansi VoIP *(Voice Over Internet Protocol)* Menggunakan VPN *(Virtual Private Network)* Pada Media *Wired* Dan *Wireless*” dengan baik. Tak lepas shalawat serta salam selalu tercurahkan kepada junjungan kita Nabi Muhammad SAW yang telah menjadi suri tauladan dalam menghadapi segala permasalahan. Skripsi ini disusun dalam rangka untuk memenuhi persyaratan memperoleh gelar Sarjana Teknik, di Fakultas Teknik Universitas Brawijaya.

Penulisan karya tulis skripsi ini tidak dapat terselesaikan dengan baik dan lancar tanpa dukungan dan bantuan dari berbagai pihak baik secara langsung maupun tidak langsung. Secara khusus penulis ingin mengucapkan terima kasih sebesar-besarnya kepada Ayahanda Samsul Arifin dan Ibunda Endah Kusumawati yang dengan penuh kasih sayang dan kesabaran telah mengasuh, membesarkan, mendidik, memberikan pelajaran hidup yang tak ternilai harganya.

Penulis menyadari bahwa penyusunan skripsi ini tidak lepas dari bantuan, bimbingan serta dorongan dari berbagai pihak. Pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar – besarnya kepada:

1. Bapak Hadi Suyono, S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Elektro Universitas Brawijaya.
2. Ibu Ir. Nurussa’adah, M.T. selaku Sekretaris Jurusan Teknik Elektro Universitas Brawijaya.
3. Bapak Ali Mustofa, S.T. ,M.T. selaku Ketua Program Studi S1 Teknik Elektro.
4. Ibu Rusmi Ambarwati, S.T., M.T. selaku Ketua Kelompok Dosen Keahlian Telekomunikasi Program Studi S1 Jurusan Teknik Elektro Universitas Brawijaya.
5. Bapak Ir. Wahyu Adi Prijono, M.T dan Ali Mustofa, S.T. ,M.T. selaku dosen pembimbing yang telah memberikan nasehat, arahan, motivasi, saran dan masukan yang sangat bermanfaat.
6. Bapak dan Ibu dosen Jurusan Teknik Elektro Universitas Brawijaya yang tidak dapat penulis sebutkan satu per satu yang telah memberikan bekal ilmu kepada

penulis dalam menyelesaikan studi dan sebagai bekal untuk mempelajari ilmu setelah lulus.

7. Teman-teman yang selalu memberikan semangat, Galoh Utomo, Habib Nurseha, Dandhi Tri Laksono, Fahmi Rizaldi, Ariska Dwi Kartini, Adrian Satria Permana, M. Yodi Satria, M. Azril Muttaqin and saudaraku M. Danang Mirza.
8. Keluarga besar Teknik Elektro angkatan 2013, teman-teman Telekomunikasi 2013 atas do'a, semangat, serta dukungan yang diberikan pada penulis.
9. Semua pihak yang tidak dapat disebutkan satu persatu sehingga skripsi ini dapat diselesaikan dengan baik.

Dalam penyusunan skripsi ini, penulis menyadari bahwa skripsi ini belumlah sempurna, karena keterbatasan ilmu dan kendala–kendala lain yang terjadi selama pengerjaan skripsi ini. Oleh karena itu, penulis berharap kritik dan saran untuk penyempurnaan tulisan di masa yang akan datang. Semoga tulisan ini dapat bermanfaat dan dapat digunakan untuk pengembangan lebih lanjut.

Malang, 26 Juli 2018

Penulis

RINGKASAN

Arif Putra Adhitama, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya, Juli 2018, *Analisis Keamanan Dan Performansi VoIP (Voice Over Internet Protocol) Menggunakan VPN (Virtual Private Network) Pada Media Wired Dan Wireless*, Dosen Pembimbing: Ir. Wahyu Adi Prijono, MT., Ali Mustofa, ST., MT.

Voice Over Internet Protocol (VoIP) dikenal juga dengan sebutan IP *Telephony* didefinisikan sebagai suatu sistem yang menggunakan jaringan internet untuk mengirimkan data paket suara dari suatu tempat ke tempat yang lain menggunakan perantara protokol IP. Namun, penggunaan VoIP memiliki kelemahan yaitu data komunikasi tidak terjamin keamanannya. Untuk mendapatkan jaminan keamanan sambungan VoIP, maka digunakan *Virtual Private Network* (VPN) yang dapat menambahkan enkripsi pada data percakapan. Penelitian ini dilakukan untuk mengetahui tingkat keamanan dari layanan VoIP sebelum dan sesudah diamankan menggunakan VPN.

Penelitian ini dilakukan untuk mengetahui perubahan performansi dan tingkat keamanan dari layanan VoIP sebelum dan sesudah diamankan menggunakan VPN pada media *wired* dan *wireless*. Parameter yang digunakan untuk menganalisa performansi VoIP meliputi *delay end to end, packet loss*, dan *throughput*.

Dari hasil perhitungan simulasi, kualitas performansi VoIP jaringan tanpa VPN dan VPN pada media *wired* dan *wireless* telah sesuai dengan standar TIPHON. Untuk *delay end to end* masuk dalam indeks *delay* < 150 ms atau "Sangat Bagus", kemudian untuk *packet loss* masuk dalam indeks 0-3 % atau "Bagus", dan *throughput* masuk dalam indeks 100-75 % atau "Sangat Bagus".

Kata kunci: VoIP, VPN, *delay end to end*, *packet loss*, dan *throughput***.**

SUMMARY

***Arif Putra Adhitama**, Electrical Engineering Departement, Engineering Faculty, Brawijaya University, July 2018, VoIP (Voice Over Internet Protocol) Security And Performance Analysis Using VPN (Virtual Private Network) On Wired And Wireless Media, Academic supervisors: Ir. Wahyu Adi Prijono, MT., Ali Mustofa, ST., MT.*

Voice Over Internet Protocol (VoIP) also known as IP Telephony is defined as a system that uses Internet network to transmit voice packet data from one place to another using IP protocol intermediaries. However, the use of VoIP has the disadvantage of data communication is not guaranteed security. To get the security guarantee of VoIP connection, it is used Virtual Private Network (VPN) which can add encryption to data of conversation. This study was conducted to determine the security level of VoIP service before and after secured using VPN.

This research was conducted to find out the change of performance and security level of VoIP service before and after secured using VPN on wired and wireless media. Parameters used to analyze VoIP performance include end to end delay, packet loss, and throughput.

From the results of simulation calculations, the quality of VoIP network performance without VPN and VPN on wired and wireless media has been in accordance with TIPHON standards. For end to end delays in the delay index <150 ms or "Very Good", then for packet loss in the index 0-3% or "Good", and throughput enter in the index of 100-75% or "Very Good".

***Keywords:** VoIP, VPN, end to end delay, packet loss, and throughput.*

DAFTAR ISI

Halaman

PENGANTAR i
RINGKASAN iii
SUMMARY iv
DAFTAR ISI v
DAFTAR TABEL viii
DAFTAR GAMBAR ix
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Batasan Masalah 2
1.4 Tujuan 2
1.5 Sistematika Penulisan 2
BAB II TINJAUAN PUSTAKA 5
2.1 Umum 5
2.2 Protokol TCP/IP 5
2.3 IP *Header* 7
2.4 TCP dan UDP *Header* 9
2.4.1 TCP *Header* 9
2.4.2 UDP *Header* 11
2.5 *Voice Over Internet Protocol* (VoIP) 12
2.6 Unsur Pembentuk VoIP 13
2.6.1 *User agent* 13
2.6.2 *Proxy* 13
2.6.3 Protokol 13
2.6.3.1 *Session Initiation Protocol* (SIP) 13
2.6.3.2 H.323 15
2.6.4 *Codec* 16
2.7 Cara Kerja VoIP 17
2.8 Format Paket VoIP 17
2.9 Sistem Panggilan VoIP 18
2.10 *Virtual Private Network* (VPN) 19
2.11 Prinsip Kerja VPN 19
2.12 Keunggulan dan Kelemahan 20
2.12.1 Keunggulan VPN 20
2.12.2 Kelemahan VPN 20
2.13 Protokol VPN 21
2.14 Media Transmisi 22

2.14.1 *Wired* 22
2.14.1.1 *Unshielded Twisted Pair* (UTP) 22
2.14.1.2 *Shielded Twisted Pair* (STP) 23
2.14.2 *Wireless* 24
2.14.2.1 Standarisasi *Wireless* 24
2.15 *Parameter Quality of Service* (QoS) 24
2.15.1 Paket Data VoIP 25
2.15.2 Perhitungan SNR 25
2.15.3 Perhitungan *Bit rate* 25
2.15.4 Perhitungan BER 26
2.15.5 *Delay end to end* 27
2.15.6 *Packet Loss* 31
2.15.7 *Throughput* 32

BAB III METODE PENELITIAN 33

3.1 Umum 33
3.2 Identifikasi Data 33
3.3 Pengambilan Data 33
3.3.1 Pengambilan Data Primer 33
3.3.2 Pengambilan Data Sekunder 34
3.4 Analisis Data 34
3.4.1 Diagram Alir metode analisis perhitungan 34
3.4.2 Diagram Alir metode simulasi 35
3.5 Topologi Jaringan 37
3.6 Konfigurasi Jaringan 38
3.6.1 Konfigurasi Jaringan *Wired* 38
3.6.2 Konfigurasi Jaringan *Wireless* 39
3.7 Identifikasi Perangkat Pendukung 40
3.8 Pengambilan Kesimpulan dan Saran 41

BAB IV HASIL DAN PEMBAHASAN 43

4.1 Umum 43
4.2 Analisa Konfigurasi Jaringan 43
4.2.1 Konfigurasi tanpa VPN 43
4.2.2 Konfigurasi dengan VPN 44
4.3 Analisis Perhitungan pada JaringanVoIP 45
4.3.1 Analisis Paket Data Aplikasi VoIP 45
4.3.2 Analisis *Signal to Noise Ratio* (SNR) 45
4.3.3 Penentuan *Bit rate* 46
4.3.4 Analisis *Bit Error Rate* (BER) 47
4.3.5 Analisis *Delay end to end* 48
4.3.6 Analisis *Packet loss* 52

4.3.7 Analisis *Throughput* 53
4.4 Analisis VoIP pada Jaringan VPN 53
4.4.1 Analisis *Delay end to end* 54
4.4.2 Analisis *Packet loss* 57
4.4.3 Analisis *Throughput* 58
4.5 Analisis Simulasi VoIP 59
4.5.1 Analisis *Delay end to end* 60
4.5.2 Analisis *Packet loss* 62
4.5.3 Analisis *Throughput* 63
4.6 Perbandingan Hasil Perhitungan dengan Simulasi 65
4.6.1 *Delay end to end* 65
4.6.2 *Packet loss* 66
4.6.3 *Throughput* 67
4.7 Pengujian Keamanan VoIP 69

BAB V PENUTUP 71

5.1 Kesimpulan 71
5.2 Saran 71

DAFTAR PUSTAKA 72

LAMPIRAN 73

DAFTAR TABEL

Halaman

Tabel 2.1	Teknik kompresi standar ITU-T	16
Tabel 2.2	Indeks *Delay end to end*	27
Tabel 2.3	Indeks *Packet Loss*	31
Tabel 2.4	Indeks *Throughput*	32
Tabel 4.1	Spesifikasi *Codec* G.711	45
Tabel 4.2	Parameter OFDM	46
Tabel 4.3	*Receiver SNR Standard* IEEE 802.11n dan 802.3u	46
Tabel 4.4	Hasil Perhitungan *Delay end to end*	52
Tabel 4.5	Hasil Perhitungan *Packet Loss*	53
Tabel 4.6	Hasil Perhitungan *Throughput*	53
Tabel 4.7	Hasil Perhitungan *Delay end to end* VPN	57
Tabel 4.8	Hasil Perhitungan *Packet Loss* VPN	58
Tabel 4.9	Hasil Perhitungan *Throughput* VPN	59
Tabel 4.10	Hasil Pengukuran *Delay end to end*	61
Tabel 4.11	Hasil Pengukuran *Packet Loss*	62
Tabel 4.12	Hasil Pengukuran *Throughput*	64
Tabel 4.13	Perbandingan nilai *delay end to end wired* perhitungan dan simulasi	65
Tabel 4.14	Perbandingan nilai *delay end to end wireless* perhitungan dan simulasi	66
Tabel 4.15	Perbandingan nilai *packet loss wired* perhitungan dan simulasi	66
Tabel 4.16	Perbandingan nilai *packet loss wireless* perhitungan dan simulasi	67
Tabel 4.17	Perbandingan nilai *throughput wired* perhitungan dan simulasi	68
Tabel 4.18	Perbandingan nilai *throughput wireless* perhitungan dan simulasi	68

DAFTAR GAMBAR

Halaman

Gambar 2.1 *Layer* TCP/IP 5
Gambar 2.2 *Datagram* IP *Header* 7
Gambar 2.3 Format *Header* TCP 9
Gambar 2.4 Format *Header* UDP 11
Gambar 2.5 Desain arsitektur jaringan VoIP 17
Gambar 2.6 Format Paket VoIP 18
Gambar 2.7 Proses komunikasi VoIP pada protokol SIP 18
Gambar 2.8 Prinsip kerja VPN 19
Gambar 2.9 Format paket PPTP 21
Gambar 2.10 Kabel UTP 22
Gambar 2.11 Kabel STP 23
Gambar 2.12 Model antrian M/M/1 30
Gambar 3.1 Diagram Alir metode analisis perhitungan 35
Gambar 3.2 Diagram Alir metode simulasi 36
Gambar 3.3 Topologi jaringan *wired* 38
Gambar 3.4 Topologi jaringan *wireless* 38
Gambar 4.1 Tampilan *Traceroute* pada jaringan tanpa VPN 44
Gambar 4.2 Tampilan *Traceroute* pada jaringan VPN 44
Gambar 4.3 Format paket PPTP 54
Gambar 4.4 Tampilan GUI Trixbox 60
Gambar 4.5 Diagram perbandingan *Delay end to end* 61
Gambar 4.6 Diagram perbandingan *Packet Loss* 63
Gambar 4.7 Diagram perbandingan *Throughput* 64
Gambar 4.8 Perbandingan *delay end to end wired* hasil perhitungan dan simulasi .. 65
Gambar 4.9 Perbandingan *delay end to end wireless* hasil perhitungan dan simulasi 66
Gambar 4.10 Perbandingan *packet loss wired* hasil perhitungan dan simulasi 67
Gambar 4.11 Perbandingan *packet loss wireless* hasil perhitungan dan simulasi 67
Gambar 4.12 Perbandingan *throughput wired* hasil perhitungan dan simulasi 68
Gambar 4.13 Perbandingan *throughput wireless* hasil perhitungan dan simulasi 69

Gambar 4.14 *Sniffing* VoIP tanpa VPN 70
Gambar 4.15 *Sniffing* VoIP dengan VPN 70

BAB I
PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan teknologi informasi di era globalisasi ini, jaringan internet tidak hanya berfokus pada layanan paket data dan aplikasi seperti WWW (*World Wide Web*) http, smtp, dan ftp. Saat ini kebutuhan akan layanan komunikasi menjadi sangat penting, karena telekomunikasi suara merupakan media yang paling praktis untuk menyampaikan informasi. Hal ini menyebabkan hadirnya teknologi pemrosesan sinyal digital yang mempunyai kemampuan modular dengan berbasis IP (*Internet Protocol*) yang diintegrasikan dengan komunikasi data dan suara.

Voice Over Internet Protocol (VoIP) dikenal juga dengan sebutan IP *Telephony* didefinisikan sebagai suatu sistem yang menggunakan jaringan internet untuk mengirimkan data paket suara dari suatu tempat ke tempat yang lain menggunakan perantara protokol IP. Data suara diubah menjadi kode digital dan dialirkan melalui jaringan yang mengirimkan paket-paket data. Kebutuhan akan komunikasi yang murah dan handal menjadikan aplikasi ini banyak dipergunakan.

Berkembangnya layanan VoIP ini mengakibatkan munculnya masalah, salah satunya adalah mengenai masalah keamanan. VoIP merupakan layanan komunikasi berbasis IP dimana salah satu kelemahan dari jaringan IP adalah data yang terkirim tidak terjamin kerahasiaannya sehingga siapapun dapat menangkap dan memanipulasi data tersebut. Dari sinilah memunculkan pemikiran bagaimana cara mengamankan data VoIP tanpa mengurangi performansi dari jaringan VoIP itu sendiri. Salah satunya dengan cara menggunakan VPN.

Virtual Private Network (VPN) adalah suatu koneksi antara satu jaringan dengan jaringan lain secara pribadi melalui jaringan publik (internet). Jaringan VPN bersifat pribadi, dimana hanya orang tertentu saja yang dapat mengaksesnya. Sambungan VPN dilewatkan pada sebuah *tunnel* virtual antara 2 *node*, data yang melalui *tunnel* ini telah dienkripsi sehingga aman dan tetap rahasia meskipun dikirim lewat jaringan publik.

1.2 Rumusan Masalah

Berdasarkan penjelasan dari latar belakang diatas, rumusan masalah ditekankan pada :

1. Bagaimana performansi VoIP yang dihasilkan setelah dan sebelum menggunakan VPN pada media transmisi *wired* dan *wireless*?
2. Bagaimana pengaruh VPN terhadap keamanan VoIP?

1.3 Batasan Masalah

Untuk menyederhanakan penilitian maka dibawah ini merupakan batasan yang dipakai dalam melakukan penelitian sebagai berikut :

1. Pengaturan dan konfigurasi VoIP menggunakan *software* Trixbox.
2. Protokol VoIP yang digunakan adalah SIP (*Session Initiation Protocol*).
3. *Codec* yang digunakan untuk panggilan VoIP adalah *audio codec* sesuai ITU G.711
4. Protokol VPN yang digunakan adalah PPTP (*Point to Point Tunneling Protokol*) pada Mikrotik.
5. Penyadapan komunikasi VoIP menggunakan *software* Cain dan Abel.
6. Percobaan dilakukan pada dua media yang berbeda yaitu media yang menggunakan *wired* dan media yang menggunakan *wireless*.
7. Parameter yang akan dianalisa untuk menguji kualitas jaringan meliputi *delay end to end, packet loss,* dan *throughput.*

1.4 Tujuan

Skripsi ini bertujuan untuk menganalisa perubahan performansi dan tingkat keamanan dari layanan VoIP sebelum dan sesudah diamankan menggunakan VPN pada media *wired* dan *wireless* dengan menganalisa *delay end to end, packet loss,* dan *throughput*.

1.5 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penyusunan skripsi ini adalah sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi penjelasan mengenai Latar Belakang, Rumusan Masalah, Batasan Masalah, Tujuan, dan Sistematika Penulisan.

BAB II TINJAUAN PUSTAKA

Pada bab ini menguraikan dan menjelaskan teori yang menjadi dasar dan pendukung dari skripsi ini. Diantaranya teori dasar TCP/IP, VoIP, VPN, *wireless* dan *wired*.

BAB III METODOLOGI

Metodologi penulisan mencakup penjelasan mengenai identifikasi data, pengambilan data, analisis data perancangan topologi jaringan, dan konfigurasi jaringanan.

BAB IV HASIL DAN PEMBAHASAN

Membahas proses pengambilan dan analisis data pada masing-masing jaringan untuk layanan VoIP.

BAB V PENUTUP

Berisi kesimpulan dan saran yang diperoleh dari hasil analisis.

BAB II
TINJAUAN PUSTAKA

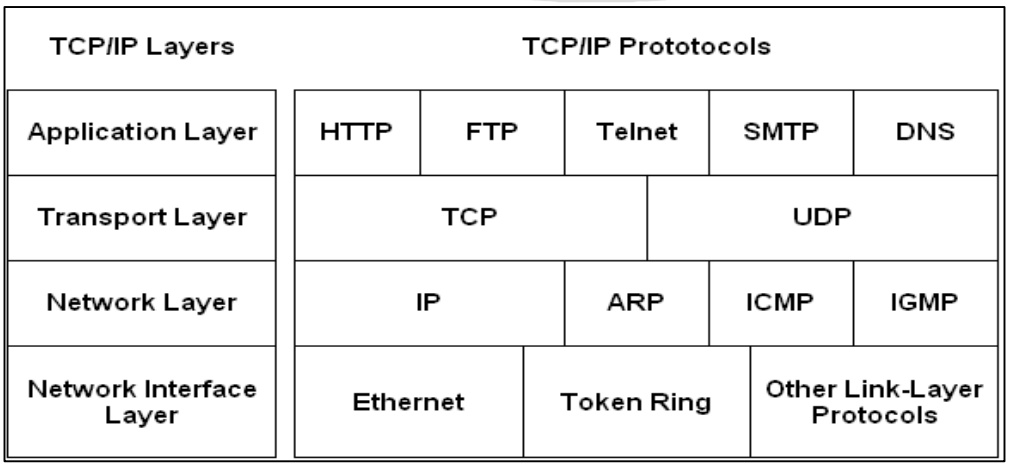
2.1 Umum

Perkembangan teknologi telekomunikasi yang begitu pesat ditunjukan dari mulai banyaknya pengguna dan layanan yang ditawarkan terutama layanan berbasis IP. VoIP merupakan layanan untuk berkomunikasi yang berbasis IP. Saat ini VoIP telah banyak digunakan karena lebih efisien dan dapat menekan biaya untuk pembuatan jaringan baru seperti pada telepon PSTN. Berkembangnya layanan VoIP ini mengakibatkan munculnya masalah, salah satunya adalah mengenai masalah keamanan. Melakukan komunikasi menggunakan VoIP sangat rentan sekali untuk di sadap maka dari itu diperlukan pengaman yaitu menggunakan VPN untuk mengenkripsi data ketika melakukan suatu komunikasi.

Pada bab ini akan dijelaskan teori penunjang mengenai TCP/IP, VoIP, dan VPN yang terdiri dari sistem layanan VoIP dan VPN serta parameter-parameter yang digunakan untuk mengetahui performansi VoIP.

2.2 Protokol TCP/IP

TCP/IP merupakan standar komunikasi data yang digunakan dalam proses tukar-menukar data dari satu komputer ke komputer lain pada jaringan Internet. Standar komunikasi protokol TCP/IP terdiri dari 4 layer yang saling berkomunikasi. Berikut ini gambaran layer pada TCP/IP.



Gambar 2.1 Layer TCP/IP
(**Sumber:** http://sourcedaddy.com/windows-xp/the-tcp-ip-protocol-framework.html)

Berikut ini penjelasan masing-masing layer tersebut :

1. ***Network Interface Layer***

Network Interface Layer atau *Network Access Layer* bertanggung jawab untuk mengirim paket TCP / IP pada media jaringan dan menerima paket TCP / IP dari media jaringan. TCP/IP dapat digunakan untuk menghubungkan berbagai jenis jaringan antara lain teknologi LAN seperti *Ethernet*, *Token Ring* dan WAN kemudian teknologi seperti X.25 dan *Frame Relay*. *Network Interface Layer* mempunyai fungsi yang mirip dengan *Data Link Layer* pada OSI. Lapisan ini mengatur enkapsulasi dari *datagram* IP menjadi *frame* yang kemudian disalurkan pada media fisik berupa serat optik, kabel, atau gelombang radio. Selain itu pada lapisan ini juga berfungsi memberikan servis untuk deteksi dan koreksi kesalahan dari data yang ditransmisikan.

2. ***Network Layer***

Network Layer bertanggung jawab untuk enkapsulasi dan fungsi *routing*. Protokol inti dari lapisan internet adalah IP, ARP, dan ICMP.

- *Internet Protocol* (IP) adalah protokol *routable* bertanggung jawab untuk pengalamatan IP, *routing*, fragmentasi dan *reassembly* paket.
- *Address Resolution Protocol* (ARP) bertanggung jawab untuk resolusi alamat lapisan Internet ke alamat lapisan *Network Interface* seperti alamat *hardware*.
- *Internet Control Message Protocol* (ICMP) bertanggung jawab untuk menyediakan fungsi diagnostik dan pelaporan kesalahan karena pengiriman tidak berhasil paket IP.

3. ***Transport Layer***

Transport Layer bertanggung jawab untuk menyediakan servis untuk *Application layer* dengan layanan *session* dan komunikasi *datagram*. *Transport Layer* terdiri dari 2 protokol utama yaitu *Transmission Control Protocol* (TCP) dan *User Datagram Protocol* (UDP).

- TCP (*Transmission Control Protocol*) menyediakan fungsi pengiriman data secara *connection-oriented*, pencegahan duplikasi data, *congestion control*, *flow control*, dan layanan komunikasi *end-to-end* yang handal. TCP juga bertanggung jawab untuk pembentukan koneksi TCP, pengurutan dan *acknowledgment* dari paket yang dikirim, dan pemulihan (retransmisi) paket yang hilang selama transmisi.

- UDP (*User Datagram Protocol*) menyediakan fungsi pengiriman *connectionless*, dan jalur yang tidak reliabel. UDP banyak digunakan pada aplikasi yang membutuhkan kecepatan tinggi dan dapat mentoleransi terhadap kerusakan data.

4. *Application Layer*

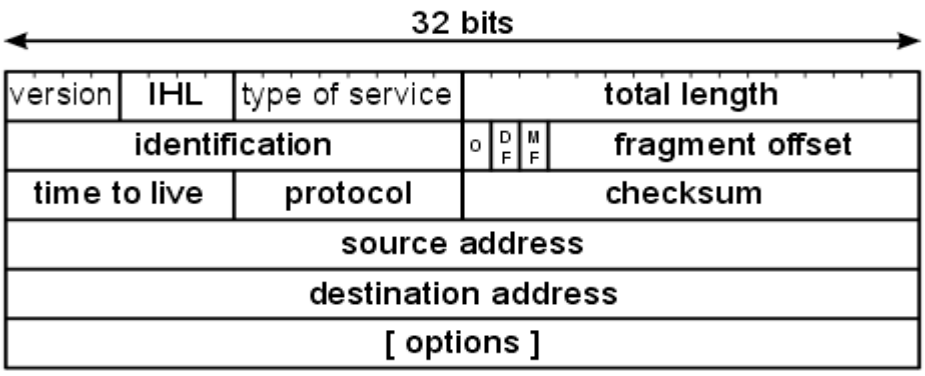
Aplikasi Layer merupakan lapisan terakhir dalam arsitektur TCP/IP yang berfungsi mendefinisikan aplikasi-aplikasi yang dijalankan pada jaringan. Terdapat banyak protokol pada lapisan ini sesuai dengan banyaknya lapisan TCP/IP yang dapat dijalankan. Contohnya adalah SMTP (*Simple Mail Transfer Protocol*) untuk pengiriman *email*, FTP (*File Transfer Protocol*) untuk transfer data, HTTP (*Hyper Text Transfer Protocol*) untuk aplikasi *web* dan lain-lain.

2.3 IP *Header*

IP menyediakan layanan pengiriman bersifat *connectionless* yang berarti bahwa data terkirim begitu saja tanpa ada pembentukan jalur terlebih dahulu (*connection set-up*), tetapi langsung dikirim dari jaringan.

Jika dalam perjalanan paket tersebut terjadi hal-hal yang tidak diinginkan (jalur terputus, *router* mengalami kongesti, atau *host/network* tujuan sedang *down*), protokol IP hanya memberitahukan pengiriman paket melalui protokol ICMP, bahwa terjadi masalah dalam pengiriman paket IP ke *host* pengiriman. Satuan dasar yang digunakan untuk transfer data adalah *datagram*.

IP juga bersifat *datagram delivery service* yang berarti setiap paket data yang dikirim adalah independen terhadap data yang lain. Akibatnya jalur yang ditempuh oleh masing-masing paket data IP ke tujuannya bisa jadi berbeda satu dengan yang lainnya. Karena jalur yang ditempuh berbeda, kedatangan paket pun jadi tidak berurutan (*out of sequence*). Berikut adalah gambar *datagram* IP *header*.



Gambar 2.2 Datagram IP Header
(**Sumber:** telescript.denayer.wenk.be)

Berdasarkan Gambar 2.2 diatas, setiap paket IP membawa data yang terdiri atas :

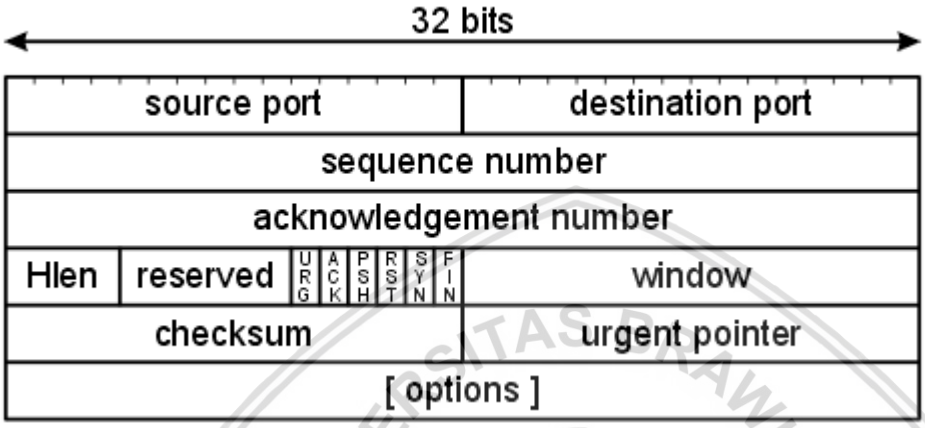
1. *Version*, berisi versi dari protokol IP yang dipakai. Saat ini protokol yang digunakan internet adalah IP versi 4.
2. *Header Length*, berisi panjang dari *header* paket IP.
3. *Type of Service*, mengindikasikan bagaimana *datagram* diproses.
4. *Total Length*, mengindikasikan panjang panjang total *datagram* dalam *byte* termasuk *header* IP dan *header* TCP/IP.
5. *Identification*, *Flags*, dan *Fragment Offset*, berisi beberapa data yang berhubungan dengan fragmentasi paket. Paket yang dilewatkan melalui berbagai jenis jalur akan mengalami fragmentasi (dipecah-pecah menjadi beberapa paket yang lebih kecil) sesuai dengan besar data maksimal yang biasa ditransmisikan melalui jalur tersebut.
6. *Time to Live*, berisi jumlah *router* maksimal yang boleh dilewati paket IP. Setiap kali paket IP melewati satu *router*, isi dari *field* ini dikurangi satu. Jika TTL telah habis dan paket tetap belum sampai tujuan, maka paket ini akan dibuang dan *router* terakhir akan mengirimkan paket ICMP *time exceeded*. Hal ini dilakukan untuk mencegah paket IP terus-menerus berada didalam *network*.
7. *Protocol*, menunjukan jenis protokol lapisan transport yang digunakan IP (isi data dari paket).
8. *Header Cheksum*, berisi nilai *cheksum* yang dihitung dari seluruh *field* dari *header* paket IP. Sebelum dikirimkan protokol terlebih dahulu menghitung *checksum* dari paket IP untuk nantinya dihitung kembali di sisi penerima. Jika terjadi perbedaan, maka paket ini dianggap rusak atau dibuang.
9. *Source* dan *Destination IP Address*, menandai nomor jaringan dan nomor *host* pengirim dan penerima. Dengan *field* ini dapat diketahui alamat pengirim dan penerima sehingga paket IP hanya akan sampai pada alamat IP yang dituju.
10. *Option*, diantaranya :

- *Strict source route*, berisi daftar lengkap IP *address* dari *router* yang harus dilalui oleh paket ini dalam perjalanan ke *host* tujuan. Balasan dari *host* tujuan ke *host* pengirim diharuskan melalui *router* yang sama.
- *Loose source route*, dengan mengeset *option* ini paket yang dikirim harus singgah dibeberapa *router*. Jika diantara kedua *router* terdapat *router* yang lain, maka paket masih dapat melalui *router* tersebut.

2.4 TCP dan UDP *Header*

2.4.1 TCP *Header*

TCP menyediakan layanan yang dikenal sebagai *connection oriented* (terjadi pembentukan hubungan sebelum mentransfer data), *reliable* (menerapkan proses deteksi *error* paket dan retransmisi) dan *byte stream service* (pengurutan paket). Susunan *Header* TCP dapat dilihat seperti pada Gambar 2.3 dibawah ini :



Gambar 2.3 Format Header TCP
(**Sumber:** http://azuldah.blogspot.co.id/2016/10/)

Berikut ini penjelasan masing-masing dari *field Header* TCP :

- *Source Port* (16 *bit*)

  Mengindikasikan sumber protokol lapisan aplikasi yang mengirimkan segmen TCP yang bersangkutan. Gabungan antara *field Source* IP *Address* dalam *header* IP dan *field Source Port* dalam *field header* TCP disebut juga sebagai *source socket*, yang berarti sebuah alamat global dari mana segmen dikirimkan.

- *Destination Port* (16 *bit*)

  Mengindikasikan tujuan protokol lapisan aplikasi yang menerima segmen TCP yang bersangkutan. Gabungan antara *field Destination* IP *Address* dalam *header* IP dan *field Destination Port* dalam *field header* TCP disebut juga sebagai *socket* tujuan, yang berarti sebuah alamat global ke mana segmen akan dikirimkan.

- *Sequence Number* (32 *bit*)

  Mengindikasikan nomor urut dari *oktet* pertama dari data di dalam sebuah segmen TCP yang hendak dikirimkan. *Field* ini harus selalu diset, meskipun tidak ada data (*payload*) dalam segmen.

  Ketika memulai sebuah sesi koneksi TCP, segmen dengan *flag* SYN (*Synchronization*) diset ke nilai 1, *field* ini akan berisi nilai *Initial Sequence Number*

(ISN). Hal ini berarti, *oktet* pertama dalam aliran *byte* (*byte stream*) dalam koneksi adalah ISN+1.

- *Acknowledgment Number* (32 *bit*)

  Mengindikasikan nomor urut dari *oktet* selanjutnya dalam aliran *byte* yang diharapkan untuk diterima oleh pengirim dari si penerima pada pengiriman selanjutnya. *Acknowledgment number* sangat dipentingkan bagi segmen-segmen TCP dengan *flag* ACK diset ke nilai 1.

- *Data Offset* (4 *bit*)

  Mengindikasikan di mana data dalam segmen TCP dimulai. *Field* ini juga dapat berarti ukuran dari *header* TCP. Seperti halnya *field Header Length* dalam *header* IP, *field* ini merupakan angka dari *word* 32-bit dalam *header* TCP. Untuk sebuah segmen TCP terkecil (di mana tidak ada opsi TCP tambahan), *field* ini diatur ke nilai 0x5, yang berarti data dalam segmen TCP dimulai dari *oktet* ke 20 dilihat dari permulaan segmen TCP. Jika *field Data Offset* diset ke nilai maksimumnya (24=16) yakni 15, *header* TCP dengan ukuran terbesar dapat memiliki panjang hingga 60 *byte*.

- *Reserved* (6 *bit*)

  Direservasikan untuk digunakan pada masa depan. Pengirim segmen TCP akan mengeset *bit-bit* ini ke dalam nilai 0.

- *Flags* (6 *bit*)

  Mengindikasikan *flag-flag* TCP yang memang ada enam jumlahnya, yang terdiri atas: URG (*Urgent*), ACK (*Acknowledgment*), PSH (*Push*), RST (*Reset*), SYN (*Synchronize*), dan FIN (*Finish*).

- *Window* (16 *bit*)

  Mengindikasikan jumlah *byte* yang tersedia yang dimiliki oleh *buffer host* penerima segmen yang bersangkutan. *Buffer* ini disebut sebagai *Receive Buffer*, digunakan untuk menyimpan *byte stream* yang datang. Dengan mengimbuhkan ukuran *window* ke setiap segmen, penerima segmen TCP memberitahukan kepada pengirim segmen berapa banyak data yang dapat dikirimkan dan disangga dengan sukses. Hal ini dilakukan agar si pengirim segmen tidak mengirimkan data lebih banyak dibandingkan ukuran *Receive Buffer*. Jika tidak ada tempat lagi di dalam *Receive buffer*, nilai dari *field* ini adalah 0. Dengan nilai 0, maka si pengirim tidak akan dapat

mengirimkan segmen lagi ke penerima hingga nilai *field* ini berubah (bukan 0). Tujuan hal ini adalah untuk mengatur lalu lintas data atau *flow control*.

- *Checksum* (16 *bit*)

  Mampu melakukan pengecekan integritas segmen TCP (*header*-nya dan *payload*-nya). Nilai *field Checksum* akan diatur ke nilai 0 selama proses kalkulasi *checksum*.

- *Urgent Pointer* (16 *bit*)

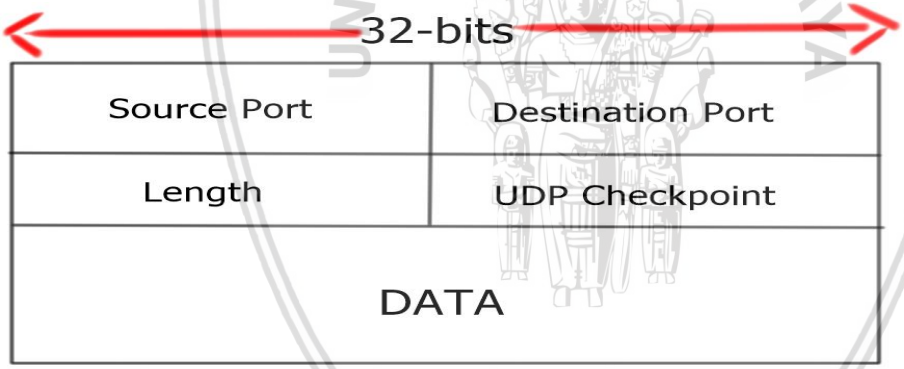
  Menandakan lokasi data yang dianggap "*urgent*" dalam segmen.

- *Options* (32 *bit*)

  Berfungsi sebagai penampung beberapa opsi tambahan TCP. Setiap opsi TCP akan memakan ruangan 32 *bit*, sehingga ukuran *header* TCP dapat diindikasikan dengan menggunakan *field Data offset*.

2.4.2 UDP *Header*

UDP (*User Datagram Protocol*) adalah salah satu protokol lapisan transport TCP/IP yang mendukung komunikasi yang tidak handal (*unreliable*), tanpa koneksi (*connectionless*) antara *host-host* dalam jaringan yang menggunakan TCP/IP.



Gambar 2.4 Format Header UDP
(**Sumber:** http://network-insight.net/2014/10/fingerprinting-with-udp-and-tcp-scans)

Berikut ini penjelasan masing-masing dari *field Header* UDP :

- *Source Port* (16 *bit*)

  Digunakan untuk mengidentifikasikan sumber protokol lapisan aplikasi yang mengirimkan pesan UDP yang bersangkutan. Penggunaan *field* ini adalah opsional, dan jika tidak digunakan, akan diset ke angka 0. Beberapa protokol lapisan aplikasi dapat menggunakan nilai *field* ini dari pesan UDP yang masuk sebagai nilai *field port* tujuan (*Destination Port*) sebagai balasan untuk pesan tersebut.

- *Destination Port* (16 *bit*)

  Digunakan untuk mengidentifikasikan tujuan protokol lapisan aplikasi yang menjadi tujuan pesan UDP yang bersangkutan. Dengan menggunakan kombinasi antara alamat IP dengan nilai dari *field* ini untuk membuat sebuah alamat yang signifikan untuk mengidentifikasikan proses yang berjalan dalam sebuah *host* tertentu yang dituju oleh pesan UDP yang bersangkutan.

- *Length* (16 *bit*)

  Digunakan untuk mengindikasikan panjang pesan UDP (pesan UDP ditambah dengan *header* UDP) dalam satuan *byte*. Ukuran paling kecil adalah 8 *byte* (ukuran *header* UDP) dan ukuran paling besar adalah 65515 *bytes* (65535 [216] -20 [ukuran *header* protokol IP]). Panjang maksimum aktual dari pesan UDP akan disesuaikan dengan menggunakan nilai *Maximum Transmission Unit* (MTU) dari saluran di mana pesan UDP dikirimkan. *Field* ini bersifat terulang-ulang. Panjang pesan UDP dapat dihitung dari *field Length* dalam *header* UDP dan *field* IP *Header Length* dalam *header* IP.

- *Checksum* (16 *bit*)

  Berisi informasi pengecekan integritas dari pesan UDP yang dikirimkan (*header* UDP dan pesan UDP). Penggunaan *field* ini adalah opsional. Jika tidak digunakan, *field* ini akan bernilai 0.

2.5 *Voice Over Internet Protocol* (VoIP)

IP Telephony atau yang biasa disebut dengan VoIP (*Voice Over Internet Protocol*) merupakan teknologi yang memanfaatkan IP (*Internet Protocol*) untuk menyediakan komunikasi suara secara *real-time*. VoIP merupakan teknologi yang mampu melewatkan trafik suara yang berbentuk paket melalui jaringan IP. Jaringan IP sendiri merupakan jaringan komunikasi data yang berbasis *packet-switch*. Sinyal suara sebelum dipaketkan mengalami *voice coding* atau pengubahan format suara kedalam bentuk digital agar dapat dilewatkan melalui jaringan IP (Winarno, 2007).

VoIP memberikan nilai ekonomis bila dibandingkan dengan telekomunikasi menggunakan jaringan telekomunikasi secara terpisah. Nilai ekonomis dapat dicapai karena jaringan VoIP tidak membutuhkan pemasangan jaringan baru, melainkan dapat melalui jaringan internet yang telah ada. Unsur-unsur pembentuk VoIP diperlukan untuk membangun jaringan VoIP.

2.6 Unsur Pembentuk VoIP

Ada empat unsur pembentuk jaringan VoIP, yaitu : *User agent*, *Proxy*, *Protocol*, dan CODEC (*Coder-Decoder*). Dari unsur-unsur tersebut jaringan VoIP terbentuk. Untuk memahami lebih dalam, berikut akan dipaparkan satu persatu dari unsur-unsur pembentuk VoIP tersebut.

2.6.1 *User Agent*

User agent seperti layaknya telepon yang kita kenal, berfungsi untuk melakukan pemanggilan atau menerima telepon. *User agent* dapat berupa *software* atau biasa disebut dengan *softphone.* Contoh *user agent* dengan jenis *softphone* adalah : Siphone, X-Lite, QuteCom, NetMeetin, VoIP Rakyat *Communicator* dan masih banyak lagi.

2.6.2 *Proxy*

Proxy yang dimaksud dalam teknologi VoIP merupakan aplikasi *server* yang mengatur jaringan VoIP. *Proxy* merupakan komponen yang menerima registrasi *user agent* dan bertugas mengatur penomoran dan *call routing*. Terdapat dua jenis *Proxy* yang digunakan, yaitu berupa *hardware* mesin IPPBX dan berupa *software* yang disebut sebagai *softswitch*. *Softswitch* versi *open source* yang terkenal dan teruji kehandalannya adalah Trixbox, Asterisk, SER (*SIP Express Router*), dan *Yate*. Sedangkan untuk versi *non-open source* adalah Axon dan OnDo SIP *server*.

2.6.3 Protokol

Dalam membangun jaringan VoIP diperlukan protokol agar komunikasi antar terminal (*user agent*) maupun antar *proxy* bisa terjadi. Protokol merupakan sebuah standart yang harus dipenuhi agar komunikasi VoIP terjadi. Ada tiga macam protokol yang digunakan, yaitu IETF (*Internet Enginering Task Force)* yang lebih dikenal dengan *Sesion Initation Protokol* (SIP), protokol H.323 yang dikembangkan oleh ITU-T (*International Telecommunication Union-Telecommunication)* dan protokol asterisk yang dikenal dengan sebutan IAX (*The Inter-Asterik Exchange*) (Winarno, 2007).

2.6.3.1 *Session Initiation Protocol* (SIP)

Session Initiation Protocol (SIP) adalah standar IETF (*Internet Engineering Task Force)* yaitu protokol pensinyalan pada layer aplikasi yang berbasis ASCII dan berfungsi untuk membangun, memodifikasi, dan mengakhiri suatu sesi multimedia yang melibatkan satu atau

beberapa pengguna. Sesi multimedia adalah pertukaran arus data antar pengguna yang meliputi suara, video, atau teks.

- **Komponen SIP**

Dalam hubungannya dengan IP *Telephony*, ada dua komponen yang ada dalam sistem SIP, yaitu :

1. *User Agent*

   User Agent merupakan sistem akhir (*end system*) yang digunakan untuk berkomunikasi. *User Agent* terdiri atas 2 bagian, yaitu:

   1. *User Agent Client* (UAC)

      UAC merupakan aplikasi pada *client* yang didesain untuk memulai SIP *request*.

   2. *User Agent Server* (UAS)

      UAS merupakan aplikasi *server* yang memberikan *user* jika menerima *request* dan memberikan respon terhadap *request* tersebut. Respon dapat berupa menerima atau menolak *request*.

2. *Network Server*

   Agar *user* pada jaringan SIP dapat memulai suatu panggilan dan dapat pula dipanggil, maka *user* terlebih dahulu harus melakukan registrasi agar lokasinya dapat diketahui. Regristrasi dapat dilakukan dengan mengirimkan pesan REGISTER ke *server* SIP, ada dua tipe *network server*, yaitu :

   1. *Proxy Server*

      Proxy Server adalah *server* yang menerima *request*, mengolah serta meneruskan *request* yang diterimanya ke *next hop server* setelah mengubah beberapa *header* pada pesan *request*. *Next hop server* dapat berupa *server* SIP atau *server* lainnya dimana *proxy server* tidak perlu tahu. *Proxy server* dapat berfungsi sebagai *client* dan *server* karena *proxy server* dapat memberikan *request* dan respon.

   2. *Redirect Server*

      Komponen ini merupakan *server* yang menerima pesan *request* serta memberikan respon terhadap *request* tersebut yang berisi alamat dari *next hop server*.

2.6.3.2 H.323

H.323 merupakan protokol yang pertama kali ada diantara protokol VoIP yang lain. Protokol ini merupakan protokol yang stabil dan andal. H.323 ini terdiri dari kumpulan beberapa protokol lain yang berfungsi untuk mengatur *session* dan media transfer.

- **Komponen H.323**

Standar H.323 terdiri atas empat komponen yaitu terminal, *gateway*, *gatekeeper*, dan *multipoint control unit* yang jika disatukan dalam jaringan akan memberikan layanan komunikasi multimedia *point to point* dan *multipoint*. Berikut ini adalah penjelasan dari komponen-komponen tersebut :

1. Terminal

   Terminal digunakan untuk berkomunikasi multimedia yang *realtime bidirectional* (dua arah). Terminal H.323 dapat berupa personal komputer atau sebuah peralatan yang menjalankan aplikasi multimedia H.323. Peralatan-peralatan tersebut harus mendukung komunikasi suara (*audio*) dan sebagai tambahan bisa mendukung juga komunikasi data dan vidio.

2. *Gateway*

   Sebuah *gateway* menghubungkan dua buah jaringan yang berbeda. *Gateway* H.323 menghubungkan jaringan H.323 dengan jaringan non-H.323. *Gateway* dapat bertindak sebagai terminal bahkan dengan menggunakan pensinyalan H.245, *gateway* dapat beroperasi sebagai MCU untuk *call* yang sama yang diinisialisasikan secara *point to point*. *Gatekeeper* mengenal apakah suatu terminal adalah *gateway* karena hal ini diaplikasikan ketika terminal/*gateway* melakukan *register* dengan *gatekeeper*.

3. *Gatekeeper*

   Sebuah *gatekeeper* dapat dipertimbangkan sebagai pusat dari jaringan H.323. *Gatekeeper* menyediakan pelayanan-pelayanan yang penting seperti *call routing*, pengalamatan, otorisasi dan otentifikasi dari terminal dan *gateway*, manajemen *bandwidth*, *accounting*, pembiayaan dan rekening. *Gatekeeper* juga bisa menyediakan layanan *call-routing*. *Gatekeeper* merupakan komponen logika H.323 tetapi diaplikasikan sebagai bagian dari *gateway* atau MCU.

4. *Multipoint Control Unit* (MCU)

   Multipoint Control Unit (MCU) memberikan dukungan untuk konferensi tiga atau lebih terminal H.323. semua terminal yang akan berpartisipasi dalam konferensi melakukan koneksi terlebih dahulu dengan *Multipoint Control Unit*. *Multipoint Control Unit* mengatur konferensi *resource*, negosiasi antar terminal untuk tujuan penentuan *audio* atau vidio coder/decoder (CODEC) yang digunakan, dan memungkinkan menangani media *stream*. *Gatekeeper*, *gateway* dan *Multipoint Control Unit* merupakan komponen standar H.323 yang secara logika terpisah tetapi dapat diimplementasikan sebagai *single physical device*.

2.6.4 *Codec*

Codec merupakan kependekan dari *Compression/Decompression*. *Codec* merupakan teknologi yang memaketkan data *voice* ke dalam format lain dengan perhitungan matematis tertentu, sehingga menjadi lebih teratur dan mudah dipaketkan. *Codec* bertujuan untuk mengurangi penggunaan *bandwith* di dalam transmisi sinyal pada setiap pemanggilan tanpa mengorbankan kualitas suara.

International Telecommunication Union – Telecommunication (ITU-T) membuat beberapa standar untuk *voice coding* yang direkomendasikan untuk implementasi VoIP. Berikut ini standar yang digunakan dalam VoIP:

Tabel 2.1 Teknik kompresi standar ITU-T

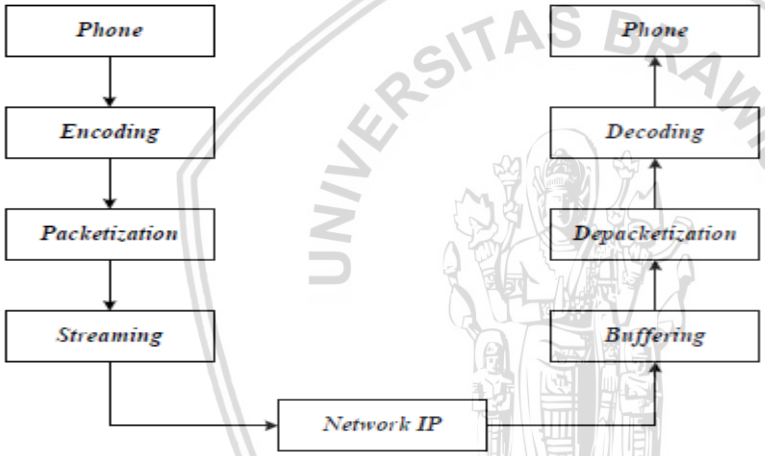
Teknik Kompresi	**Bit Rate (kbps)**	**Sample/Frame Size (ms)**	**Ukuran Voice *Payload* (Bytes)**	**Delay Codec**	**MOS**
G.711	64	20	160	0.75 ms	4.1
G.726	32	20	80	1 ms	3.85
G.728	16	2.5	60	3-5 ms	3.61
G.729	8	10	20	10 ms	3.92
G.723. 1a	6.3	30	24	30 ms	3.9
G.723. 1b	5.3	30	20	30 ms	3.65

(Sumber: http://www.newport-network.com/VoIP -*Bandwidth*.pdf)

2.7 Cara Kerja VoIP

Cara kerja VoIP yaitu dengan merubah data yang berupa sinyal analog kemudian dikonversikan dengan ADC (*Analog to Digital Converter*) menjadi bentuk data digital. Selain diubah menjadi format digital, data suara juga mengalami proses kompresi agar penggunaan *bandwith* di dalam proses transmisi dapat dikurangi. Data digital yang telah dikompresi kemudian dienkapsulasi kedalam paket-paket sehingga dapat dengan mudah ditransmisikan melalui IP.

Setelah itu, data digital akan ditransmisikan ke tujuan. Setelah sampai, data digital akan didekapsulasi dan dikonversi kembali menjadi sinyal analog dengan DAC (*Digital to Analog Converter*) sehingga dapat diterima sesuai dengan data sinyal yang ditransmisikan. Berikut ini arsitektur jaringan VoIP secara umum ditunjukkan pada Gambar 2.5.



Gambar 2.5 Desain arsitektur jaringan VoIP
(Sumber: Ebna Masum, 2011)

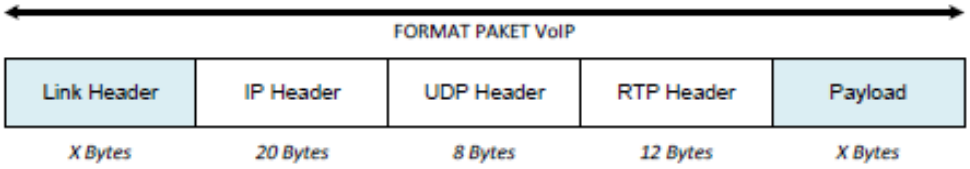
2.8 Format Paket VoIP

Format paket VoIP terdiri atas dua bagian, yakni *header* dan *payload* (beban). *Header* terdiri atas IP *header*, *Real-time Transport Protocol*, *User Datagram Protocol* (UDP), dan *link header*.

IP *header* bertugas menyimpan informasi *routing* untuk mengirimkan paket-paket ke tujuan. Pada tiap *header* IP disertakan tipe layanan atau *Type of service* (ToS) yang memungkinkan paket tertentu seperti paket suara yang *real time*.

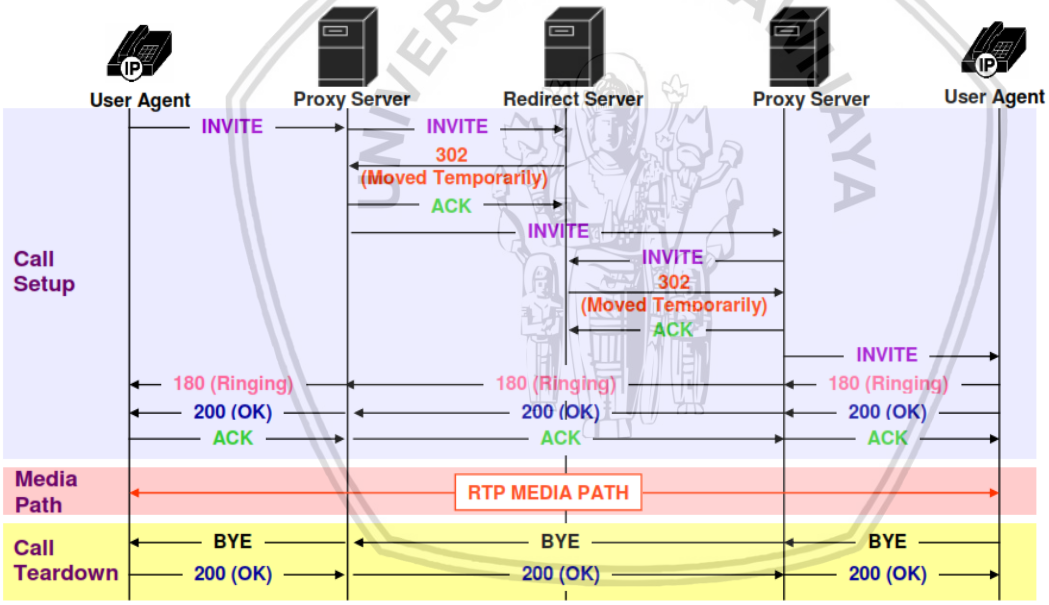
UDP *header* memiliki ciri-ciri yaitu tidak menjamin paket akan mencapai tujuan sehingga UDP cocok digunakan pada aplikasi *voice real time* yang sangat peka terhadap *delay* dan *latency*.

RTP *header* adalah *header* yang berfungsi untuk melakukan *framing* dan segmentasi data *real time*. Seperti UDP, RTP juga tidak mendukung paket untuk mencapai ke tujuan. RTP menggunakan protokol kendali yang disebut RTCP (*Real-time Transport Control Protocol*) yang mengendalikan QoS dan sinkronisasi media *stream* yang berbeda. Untuk *link header*, besarnya sangat tergantung pada media yang digunakan. Format paket VoIP dapat dilihat pada Gambar 2.6 berikut :



Gambar 2.6 Format paket VoIP
(Sumber: www.cisco.com)

2.9 Sistem Panggilan VoIP



Gambar 2.7 Proses komunikasi VoIP pada protokol SIP
(Sumber: www.cisco.com)

Berikut ini merupakan penjelasan tahapan proses komunikasi VoIP pada protokol SIP :

1. Proses *regristasi user*, pengenalan *user* dan penentuan lokasi *user*. Kemudian *User Agent* A memulai panggilan ke *User Agent* B dengan mengirimkan sinyal *Invite* melalui *Proxy Server*, lalu *Proxy Server* meneruskan *Invite* ke *Location/Redirect Server* yang kemudian diteruskan ke *Proxy Server* di sisi *User Agent* B.
2. Lalu *User Agent* B mengirim sinyal balasan ke *User Agent* A dan memberikan pesan menerima atau menolak panggilan.

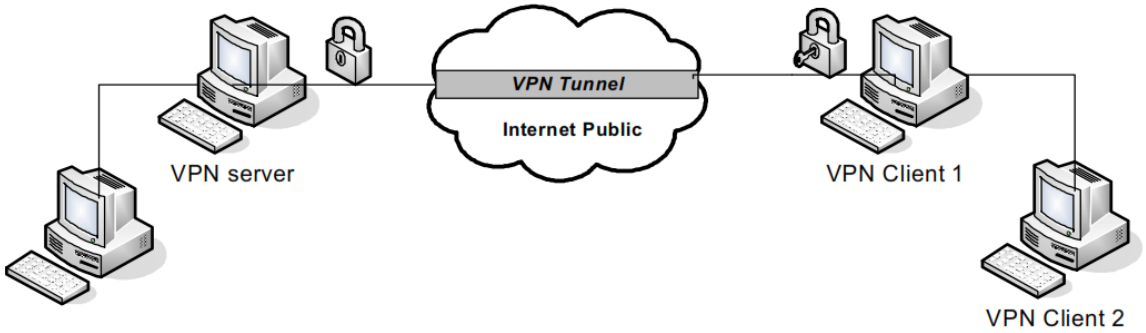
3. *User Agent* A menerima sinyal balasan ACK dari *User Agent* B berupa pesan OK.
4. Kanal suara dua arah terbentuk pada protokol transport *real-time* (RTP) dan percakapan dilakukan antara *User Agent* A dan *User Agent* B.
5. Setelah proses komunikasi selesai, *User Agent* B keluar dan memberikan sinyal berupa pesan BYE ke *User Agent* A.
6. *User Agent* A menerima pesan BYE dan memberikan sinyal berisi pesan OK ke *User Agent* B untuk mengakhiri proses komunikasi.

2.10 *Virtual Private Network* (VPN)

Menurut *Internet Engineering Task Force* (IETF), VPN merupakan suatu bentuk private internet yang melalui *public network* (internet), dengan menekankan pada keamanan data dan akses global melalui internet. Hubungan ini dibangun melalui suatu *tunnel* (terowongan) virtual antara 2 *node*. VPN disebut dengan *Virtual Network* karena menggunakan jaringan internet sebagai media perantaranya atau koneksinya bukan secara langsung. Dan disebut *Private Network* karena VPN sifatnya privat maksudnya hanya orang tertentu saja yang dapat mengaksesnya.

2.11 Prinsip Kerja VPN

Dalam proses *tunneling* VPN, *frame* data yang akan dikirimkan dibungkus (enkapsulasi) terlebih dahulu. *Frame* tersebut diberikan *header* tambahan yang berupa informasi *routing*. Pada VPN juga berlangsung proses *autentifikasi* dan *enkripsi*. *Autentifikasi* adalah suatu proses untuk memastikan bahwa kedua ujung koneksinya merupakan benar *user* yang diinginkan. Sedangkan *enkripsi* merupakan proses perubahan pesan asli (*plain text*) menjadi pesan yang tidak dapat dibaca (*chipper text*) dengan menggunakan kunci.



Gambar 2.8 Prinsip kerja VPN
(**Sumber:** Markus Feilner, 2006)

Dari Gambar 2.8 prinsip kerja VPN dengan protokol PPTP dapat dijelaskan sebagai berikut:

- VPN membutuhkan sebuah *server* yang berfungsi sebagai penghubung antar PC, *server* VPN ini bisa berupa komputer dengan aplikasi VPN *server* atau sebuah *router*, misalnya MikroTik RB 750.
- Untuk memulai sebuah koneksi, komputer dengan aplikasi VPN *client* mengontak *server* VPN, VPN *server* kemudian memverifikasi *username* dan *password*. Apabila berhasil maka VPN *server* memberikan IP *Address* baru pada komputer *client* dan selanjutnya sebuah koneksi / *tunnel* akan terbentuk.
- Untuk selanjutnya komputer *client* bisa digunakan untuk mengakses berbagai *resource* (komputer atau LAN) yang berada dibelakang VPN *server* misalnya melakukan transfer data, *print* dokumen, *browsing* dengan *gateway* yang diberikan dari VPN *server*, melakukan *remote desktop* dan lain sebagainya.

2.12 Keunggulan dan Kelemahan VPN

2.12.1 Keunggulan VPN

Beberapa keunggulan dari teknologi VPN diantaranya adalah :

- *Remote Access*, dengan VPN kita dapat mengakses komputer atau jaringan kantor, dari mana saja selama terhubung ke internet.
- Keamanan, dengan koneksi VPN kita bisa berselancar dengan aman ketika menggunakan akses internet publik seperti *hotspot* atau internet cafe.
- Menghemat biaya *setup* jaringan, VPN dapat digunakan sebagai teknologi alternatif untuk menghubungkan jaringan lokal yang luas dengan biaya yang relatif kecil, karena transmisi data teknologi VPN menggunakan media jaringan publik yang sudah ada tanpa perlu membangun jaringan pribadi.

2.12.2 Kelemahan VPN

Beberapa kekurangan dari VPN diantaranya adalah :

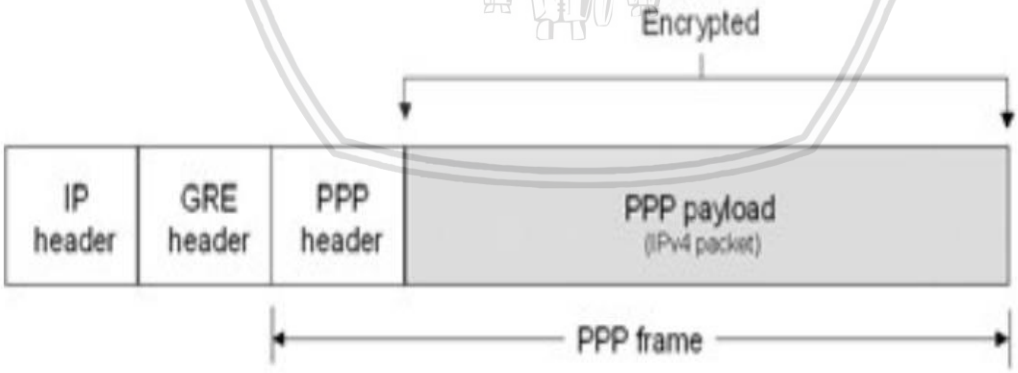
- Koneksi internet (jaringan publik) yang tidak bisa di prediksi. Hal ini disebabkan karena jaringan vpn hanya menumpang koneksi pada jaringan pihak lain sehingga otomatis tidak mempunyai kontrol terhadap jaringan tersebut.

- Faktor penggunaan jaringan publik memerlukan perhatian yang lebih dalam hal keamanan untuk mencegah terjadinya hal-hal yang tidak diinginkan seperti penyadapan, hacking dan tindakan *cyber crime* pada jaringan VPN.

2.13 Protokol VPN

Untuk bisa saling berhubungan antar *user* pada komunikasi VPN diperlukan protokol untuk menghubungkan komunikasi tersebut. Salah satu protokol yang biasa digunakan adalah protokol PPTP (*Point-to-Point Tunneling Protocol*).

PPTP merupakan protokol jaringan yang memungkinkan pengamanan transfer data dari *remote client* ke *server* dengan membuat sebuah VPN melalui TCP/IP. Teknologi jaringan PPTP merupakan pengembangan dari *remote access Point-to-Point protocol* yang dikeluarkan oleh *Internet Engineering Task Force* (IETF). PPTP merupakan protokol jaringan yang merubah paket PPP menjadi IP *datagram* agar dapat ditransmisikan melalui internet. PPTP juga dapat digunakan pada jaringan *private* LAN-*to*-LAN. Fasilitas utama dari penggunaan PPTP adalah dapat digunakannya *public-switched telephone network* (PSTNs) untuk membangun VPN. Pembangunan PPTP yang mudah dan berbiaya murah untuk digunakan secara luas, menjadi solusi untuk *remote users* dan *mobile users* karena PPTP memberikan keamanan dan enkripsi komunikasi melalui PSTN ataupun internet. Gambar 2.9 adalah paket data pada PPTP.



Gambar 2.9 Format paket PPTP
(**Sumber:** technet.microsoft.com)

Pada proses enkapsulasi, PPTP mengenkapsulasi PPP *frames* pada IP *datagram* untuk ditransmisikan pada jaringan. PPTP juga menggunakan koneksi TCP untuk mengelola *tunnel* dan GRE (*Generic Routing Encapsulation*).

Proses *tunneling* pada PPTP terjadi dengan cara membungkus paket informasi untuk kemudian ditransmisikan melalui jaringan internet. Pada proses ini PPTP menggunakan koneksi TCP yang dikenal sebagai PPTP *control connection* untuk menciptakan, merawat dan mengakhiri *tunnel* serta *Generic Routing Encapsulation* (GRE).

Dalam hal enkripsi, PPTP menggunakan mekanisme otentikasi yang sama dengan PPP seperti *Extensible Authentication Protocol* (EAP), *Challange Handshake Protokol* (CHAP), *Shiva Password Authentication Protocol* (SPAP) dan *Password Authentication Protocol* (PAP).

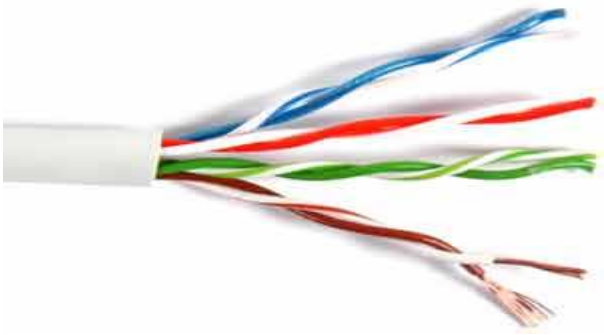
2.14 Media Transmisi

Media transmisi adalah media yang menghubungkan antara pengirim dan penerima informasi (data). Media transmisi dapat dibagi menjadi 2 jenis yaitu media *wired* dan media *wireless*. Percobaan pada skripsi ini menggunakan *Twisted Pair Cable* untuk media *wired* dan gelombang radio untuk media *wireless*-nya.

2.14.1 *Wired*

2.14.1.1 *Unshielded Twisted Pair* (UTP)

Kabel UTP adalah sebuah jenis kabel jaringan yang menggunakan bahan dasar tembaga yang tidak dilengkapi dengan *shield internal*. UTP merupakan jenis kabel yang paling umum yang sering digunakan di dalam jaringan lokal (LAN), karena harganya yang rendah, fleksibel dan kinerja yang ditunjukkannya relatif bagus. Dalam kabel UTP terdapat insulasi satu lapis yang melindungi kabel dari ketegangan fisik atau kerusakan tapi tidak seperti kabel *Shielded Twisted Pair* (STP), insulasi tersebut tidak melindungi kabel dari interferensi elektromagnetik. Kabel UTP biasanya menggunakan konektor *Registered Jack* 45 (RJ-45) seperti pada Gambar 2.10.



Gambar 2.10 Kabel UTP
(**Sumber:** http://www.materitkj.com/2015/12/kabel-unshielded-twisted-pair-utp.html)

Kabel UTP dapat dikelompokkan sebagai berikut :

- Kategori 1 – Digunakan untuk komunikasi telepon. Tidak cocok untuk transmisi data.
- Kategori 2 – Mampu mengirimkan data pada kecepatan hingga 4 *megabyte* per detik (*Mbps*).
- Kategori 3 – Digunakan untuk jaringan 10Base-T. Dapat mengirimkan data pada kecepatan hingga 10 Mbps.
- Kategori 4 – Digunakan untuk jaringan *Token Ring*. Dapat mengirimkan data pada kecepatan hingga 16 Mbps.
- Kategori 5 – Dapat mengirimkan data pada kecepatan hingga 100 *Mbps*.
- Kategori 5e – Digunakan dalam jaringan yang berjalan pada kecepatan sampai dengan 1000 Mbps.
- Kategori 6 – Terdiri dari empat pasang dari 24 *American Wire Gauge* (AWG) kabel tembaga.

2.14.1.2 *Shielded Twisted-Pair* (STP)

Shielded twisted-Pair adalah kabel tembaga yang memiliki pembungkus pada masing-masing pasangan kabelnya. Perlindungan tersebut terdapat pada setiap pasang kabelnya yang dilindungi timah dan setiap pasangan kabel tersebut masing-masing dilapisi dengan pelindung. Kabel ini sama dengan UTP, perbedaannya hanya dilapisan pelindungnya, lapisan pelindung tersebut berfungsi untuk melindungi interferensi gelombang elektromagnetik baik dalam maupun diluar.

Gambar 2.11 Kabel STP
(**Sumber:** https://jarkomtutorial.wordpress.com/mengenal -kabel-stp-shielded-twistedpair/)

2.14.2 *Wireless*

Local Area Network (LAN) adalah jaringan dari gabungan beberapa komputer yang tersambung melalui kabel UTP. Seiring dengan kemajuan teknologi serta kebutuhan untuk akses jaringn bergerak, maka munculah *Wireless Local Area Network* (WLAN) dimana antar komputer terhubung dengan teknologi *Radio Frekuency* (RF).

2.14.2.1 Standarisasi *Wireless*

Berikut adalah 4 standarisasi WLAN yang digunakan saat ini :

1. **802.11a**

   IEEE 802.11a merupakan standart jaringan *wireless* yang bekerja pada frekuensi 5 GHz dengan kecepatan transfer datanya mencapai 58 Mbps. Standar ini bisa menjangkau jarak maksimum 150 kaki (45.7 m).
2. **802.11b**

   IEEE 802.11b merupakan standart jaringan *wireless* yang masih menggunakan frekuensi 2.4 GHz dengan kecepatan trasfer datanya mencapai 11 Mbps dan jangkau sinyal sampai dengan 300 kaki (91 m).
3. **802.11g**

   IEEE 802.11g merupakan standart jaringan *wireless* yang merupakan gabungan dari standart 802.11b yang menggunakan frekuensi 2.4 GHz namun kecepatan transfer datanya bisa mencapai 54 Mbps. Standar 802.11g memiliki jangkauan maksimum 300 kaki (91 m).
4. **802.11n**

   IEEE 802.11n yaitu standart jaringan *wireless* baru yang bekerja pada frekuensi 2.4 Ghz atau 5 GHz. Standar ini memiliki kecepatan transfer data mencapai 540 Mbps. Jarak jangkauan maksimum bisa mencapai 984 kaki (250 m).

2.15 Parameter *Quality of Service* (QoS)

Quality of Service (QoS) menurut Onno W. Purbo adalah kemampuan suatu jaringan untuk menyediakan layanan yang lebih baik pada trafik data tertentu pada berbagai jenis *patform* teknologi.

2.15.1 Paket Data VoIP

Paket data adalah satuan informasi dasar yang dapat ditransmisikan di atas jaringan atau melalui saluran komunikasi digital. Paket data (*payload*) pada saat berkomunikasi suara tergantung pada *codec* yang digunakan. Untuk menghitung *payload* yang dibutuhkan, maka sangat penting untuk mengetahui model susunan protokol IP. Untuk VoIP, protokol yang terkait diantaranya : RTP, UDP, IP dan *Network Interface* (seperti *ethernet* dan *tokenring*). Untuk mencari nilai *payload* maka digunakan persamaan:

$$P_{LA} = B_{codec} \times \ frame\ rate \tag{2-1}$$

setelah diketahui besar *payload* paket data pada audio, maka banyaknya bit yang terisi dalam paket VoIP dapat dihitung dengan persamaan :

$$P_{A-size} = header_{IP/UDP/RTP} + P_{LA} \tag{2-2}$$

Sedangkan jumlah paket *audio* yang dihasilkan tiap detik dihitung dengan persamaan:

$$P_A = \frac{B_{codec}}{P_{LA}} \tag{2-3}$$

Sehingga *bandwidth audio* dapat dihitung dengan persamaan :

$$B_A = P_{A-size} \times \ P_A \tag{2-4}$$

2.15.2 Perhitungan SNR

Signal to Noise Ratio (SNR) merupakan perbandingan antara sinyal dengan *noise* yang terjadi pada sistem. Besarnya pengaruh redaman sinyal terhadap sinyal yang ditransmisikan dapat dinyatakan dengan perbandingan antara sinyal dengan *noise* (SNR) yang dinyatakan dalam persamaan berikut [E. Glatz, 1999] :

$$SNR_{(dB)} = P_r(dBm) - N_o(dBm) \tag{2-5}$$

dengan :

SNR = *signal to noise ratio* (dB)

P_r = daya yang diterima (dBm)

N_o = daya *noise* saluran transmisi (dBm)

Daya yang diterima oleh penerima sangat dipengaruhi oleh propagasi sinyal dari pemancar ke penerima.

2.15.3 Perhitungan *Bit rate*

Perhitungan *bit rate* untuk modulasi QPSK menggunakan persamaan berikut :

$$bit\ rate = N_{used} \times \ bm \ \times \frac{c_r}{T_s} \tag{2-6}$$

Dimana :

N_{used} = jumlah *subcarrier*

bm = jumlah *bit* yang digunakan pada modulasi

Cr = *coding rate*

2.15.4 Perhitungan BER

BER (*bit error rate*) atau dengan sebutan lain probabilitas *error bit* merupakan nilai ukur kualitas sinyal yang diterima untuk sistem transmisi data digital. BER juga dapat didefinisikan sebagai perbandingan jumlah *bit error* terhadap total bit yang diterima. Nilai BER untuk modulasi QPSK dapat dihitung dengan persamaan (Andrea Goldsmith, 2005:167):

$$P_s = erfc\left(\sqrt{\frac{E_s}{2N_0}}\right) \tag{2-7}$$

$$P_b = \frac{1}{2} erfc\left(\sqrt{\frac{E_b}{N_0}}\right) \tag{2-8}$$

Dengan :

P_b = kemungkinan untuk terjadinya bit *error* (dB)

Ps = kemungkinan untuk terjadinya simbol *error* (dB)

E_s = energi per *symbol*

E_b = energi per bit (dB)

N_0 = *Noise power spectral density* (dB)

Eb/No adalah suatu parameter yang berhubungan dengan SNR yang biasanya digunakan untuk menentukan laju data digital dan sebagai ukuran mutu standar untuk kinerja sistem komunikasi digital. Hubungan SNR dengan Eb/No ditunjukkan dalam persamaan:

$$\frac{Eb}{No}(dB) = \frac{S}{N} - 10\log\frac{B}{R} \tag{2-9}$$

Dengan :

$\frac{S}{N}$ = SNR (dB)

B = *bandwidth* (Mbps)

R = *bit rate* (Mbps)

2.15.5 *Delay end to end*

Delay end to end adalah waktu total yang dibutuhkan untuk mengirimkan paket data dari sumber sampai ke tujuan. *Delay end-to-end* pada jaringan IP merupakan penjumlahan *delay-delay* yang terjadi dalam perjalanan paket dari sumber ke tujuan. Pada aplikasi VoIP yang bersifat *full duplex*, maka *delay* dihitung dari sumber ke penerima sampai ke sumber lagi.

Delay end to end sangat mempengaruhi kualitas layanan suara, karena pada dasarnya suara memiliki karakteristik "*timing*". Urutan pengucapan tiap suku kata yang ditransmisikan harus sampai ke sisi penerima dengan urutan yang sama pula sehingga dapat terdengar dengan baik secara *real-time.* TIPHON membagi karakteristik waktu tunda seperti ditunjukkan pada Tabel 2.2.

Tabel 2.2 Indeks Delay end to end

Kategori *Delay*	*Delay* (ms)
Sangat Bagus	0 – 150
Bagus	150 – 300
Sedang	300 – 450
Jelek	>450

(Sumber: TIPHON)

Delay end-to-end dapat dituliskan sebagai berikut:

$$t_{end-to-end} = t_{codec} + t_{jaringan} \quad (2\text{-}10)$$

Dengan :

t_{codec} = *delay* codec (ms)

$t_{jaringan}$ = *delay jaringan* (ms)

- *Delay codec*

Delay ini tediri dari waktu untuk mengakumulasi sampel suara ke dalam *frame* suara, waktu untuk mengkompresi paket suara, waktu untuk memuati *frame* suara ke dalam paket dan mentransfer paket tersebut ke jaringan transport. *Delay* codec dapat disebut juga waktu yang dibutuhkan untuk mengkompresi satu blok sample PCM. *Delay codec* pada aplikasi VoIP dapat dihitung dengan persamaan sebagai berikut:

$$t_{CODEC}(ms) = t_{encoding}(ms) + t_{decoding}(ms) \quad (2\text{-}11)$$

Dengan :

t_{CODEC} *(ms)* = *delay codec* aplikasi VoIP (ms)

- *Delay* pada jaringan

Merupakan besarnya *delay* yang diperlukan untuk mengirimkan data dari sisi VoIP *client* ke VoIP *client* tujuan. *Delay* jaringan dapat dihitung dengan persamaan sebagai berikut:

$$t_{jaringan} = t_{proces} + t_{pros} + t_{trans} + t_w \quad (2\text{-}12)$$

Dengan:

$t_{jaringan}$ = *delay* total pada jaringan (ms)

t_{proces} = *delay* proses (ms)

t_{pros} = *delay* propagasi (ms)

t_{trans} = *delay* transmisi (ms)

t_w = *delay* antrian (ms)

- *Delay* proses

Delay proses adalah waktu yang dibutuhkan untuk memproses paket data dan untuk menentukan ke mana data tersebut akan diteruskan. *Delay* proses berupa *delay* enkapsulasi dan *delay* dekapsulasi.

$$W_{segmen} = Header_{RTP} + Header_{UDP} \quad (2\text{-}13)$$

Dengan :

W_{segmen} = panjang segmen pada layer 3 (byte)

$Header_{RTP}$ = panjang *header* RTP (12 byte)

$Header_{UDP}$ = panjang *header* UDP (8 byte)

Dari *layer* 3 atau *layer* transport, *segmen* kemudian dikirim ke *layer* 2 atau *layer network* untuk dienkapsulasi menjadi *datagram* IP. Apabila panjang *segmen* pada *layer* diatasnya melebihi MTU IP yaitu 1500 *byte*, maka *segmen* perlu untuk difragmentasi sebelum dienkapsulasi. Kemudian *datagram* IP dienkapsulasi dengan *header* IP, sehingga panjang *datagram* IP sebagai berikut.

$$W_{frame} = W_{datagram} + Header_{IP} \quad (2\text{-}14)$$

Dengan :

W_{frame} = panjang *segmen* TCP (byte)

$W_{datagram}$ = panjang *datagram* IP (byte)

$Header_{IP}$ = panjang *header* IP (20 byte)

Kemudian *datagram* IP dienkapsulasi dengan *header* pada *layer* 1 pada skripsi ini menggunakan e*thernet* sebagai *layer* pada *datalink*.

$$W_{frame} = W_{datagram} + Header_{ethernet} \quad (2\text{-}15)$$

Dengan:

W_{frame} = panjang *frame* Ethernet (byte)

W_{frame} = panjang *datagram* IP (byte)

$Header_{ethernet}$ = panjang *header* Ethernet (18 byte)

Sedangkan *delay* enkapsulasi adalah:

$$t_{enc} = \frac{W_{frame}}{C_{pros}} \times 8 \quad (2\text{-}16)$$

Dengan :

t_{enc} = *delay* enkapsulasi (ms)

W_{frame} = panjang frame Ethernet (byte)

C_{pros} = kecepatan pemrosesan data (Mbps)

Sedangkan *delay* dekapsulasi dirumuskan :

$$t_{enc} = \frac{W_{frame}}{C_{pros}} \times 8 \quad (2\text{-}17)$$

Dengan :

t_{doc} = *delay* dekapsulasi (ms)

W_{frame} = panjang *frame* Ethernet (byte)

C_{pros} = kecepatan pemrosesan data (Mbps)

Sehingga *delay* proses dapat dituliskan sebagai berikut

$$t_{proc} = t_{enc} + t_{doc} \quad (2\text{-}18)$$

Dengan :

t_{proc} = *delay* proses (ms)

t_{enc} = *delay* enkapsulasi (ms)

t_{doc} = *delay* dekapsulasi (ms)

- *Delay* propagasi

Delay propagasi adalah jumlah waktu yang dibutuhkan oleh data untuk berpropagasi pada media transmisi. *Delay* propagasi gelombang radio dapat ditulis dengan persamaan :

$$t_{prop} = \frac{L_k}{V_{prop}} \quad (2\text{-}19)$$

Dengan :

t_{prop} = *delay* propagasi (ms)

L_k = panjang saluran (meter)

V_{prop} = kecepatan propagasi (Mbps)

- *Delay* transmisi

Delay transmisi adalah waktu yang dibutuhkan untuk mentransmisikan paket data. *Delay* transmisi dirumuskan pada persamaan:

$$t_{trans} = \frac{W_{total-frame}}{C_{trans}} \times 8\ byte \quad (2\text{-}20)$$

Dengan :

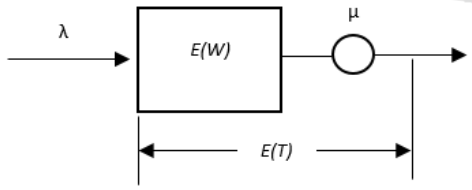
t_{trans} = *delay* transmisi (ms)

$W_{total-frame}$ = panjang total *frame* yang dikirimkan (byte)

C_{trans} = kecepatan transmisi kanal (Mbps)

- *Delay* antrian

Delay antrian adalah waktu yang dibutuhkan data selama berada dalam antrian untuk ditransmisikan. *Delay* ini disebabkan oleh waktu proses yang diperlukan oleh *router* dalam menangani paket sepanjang jaringan. Pada analisis aplikasi VoIP ini, model antrian yang digunakan dalam analisis adalah model antrian M/M/1. Model antrian M/M/1 berarti proses kedatangan paket data umumnya acak dan waktu pelayanan adalah distribusi eksponensial. Disiplin antrian yang digunakan adalah FIFO (First In First Out). Gambar 2.12 menunjukkan model M/M/1.



Gambar 2.12 Model antrian M/M/1
(**Sumber:** Schwartz, 1987)

Jika kapasitas *link* adalah C (Mbps) dan panjang paket data adalah W (*byte*), maka besarnya kapasitas *link* akan menentukan kecepatan pelayanan (μ) yaitu

$$\mu = \frac{C}{W} \quad (2\text{-}21)$$

Utilitas link didefinisikan sebagai perbandingan antara beban sistem dengan kapasitas sistem. Jika λ adalah laju kedatangan data (paket/sec) dan μ adalah kecepatan pelayanan suatu sistem (paket/sec), maka :

$$\lambda_w = \mu \times \rho \quad (2\text{-}22)$$

Dengan menerapkan teorema Little pada model antrian M/M/1 seperti terlihat dalam Gambar 2.12 didapatkan total waktu tunggu suatu paket dalam antrian M/M/1.

$$t_w = \frac{\rho/\mu}{(1-\rho)} \quad (2\text{-}23)$$

Dengan :

t_w = total waktu tunggu suatu paket dalam antrian (ms)
μ = kecepatan nilai waktu pelayanan (paket/s)
ρ = utilisasi link

2.15.6 *Packet Loss*

Packet Loss merupakan banyaknya probabilitas paket yang hilang pada sisi penerima. Probabilitas *packet loss* layanan VoIP pada suatu jaringan ditentukan berdasarkan pada probabilitas *packet loss* pada jaringan tersebut serta probabilitas *packet loss* layanan VoIP yang berbasis protokol UDP/RTP/IP. Kategori *packet loss* ditunjukkan pada Tabel 2.3.

Tabel 2.3 Indeks Packet Loss

Kategori *Packet Loss*	*Packet Loss* (%)
Sangat Bagus	0
Bagus	0 – 3
Sedang	3 – 15
Jelek	15 – 25

(Sumber: TIPHON)

Persamaan untuk menghitung *packet loss* adalah :

$$\rho_{tot} = 1 - [(1 - \rho_{VoIP})(1 - \rho_{jaringan})] \quad (2\text{-}24)$$

Dengan:

ρ_{tot} = *packet loss* total
ρ_{voip} = probabilitas *packet loss* pada VoIP
$\rho_{jaringan}$ = probabilitas *packet loss* jaringan VoIP

2.15.7 *Throughput*

Throughput merupakan salah satu parameter yang menunjukkan kinerja dari suatu komunikasi data, yaitu menunjukkan jumlah data yang diterima dengan benar pada penerima setelah melewati media transmisi pada data *link* layer dari *client* ke *client*. Kategori throughput diperlihatkan pada Tabel 2.4

Tabel 2.4 Indeks Throughput

Kategori *Throughput*	***Throughput* (%)**
Sangat Bagus	100 - 75
Bagus	75 - 50
Sedang	50 - 25
Jelek	<25

(Sumber: TIPHON)

Persamaan untuk menghitung *throughput* adalah :

$$\lambda = \frac{1}{t_v} = \frac{(1-\rho)}{t_1[1+(\alpha-1)\rho]} \quad (2\text{-}25)$$

Dimana :

$$\alpha = (1 + \frac{t_{out}}{t_1})$$

$$t_{out} = 2t_{prop} + 2t_1$$

$$\alpha = \left(1 + \frac{2t_{prop}+2t_1}{t_1}\right) = \left(3 + \frac{2t_{prop}}{t_1}\right) \quad (2\text{-}26)$$

Keterangan :

λ = *Throughput* (kbps)

ρ = probabilitas paket loss yang diterima

α = kontanta propagasi

t_1 = delay transmisi (ms)

BAB III
METODOLOGI

3.1 Umum

Sesuai dengan tujuan penulisan skripsi ini, yaitu untuk menganalisa perubahan performansi dan tingkat keamanan dari layanan VoIP sebelum dan sesudah diamankan menggunakan VPN pada media *wired* dan *wireless*. Parameter-parameter yang dianalisis meliputi *delay end to end*, *packet loss,* dan *throughput*. Langkah-langkah yang digunakan dalam penelitian ini antara lain:

3.2 Identifikasi Data

Untuk data primer, proses pengujian dan pengamatan data ditinjau dari bagaimana performansi VoIP pada 2 tipe jaringan yaitu jaringan tanpa menggunakan VPN dan jaringan menggunakan VPN . Kemudian dari 2 tipe jaringan ini, masing-masing dibedakan lagi menjadi 2 menurut media transmisi yang digunakan, yaitu jaringan dengan media *wired* dan jaringan dengan media *wireless*. Sedangkan untuk data sekunder yaitu berupa pemahaman konsep yang terkait dengan implementasi VoIP dan VPN pada media *wired* dan *wireless*. Adapun paremeter kualitas jaringan yang akan diamati yaitu :

a. *Delay end to end*
b. *Packet Loss*
c. *Throughput*

3.3 Pengambilan Data

Pengambilan data dilakukan untuk memperoleh data-data yang diperlukan dalam menyelesaikan penelitian ini. Pada skripsi ini menggunakan kedua jenis data yang diambil yaitu data primer dan data sekunder. Langkah-langkah untuk mendapatkan data pada penelitian ini adalah sebagai berikut :

3.3.1 Pengambilan Data Primer

Data primer adalah data yang dibuat oleh peneliti dengan tujuan untuk menyelesaikan permasalahan yang sedang ditangani. Parameter yang akan digunakan dalam pengambilan data, yaitu *delay end to end, packet loss,* dan *throughput*. Parameter tersebut digunakan untuk mewakili unjuk kerja dari VoIP dalam melakukan proses pemanggilan. Pengambilan data dilakukan pada 2 media transmisi berbeda, yaitu menggunakan media *wired* dan *wireless*.

Pada media tersebut dilakukan percobaan sebanyak 5 kali pada konfigurasi jaringan tanpa menggunakan VPN dan jaringan yang menggunakan VPN. Pada proses pengambilan data, jarak antara *client* dan *router* maupun *acces point* $\pm$ 10 m.

Dari perbedaan konfigurasi keamanan data dan media transmisi yang digunakan, akan dilakukan pengamatan mengenai kualitas *Voice Offer Internet Protocol* (VoIP) untuk memperoleh informasi terkait dengan kelebihan dan kekurangan dari jaringan tersebut.

3.3.2 Pengambilan Data Sekunder

Data sekunder adalah data yang diperoleh dari studi literatur yang bersumber dari buku referensi, jurnal, skripsi, internet dan forum-forum resmi mengenai VoIP dan VPN. Data sekunder yang digunakan dalam pembahasan skripsi ini antara lain sebagai berikut :

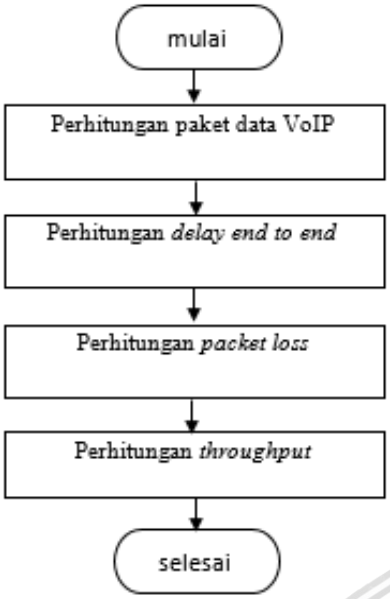
- Konsep dasar dasar konfigurasi layanan dan parameter-parameter VoIP,
- Konsep dasar konfigurasi VPN,
- Konsep dan analisis perhitungan parameter-parameter QoS performansi jaringan.

3.4 Analisis Data

Analisis data bertujuan untuk menentukan kebutuhan data yang diperlukan. Setiap data yang didapat akan dipilah sesuai dengan data yang dibutuhkan.

3.4.1 Diagram Alir metode analisis perhitungan

Sebagai referensi dalam melakukan simulasi perlu diketahui alur analisis secara perhitungan untuk lebih memahami dan mengerti data-data hasil simulasi yang akan diperoleh. Pada perhitungan ini data yang digunakan sebagai acuan yakni bersumber dari data sekunder meliputi jurnal, buku referensi, skripsi, internet dan forum-forum resmi mengenai VoIP dan VPN. Gambar 3.1 merupakan diagram alir perhitungan parameter yang dibutuhkan VoIP.



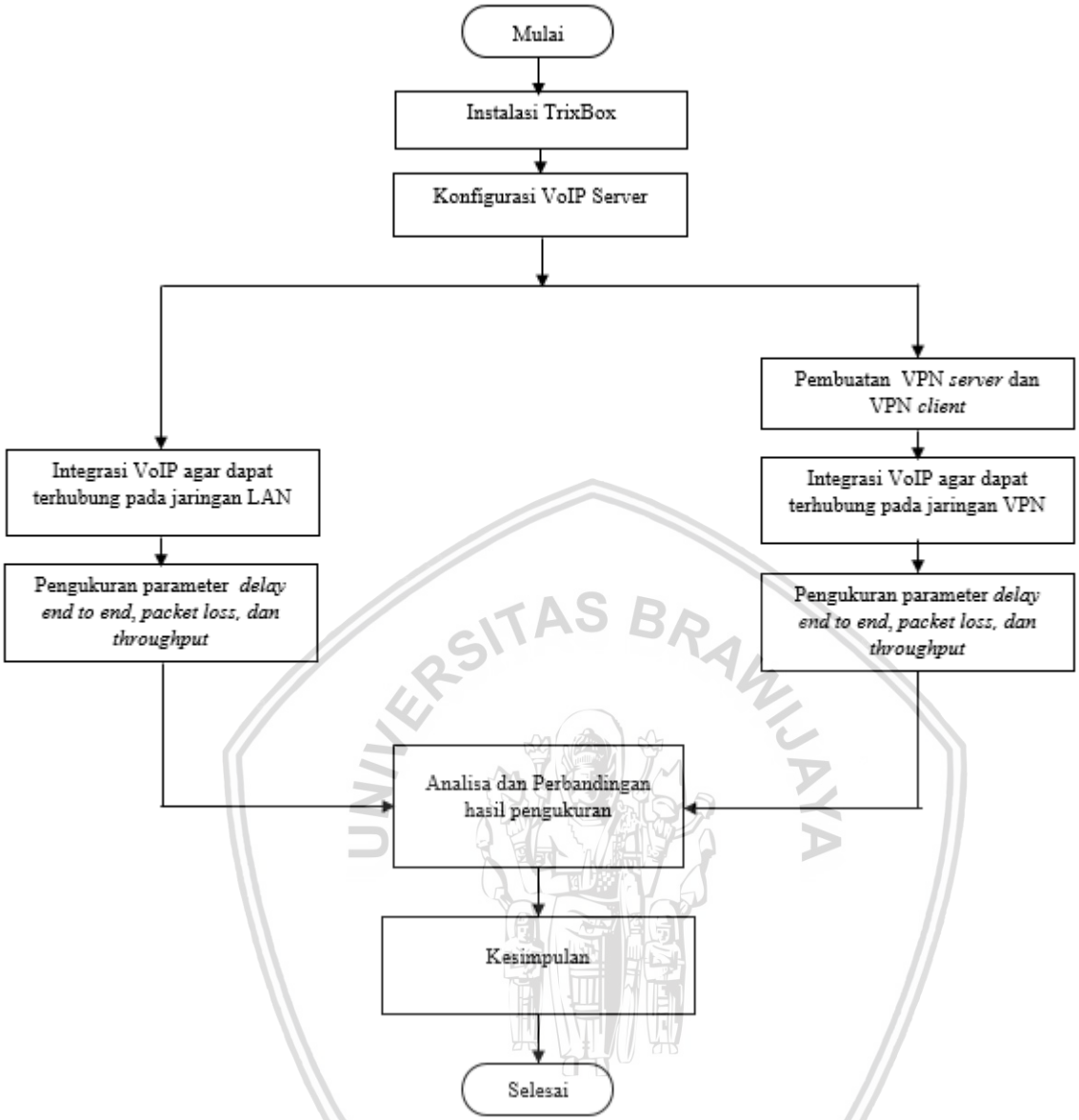
Gambar 3.1 Diagram Alir metode analisis perhitungan
(**Sumber:** Perancangan)

Perhitungan dan analisis data dalam skripsi ini meliputi kinerja berikut:

1. Kebutuhan *payload* pada layanan VoIP dengan mengunakan *audio codec* G.711, kemudian perhitungan kapasitas kanal dari sistem jaringan LAN.
2. Perhitungan *delay end to end* yang di perbolehkan pada sistem agar mampu bekerja secara optimal.
3. Probabilitas *packet loss* yang dihasilkan pada saat pengiriman paket VoIP.
4. *Troughput* yang dihasilkan dari pengiriman paket suara melalui jaringan LAN.

3.4.2 Diagram Alir metode simulasi

Langkah-langkah yang diperlukan dalam proses pengambilan dan pengolahan data primer ditunjukan pada Gambar 3.2.



Gambar 3.2 Diagram alir metode simulasi
(**Sumber:** Perancangan)

Berikut uraian penjelasan berdasarkan Diagram alir diatas :

1. Langkah pertama yang dilakukan untuk mendapatkan data primer adalah menginstal *software* TixBox yang berfungsi sebagai VoIP *server*.
2. Mengkonfigurasi VoIP *server* dengan membuat *user* ID untuk masing-masing *client* VoIP.
3. Untuk jaringan yang menggunakan VPN, diperlukan pembuatan *server* VPN pada *router* mikrotik RB 750 dan *client* VPN pada masing-masing PC *client*. Sedangkan pada jaringan tanpa VPN, *server* VoIP dapat langsung di integrasikan pada jaringan LAN.

4. Pengukuran parameter QoS pada jaringan tanpa VPN dan jaringan VPN meliputi *delay end to end*, *packet loss*, dan *throughput*. Serta, pengukuran dilakukan pada media transmisi *wired* dan *wireless*.
5. Data hasil pengukuran QoS dari masing-masing variasi jaringan dan media transmisi akan dilakukan perbandingan dan analisis.
6. Pengambilan kesimpulan dilakukan dari hasil perbandingan dan analisis QoS.

3.5 Topologi Jaringan

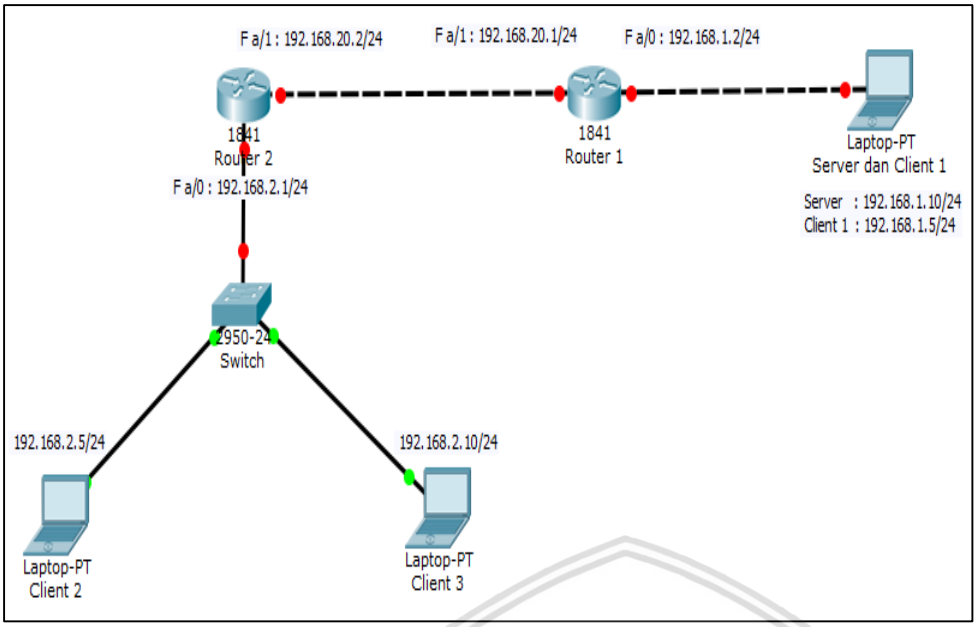
Perancangan jaringan percobaan terdiri dari 2 tipe jaringan yang dibedakan menurut keamanan jaringan, yaitu jaringan tanpa menggunakan VPN dan jaringan menggunakan VPN. Kemudian dari 2 tipe jaringan ini, masing-masing dibedakan lagi menurut media transmisi yang digunakan, yaitu jaringan dengan media *wired* dan jaringan dengan media *wireless*.

Topologi yang digunakan pada jaringan percobaan menggunakan jenis star, karena topologi star memiliki beberapa kelebihan diantaranya :

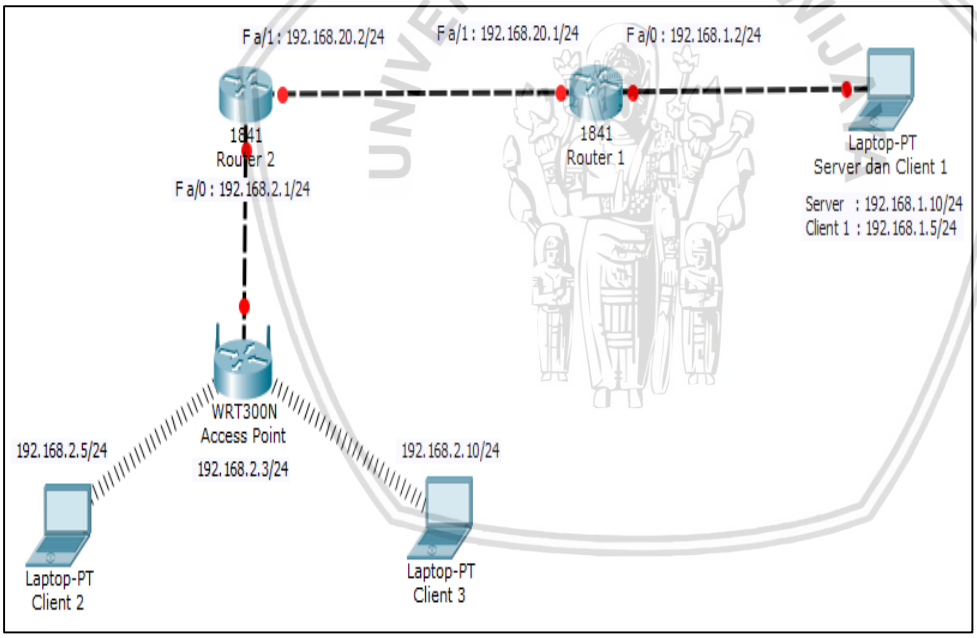
- Bersifat fleksibel.
- Kemudahan dalam deteksi masalah jika terjadi kerusakan pada jaringan.
- Apabila salah satu komputer mengalami masalah, jaringan pada topologi ini tetap berjalan dan tidak mempengaruhi komputer yang lain.

Sebagai perangkat pendukung dalam percobaan akan digunakan PC yang akan bertindak sebagai *server* sekaligus *client*. Alasan penggunaan PC *server* sekaligus *client* adalah sebagai solusi keterbatasan PC yang dimiliki karena keterbatasan alat. PC yang difungsikan sebagai *server* sekaligus *client* menggunakan aplikasi Trixbox pada VMware untuk menjalankan serv*er* VoIPnya dan aplikasi X-Lite untuk *client*-nya.

Selain itu perangkat pendukung lainnya adalah *router* Mikrotik RB750. *Router* Mikrotik RB750 selain digunakan sebagai *router* biasa, juga berfungsi sebagai *server* VPN pada saat VoIP menggunakan pengaman VPN. Semua PC *client* menggunakan aplikasi X-Lite yang berfungsi untuk melakukan sambungan VoIP. Berikut adalah topologi yang digunakan dalam percobaan untuk jaringan *wired* dan *wireless.*



Gambar 3.3 Topologi jaringan *wired*
(**Sumber:** Perancangan)



Gambar 3.4 Topologi jaringan *wireless*
(**Sumber:** Perancangan)

3.6 Konfigurasi Jaringan

3.6.1 Konfigurasi Jaringan *Wired*

Pada bagian ini akan dijelaskan mengenai perancangan konfigurasi jaringan yang menggunakan *wired*. Konfigurasi jaringan terdiri dari jaringan tanpa menggunakan VPN dan jaringan menggunakan VPN.

Gambar 3.3 diatas merupakan gambaran topologi jaringan *tes-bed wired*. PC *server* terhubung secara virtual dengan *client* 1 sehingga dapat melakukan dua peran sekaligus yaitu sebagai *server* dan *client*. Semua perangkat pada konfigurasi *wired* dihubungkan menggunakan kabel UTP. Konfigurasi jaringan yang digunakan pada *tes-bed wired* berbeda-beda. Berikut ini perbedaan konfigurasi setiap jaringan :

- Jaringan VoIP tanpa pengaman VPN

  PC *client* VoIP dihubungkan ke PC *server* melalui *router* yang sudah di atur secara *static* untuk *routing*-nya sesuai dengan alamat IP masing-masing tanpa ada pengaturan pengamanan data.
- Jaringan VoIP menggunakan pengaman VPN

  PC *client* VoIP dihubungkan ke PC *server* melalui *router* yang sudah diatur menggunakan pengaman VPN dengan protokol PPTP (*Point to Point Tunneling Protocol*). *Router* mikrotik RB750 berfungsi sebagai *server* VPN yang akan memberikan nomor IP *tunneling* kepada *client* setelah memasukan *username* dan *password* yang telah terdaftar. Pada sisi *client* VPN, PC melakukan sambungan menggunakan layanan khusus VPN yang telah tersedia di menu *network and sharing centre* pada *windows*. Dengan memasukan *username* dan *password* yang telah terdaftar pada *server* VPN ke dalam layanan sambungan VPN pada *windows*, *client* dapat saling berkomunikasi dengan aman.

3.6.2 Konfigurasi Jaringan *Wireless*

Perancangan konfigurasi dengan media *wireless* ini sama dengan media *wired*. Konfigurasi jaringan terdiri dari jaringan tanpa menggunakan VPN dan jaringan menggunakan VPN.

Gambar 3.4 diatas merupakan gambaran topologi jaringan percobaan *wireless*. PC *server* terhubung secara virtual dengan *client* 1 sehingga dapat melakukan dua peran sekaligus yaitu sebagai *server* dan *client*. Pada *router* 2, dihubungkan dengan *access point* untuk mengubah jaringan *wired* menjadi *wireless*. Dari *access point*, kemudian di pancarkan ke arah *client*. Konfigurasi jaringan yang digunakan pada *tes-bed wireless* berbeda-beda. Berikut ini perbedaan konfigurasi setiap jaringan :

- Jaringan VoIP tanpa pengaman VPN

  PC *client* VoIP dihubungkan ke PC *server* melalui *router* yang sudah di atur secara *static* untuk *routing*-nya sesuai dengan alamat IP masing-masing tanpa ada pengaturan pengamanan data.

- Jaringan VoIP menggunakan pengaman VPN

  PC *client* VoIP dihubungkan ke PC *server* melalui *router* yang sudah diatur menggunakan pengaman VPN dengan protokol PPTP (*Point to Point Tunneling Protocol*). *Router* mikrotik RB750 berfungsi sebagai *server* VPN yang akan memberikan nomor IP *tunneling* kepada *client* setelah memasukan *username* dan *password* yang telah terdaftar. Pada sisi *client* VPN, PC melakukan sambungan menggunakan layanan khusus VPN yang telah tersedia di menu *network and sharing centre* pada *windows*. Dengan memasukan *username* dan *password* yang telah terdaftar pada *server* VPN ke dalam layanan sambungan VPN pada *windows, client* dapat saling berkomunikasi dengan aman.

3.7 Identifikasi Perangkat Pendukung

Dalam penelitian ini dibutuhkan perangkat-perangkat keras pendukung untuk membangun sistem VoIP. Berikut ini spesifikasi perangkat keras yang diperlukan:

1. *Server* dan *Client* 1 (Laptop ASUS)
   - Sistem Operasi Windows 8
   - Intel i3 CPU 2.4 GHz
   - 4 GB RAM
   - *Ethernet* 10/100 Mbps
2. *Client* 2 (laptop Lenovo)
   - Sistem Operasi Windows 8
   - Intel i3 CPU 2.0 GHz
   - 2 GB RAM
   - *Ethernet* 10/100 Mbps
3. Client 3 (laptop ASUS)
   - Sistem Operasi Windows 8
   - Intel i3 CPU 2.4 GHz
   - 2 GB RAM
   - *Ethernet* 10/100 Mbps
4. *Router* Mikrotik RB750
   - 5 Port *Ethernet* 10/100 Mbps
5. *Switch* TP-LINK model TL-SF10008D
   - 8 Port *Support* 10/100 Mbps

6. *Access Point* TP-LINK model TL-WA701ND
   - *Support* 802.11b/g/n standars
   - Kecepatan hingga 100 Mbps
7. Kabel UTP
   - Categori 5
   - *Speed up to* 100 Mbps
   - Tipe kabel 100Base-T

Selain perangkat keras penunjang, diperlukan juga aplikasi perangkat lunak yang diperlukan untuk mendukung proses penelitian antara lain :

1. TrixBox, merupakan *software* yang digunakan sebagai *server* VoIP. Langkah-langkah instalasi TrixBox pada VmWare dan konfigurasi *server* VoIP akan ada pada Lampiran 1.
2. Cain and Abel, merupakan *software* yang digunakan untuk menguji keamanan VPN dengan cara melakukan penyadapan komunikasi VoIP. Langkah-langkah untuk melakukan proses penyadapan akan ada pada Lampiran 2.
3. X-Lite, merupakan *software softphone* yang berfungsi sebagai *User Agent* pada *Client.* Cara konfigurasi X-Lite agar dapat terhubung dengan server VoIP akan ada pada Lampiran 3.
4. Wireshark, merupakan aplikasi yang digunakan untuk meng-*capture* dan menganalisa *trafik* yang terjadi pada suatu *interface*.

3.8 Pengambilan Kesimpulan dan Saran

Pada tahap ini dilakukan pengambilan kesimpulan berdasarkan hasil simulasi serta analisis dan dilakukan pemberian saran-saran yang dimaksudkan kepada pembaca yang akan melakukan studi tentang penelitian ini ataupun sebagai pendukung dari penelitiannya.

BAB IV
HASIL DAN PEMBAHASAN

4.1 Umum

Bab IV menjelaskan tentang hasil dan pembahasan dari penelitian. Data yang disajikan merupakan hasil dari analisis menggunakan Trixbox sebagai *server* VoIP. Penelitian ini membahas tentang pengaruh penggunaan VPN dan media transmisi yang berbeda untuk layanan VoIP. Analisis yang dilakukan meliputi parameter *delay end to end, packet loss* dan *throughput*.

Pada bab ini juga akan ditampilkan hasil analisis performansi VoIP dengan menggunakan analisis perhitungan manual yang mengacu pada standar *codec* G.711 dari *Institute of Electrical and Electronics Engineers* (IEEE), Hal ini dilakukan sebagai perbandingan terhadap hasil simulasi dengan menggunakan Trixbox.

4.2 Analisa Konfigurasi Jaringan

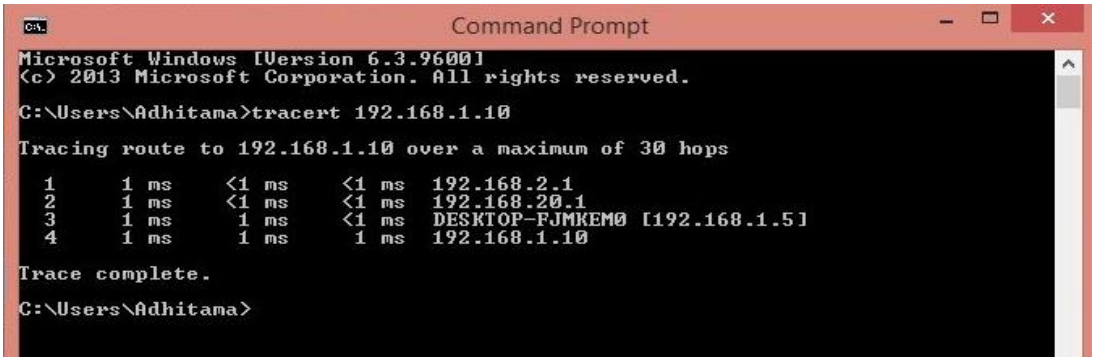
Pengambilan data dilakukan dengan menggunakan jaringan sederhana pada setiap konfigurasi jaringan yang diujikan, yaitu menggunakan satu buah laptop yang digunakan sebagai *server* sekaligus *client* dan dua buah *router* untuk menghubungkan dua laptop *client*.

Pada penelitian ini dilakukan percobaan menggunakan media *wired* dan *wireless*. Untuk media *wired*, antar *router* dihubungkan menggunakan kabel UTP dengan tipe *cross-over* dan dari *switch* ke *client* dihubungkan dengan kabel UTP tipe *straight*. Sedangkan untuk media *wireless*, pada sisi *client* ditempatkan sebuah *access point* sebagai media *wireless* yang akan menghubungkan laptop *client*. Konfigurasi yang digunakan pada penelitian ini ada 2 macam, yaitu konfigurasi jaringan tanpa menggunakan VPN dan jaringan menggunakan VPN.

4.2.1 Konfigurasi tanpa VPN

Pada konfigurasi tanpa VPN, konektifitas antar perangkat bersifat terbuka dimana semua *client* dapat melakukan panggilan ke semua *client* tanpa ada pengamanan. Dalam konfigurasi ini jenis *routing* yang digunakan adalah *static routing*, dikarenakan semua konfigurasi menggunakan alamat ipv4 maka proses transmisi data yang terjadi pada saat melalui *router* hanya *routing* dan *forwarding* paket seperti jaringan pada umumnya.

Hasil *traceroute* pada Gambar 4.1 dilakukan dari *server* ke *client*. Pada hasil *traceroute* tersebut terlihat bahwa *server* telah berhasil tersambung ke *client*.

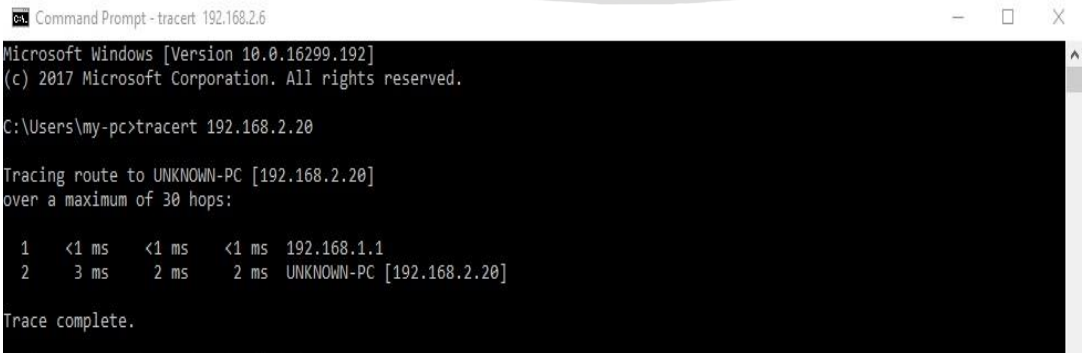


Gambar 4.1 Tampilan *Traceroute* pada jaringan tanpa VPN

4.2.2 Konfigurasi dengan VPN

Pada konfigurasi jaringan dengan VPN juga memiliki topologi yang sama dengan topologi jaringan tanpa VPN sehingga tidak ada perbedaan jumlah *hop*. Perbedaannya terletak pada konfigurasi sistem pengamanan data yang menggunakan VPN dengan protokol PPTP. Dalam konfigurasi ini fungsi *router* sebagai VPN *server* yang mengatur data berupa *username*, *password*, dan nomor IP yang didapat ketika melakukan sambungan menggunakan VPN. Sedangkan pada sisi *client* VPN menggunakan opsi *open network and sharing* yang terdapat *os windows* agar terhubung pada jaringan VPN. Untuk konfigurasi VPN *server* dan VPN *client* dapat dilihat pada Lampiran 4.

Jenis *routing* yang digunakan pada konfigurasi ini yaitu *static routing*. Hasil *traceroute* pada Gambar 4.2 merupakan hasil *traceroute* dari *server* ke *client*. Dari hasil *traceroute* menunjukkan bahwa *server* dapat tersambung dengan *client* menggunakan pengaman VPN.



Gambar 4.2 Tampilan *Traceroute* pada jaringan VPN

4.3 Analisis Perhitungan pada Jaringan VoIP

Pada analisis performansi VoIP dihitung beberapa parameter yang akan digunakan sebagai perbandingan hasil menggunakan *software* Trixbox dengan hasil secara perhitungan. Parameter-parameter perhitungan ini meliputi perhitungan *payload* dalam VoIP, *delay end to end*, *packet loss*, dan *throughput*.

4.3.1 Analisis Paket Data Aplikasi VoIP

Untuk analisis performansi VoIP pada skrispsi ini menggunakan *audio codec* G.711. Berikut spesifikasi *codec* yang digunakan.

Tabel 4.1 Spesifikasi Codec G.711

Codec	Bir Rate (Kbps)	Codec Sample Interval (ms)	Delay Codec (ms)	Mean Opinion Score (MOS)
G.711	64	10	0.75	4.1

(**Sumber:** IEEE)

Untuk mencari nilai *payload* maka digunakan Persamaan 2.1:

$$P_{LA} = B_{codec} \times \text{ frame } rate = 64 \times 10^3 \text{ bps} \times 10 \times 10^{-3} s = 640\ bit$$

Maka Payload VoIP $= 640\ bit\ = 80\ byte$

Setelah diketahui besar *payload* paket data pada *audio* maka banyaknya bit yang terisi dalam paket VoIP dapat dihitung dengan Persamaan 2.2:

$$P_{A-size} = header_{IP/UDP/RTP} + P_{LA} = 320\ bit + 640\ bit = 960\ bit$$

Sedangkan jumlah paket *audio* yang dihasilkan tiap detik dihitung dengan Persamaan 2.3:

$$P_A = \frac{B_{codec}}{P_{LA}} = \frac{64x10^3}{640} = 100\ packet\ per\ second$$

Sehingga *bandwidth audio* dapat dihitung dengan Persamaan 2.4 :

$$B_A = P_{A-size} \times\ P_A = 960x100 = 96000\ bit\ per\ second = 96 \text{ Kbps}$$

4.3.2 Analisis *Signal to Noise Ratio* (SNR)

Pada Tabel 4.2 merupakan parameter OFDM untuk jaringan LAN sesuai dengan standar IEEE yaitu:

Tabel 4.2 Parameter OFDM

Standard	IEEE 802.11n	IEEE 802.3u
Bandwidth (MHz)	20	20
FFT size	64	64
Number of subcarrier (N_{used})	56	56
Channel coding	Convolutional code (1/2, 2/3, 3/4)	Convolutional code (1/2, 2/3, 3/4)
Modulation	BPSK, QPSK, 16-QAM, 64 QAM	BPSK, QPSK, 16-QAM, 64 QAM
Factor of sampling	8/7	8/7
Spatial stream	1	1 ~ 2

(**Sumber:** IEEE)

Untuk jaringan LAN telah ditetapkan standar *Receiver* SNR oleh IEEE 802.11n dan 802.3u seperti pada Tabel 4.3 berikut :

Tabel 4.3 Receiver SNR Standard IEEE 802.11n dan 802.3u

Media Transmisi	Tipe Modulasi	*Receiver* SNR (dB)
Wired	QPSK $^1/_2$	16
Wireless	QPSK $^1/_2$	21

(**Sumber:** Standard IEEE)

4.3.3 Penentuan *Bit rate*

Karena pada penelitian ini tidak memperhitungkan penggaruh panjang *cyclic prefix*, sehingga :

$$T_s = T_b$$

$$T_s = \frac{N_{FFT}}{BW \times \eta}$$

$$T_s = \frac{64}{10^7 \times \frac{8}{7}} = 5.5999 \times 10^{-5}\ s$$

Maka perhitungan *bit rate* untuk modulasi QPSK dengan *coding rate* (cr) ½, dengan N_{used} sebesar 56, bm atau jumlah *bit* yang digunakan tiap modulasi adalah 2, dan T_s = $5.5999 \times 10^{-5}\ s$ adalah:

$$bit\ rate(QPSK\ 1/2) = N_{used} \times\ bm\ \times \frac{c_r}{T_s}$$

$$= 56 \times 2 \times \frac{\frac{1}{2}}{5.5999 \times 10^{-5}}$$
$$= 1000017.857\ bps$$
$$= 9.5\ Mbps$$

Hasil perhitungan *bit rate* pada media *wired* maupun *wireless* adalah sama karena menggunakan jenis modulasi yang sama.

4.3.4 Analisis *Bit Error Rate* (BER)

Dalam perhitungan BER diperlukan nilai Eb/No (*Energy bit to Noise Ratio*). Eb/No merupakan perbandingan energi sinyal per *bit* terhadap *noise*.

Untuk memperoleh nilai Eb/No pada media *wired* dengan nilai SNR = 16 dB yang mempunyai data *rate* sebesar 9.5 Mbps menggunakan teknik modulasi QPSK ½ dapat diperoleh dengan perhitungan sebagai berikut :

$$\frac{E_b}{N_o} = \frac{S}{N} - 10log\frac{B}{R}$$
$$= 16 - 10log\frac{10 \times 10^6}{9.5 \times 10^6}$$
$$= 16 - 0.2226$$
$$= 15.7774\ dB$$

Dengan cara perhitungan yang sama untuk memperoleh nilai Eb/No pada media *wireless* yang mempunyai nilai SNR dan data *rate* yang telah dihitung sebelumnya, maka diperoleh hasil analisis nilai Eb/No = 20.7774.

Setelah mendapatkan nilai Eb/No tiap media transmisi, langkah berikutnya adalah menghitung nilai BER pada masing-masing media transmisi yang digunakan.

1. *Wired*

Besarnya nilai BER atau probabilitas *bit error* (P_b) menggunakan teknik modulasi QPSK ½ dengan Eb/No = 15.7774 dB dapat dihitung dengan menggunakan Persamaan 2.8,

$$P_b = P_s = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

$$= \frac{1}{2} erfc \sqrt{\frac{E_b}{N_0}}$$

$$= \frac{1}{2} erfc\sqrt{15.77}$$

$$= \frac{1}{2} erfc(3.9720)$$

Sehingga nilai $P_{b\,wired} = 0.97005 \times 10^{-5}$

2. *Wireless*

Besarnya nilai BER atau probabilitas *bit error* (P_b) menggunakan teknik modulasi QPSK ½ dengan Eb/No = 20.7774 dB adalah.

$$P_b = P_s = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

$$= \frac{1}{2} erfc \sqrt{\frac{E_b}{N_0}}$$

$$= \frac{1}{2} erfc\sqrt{20.7774}$$

$$= \frac{1}{2} erfc(4.5582)$$

Sehingga nilai $P_{b\,wireless} = 0.57305 \times 10^{-5}$

4.3.5 Analisis *delay end to end*

Pada aplikasi VoIP yang bersifat *full duplex*, maka *delay* dihitung dari sumber ke penerima sampai ke sumber lagi. Untuk analisis jaringan VoIP yang digunakan, maka perhitungan dilakukan untuk mengetahui *delay end to end* aplikasi VoIP dengan Persamaan 2.10 :

$$t_{end-to-end} = t_{codec} + t_{jaringan}$$

Dengan :

t_{codec} = *delay* codec (ms)

$t_{jaringan}$ = *delay* jaringan (ms)

- ***Delay Codec***

Delay ini tediri dari waktu untuk mengakumulasi sample suara ke dalam *frame* suara, waktu untuk mengkompresi paket suara, waktu untuk memuati *frame* suara ke dalam paket dan mentransfer paket tersebut ke jaringan *transport* dan *delay hardware* yang bersifat tetap. *Delay* pada *codec audio* G.711 sebesar 0.75 *ms*, sehingga *delay codec* ini dapat dihitung dengan persamaan 2.11 berikut :

$$t_{codec} = 2 \times t_{audio} = 2 \times 0.75 = 1.5\ ms$$

- ***Delay* Jaringan**

Delay total merupakan besarnya *delay* yang diperlukan untuk mengirimkan data dari sisi pengirim ke sisi penerima. *Delay* total dapat dihitung dengan Persamaan 2.12 sebagai berikut:

$$t_{jaringan} = t_{proces} + t_{prop} + t_{trans} + t_w$$

Dengan:

$t_{jaringan}$ = *delay* total (ms)
t_{proces} = *delay* proses (ms)
t_{pros} = *delay* propagasi (ms)
t_{trans} = *delay* transmisi (ms)
t_w = *delay* antrian (ms)

- ***Delay* Proses**

Delay proses adalah waktu yang dibutuhkan untuk memproses paket data dan untuk menentukan kemana data tersebut akan diteruskan. *Delay* proses berupa *delay* enkapsulasi dan *delay* dekapsulasi. *Delay* enkapsulasi dan *delay* dekapsulasi data melalui jaringan LAN untuk menambahkan *header* pada masing-masing layer dan perubahan format data.

Ketika data VoIP (*payload audio*) dikirim dari sumber melewati *layer* aplikasi menuju *layer transport*, data akan diubah menjadi segmen. *Message* data ketika melewati *transport* data akan mengalami penambahan *header* RTP dan *header* UDP, sehingga dapat dihitung dengan Persamaan 2.13 :

$$W_{message} = W_{data} + Header_{RTP} = 80\ byte + 12byte = 92\ byte$$

Dari layer *transport* segmen ditambahkan dengan *header* UDP karena menggunakan aplikasi yang bersifat *real time*.

$$W_{segmen} = W_{message} + Header_{UDP} = 92\ byte + 8\ byte = 100\ byte$$

Dari layer *transport* data dibawa menuju layer *network*, yaitu IP untuk diubah menjadi *datagram* dengan penambahan *header* IP. Dengan menggunakan Persamaan 2.14 diperoleh:

$$W_{datagram} = W_{segmen} + Header_{IP} = 100\ byte + 20\ byte = 120\ byte$$

Pada segmen *datagram* tidak melebihi MTU *Ethernet*, maka *datagram* IP tidak difragmentasi. Dengan menggunakan Persamaan 2.15 maka *frame-frame* tadi dienkapsulasi dengan rumus :

$$W_{frame} = W_{datagram} + Header_{Ethernet} = 120\ byte + 18\ byte = 138\ byte$$

Maka besar *delay* enkapsulasi dengan kecepatan *layer ethernet* adalah 100 Mbps. Dihitung dengan persamaan 2.16 :

$$t_{enc} = \frac{W_{frame\ total}}{c_{Ethernet}} \times 8 = \frac{138}{10^8} \times 8 = 0.01104\ 10^{-3}\ s$$

Besar *delay* dekapsulasi dengan kecepatan layer *ethernet* adalah 100 Mbps. Dihitung dengan persamaan 2.17 :

$$t_{dec} = \frac{W_{frame\ total}}{c_{Ethernet}} \times 8 = \frac{138}{10^8} \times 8 = 0.01104\ \times 10^{-3}\ s$$

Sehingga didapatkan *delay* proses dengan Persamaan 2.18 :

$$t_{proc} = t_{enc} + t_{dec}$$

$$= (0.01104\ + 0.01104)\ \times 10^{-3}\ s$$

$$= 0.02208 \text{ ms}$$

- ***Delay* propagasi**

Delay propagasi adalah jumlah waktu yang dibutuhkan oleh data untuk berpropagasi pada media transmisi. Dengan jarak terjauh 20 m, maka besar *delay* propagasi pada media *wired* dari *host* sumber menuju *host* tujuan dapat dihitung dengan Persamaan 2.19 :

$$t_{prop} = \frac{L_k}{v_{prop}} = \frac{20}{2 \times 10^8} = 0.0001\ ms$$

- ***Delay* transmisi**

Delay transmisi adalah waktu yang dibutuhkan untuk meletakkan semua data pada media transmisi, dipengaruhi oleh ukura paket dan kapasitas media transmisi. *Delay* ini hanya terjadi sekali saja di sumber informasi. Besar *delay* transmisi sesuai dengan persamaan 2.20:

$$t_{trans} = \frac{w}{c} \times 8 = \frac{138}{100 \times 10^6\, bps} \times 8 = 0.0104\, ms$$

Besarnya *delay* transmisi dipengaruhi oleh kapasitas media transmisi, semakin besar kapasitas media transmisi maka *delay* transmisi akan semakin kecil.

- ***Delay* antrian**

Delay antrian adalah waktu yang dibutuhkan data selama berada dalam antrian untuk ditransmisikan. *Delay* ini disebabkan oleh waktu proses yang diperlukan oleh *router* di dalam menangani paket di sepanjang jaringan.

Pada analisis aplikasi VoIP, menggunakan model antrian M/M/1 dengan disiplin antrian FIFO. Untuk nilai waktu pelayanan dapat dihitung dengan persamaan 2.21 :

$$\mu = \frac{c}{w} = \frac{100 \times 10^6}{138 \times 8} = 90579.8\, paket/s$$

Besarnya kecepatan kedatangan dengan faktor utilisasi 0.9 dengan model antrian M/M/1, maka *delay* antrian dapat dihitung sebagai berikut :

$$\lambda_w = \mu\rho = 90579.8 \times 0.9 = 81521.8$$

Sehingga besarnya *delay* antrian adalah

$$t_w = \frac{\rho/\mu}{(1-\rho)} = \frac{\frac{0,9}{90579.8}}{(1-0.9)} = 0.09936 \text{ ms}$$

Jadi besar *delay end to end* adalah

$$\begin{aligned} t_{end-to-end} &= 2\,(t_{codec} + t_{proces} + t_{prop} + t_{trans} + t_w) \\ &= 2\,(1.5 + 0.02208 + 0.0001 + 0.0104 + 0.09936) \\ &= 3.2639 \text{ ms} \end{aligned}$$

Dengan cara yang sama diperoleh nilai *delay end to end* untuk jaringan *wireless* dengan $v_{prop} = 3 \times 10^8$ sesuai dengan kecepatan propagasi gelombang radio pada spesifikasi *access point*. Besar *delay end to end* aplikasi VoIP melalui jaringan *wired* dan jaringan *wireless* ditunjukkan pada Tabel 4.4 berikut:

Tabel 4.4 Hasil Perhitungan *Delay end to end*

Delay end to end (ms)	
Wired	*Wireless*
3.2639	3.2636

(**Sumber:** Perhitungan)

4.3.6 Analisis *Packet loss*

Packet loss adalah hilangnya paket data yang dikirimkan *host* sumber. *Packet loss* dapat dihitung dengan Persamaan 2.24 :

1. Probabilitas *Packet loss* VoIP

Perhitungan *packet loss* VoIP dipengaruhi oleh BER standar untuk VoIP dengan codec G.711 yakni 1% (10^{-2}) (www.voip-info.org), secara matematis dituliskan sesuai dengan persamaan berikut:

$$\rho_{voip} = \rho_b = 10^{-2}$$

2. Probabilitas *packet loss* jaringan

Perhitungan *packet loss* jaringan VoIP untuk tipe modulasi QPSK ½ sesuai dengan persamaan berikut:

$$\rho_{QPSK\ 1/2} = P_{QPSK\ 1/2} = 0.57305 \times 10^{-5}$$

Maka probabilitas *packet loss* total untuk modulasi QPSK ½ adalah

$$\rho_{total} = 1 - \left[\left(1 - \rho_{voip}\right)\left(1 - \rho_{QPSK\frac{1}{2}}\right)\right]$$

$$= 1 - [(1 - 10^{-2})(1 - 0.57307 \times 10^{-5})]$$

$$= 0.0101$$

Besar *packet loss* aplikasi VoIP melalui jaringan *wired* dan jaringan *wireless* ditunjukkan pada Tabel 4.5 berikut:

Tabel 4.5 Hasil Perhitungan *Packet Loss*

Packet Loss (%)	
Wired	*Wireless*
0.0101	0.0101

(**Sumber:** Perhitungan)

4.3.7 Analisis *Throughput*

Dalam analisis *throughput* menunjukkan jumlah data yang diterima dengan benar pada sisi penerima setelah melewati media transmisi pada data *link layer* dari *client to client*. Nilai a dapat dihitung dengan Persamaan 2.26 :

$$\alpha = \left(1 + \frac{2t_{prop+}2t_{trans}}{t_{trans}}\right) = \left(1 + \frac{2 \times 0.0001 \times 10^{-3} + 2 \times 0.0104 \times 10^{-3}}{0.0104 \times 10^{-3}}\right) = 3.019$$

Dengan nilai *packet loss* yang telah dicari di persamaan sebelumnya, maka nilai *throughput* dapat diketahui dengan Persamaan 2.25 berikut :

$$\lambda = \frac{1}{t_v} = \frac{(1-\rho)}{t_{trans}[1+(\alpha-1)\rho]}$$

$$\lambda = \frac{1}{t_v} = \frac{(1-0.0101)}{0.0104 \times 10^{-3}[1+(3.019-1)0.0101]}$$

$$= 93.28\ kbps$$

Dengan cara yang sama diperoleh nilai *throughput* dengan media transmisi *wireless* yang ditunjukan pada Tabel 4.6 berikut :

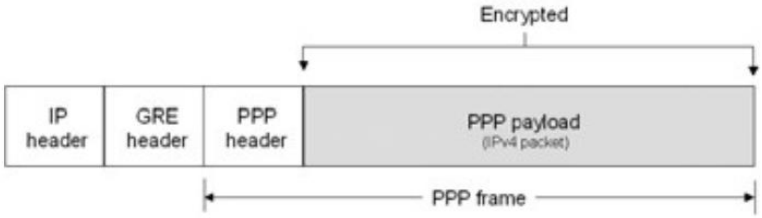
Tabel 4.6 Hasil Perhitungan *Throughput*

Throughput (kbps)	
Wired	*Wireless*
93.28	93.29

(**Sumber:** Perhitungan)

4.4 Analisis VoIP pada Jaringan VPN

Pada jaringan VPN, data yang dilewatkan antara *server* dan *client* ditransmisikan pada IP *datagram* yang memiliki paket PPP. Kemudian, *Generic Routing Encapsulation* (GRE) melakukan enkapsulasi paket IP yang berisi paket PPP menjadi paket GRE, lalu paket GRE tersebut dibungkus dalam sebuah paket IP untuk dilewatkan dalam *tunnel*.



Gambar 4.3 Format paket PPTP
(**Sumber:** technet.microsoft.com)

Paket data yang dilewatkan melalui PPTP *tunnel* akan berbeda dengan paket data yang dilewatkan melalui pengalamatan IP biasa. Paket PPP dibuat oleh PPTP *server* adalah paket data yang telah terenkripsi, GRE *header* meringkas paket PPP tersebut menjadi IP *Datagram*, kemudian IP *Datagram* dibungkus oleh IP *Delivery Header* yang membawa informasi penting untuk *datagram* untuk melintasi jaringan. IP *Datagram* tersebut dirutekan melalui jaringan hingga mencapai tujuan PPTP *client* yang terhubung ke jaringan.

4.4.1 Analisis *delay end to end*

Pada aplikasi VoIP yang bersifat *full duplex*, maka *delay* dihitung dari dari sumber ke penerima sampai ke sumber lagi. Untuk analisis jaringan VoIP yang digunakan, maka perhitungan dilakukan untuk mengetahui *delay end to end* aplikasi VoIP dengan Persamaan 2.10 :

$$t_{end-to-end} = t_{codec} + t_{jaringan}$$

Dengan :

t_{codec} = *delay* codec (ms)

$t_{jaringan}$ = *delay* total (ms)

- ***Delay codec***

Delay ini tediri dari waktu untuk mengakumulasi sampel suara ke dalam *frame* suara, waktu untuk mengkompresi paket suara, waktu untuk memuati *frame* suara ke dalam paket dan mentransfer paket tersebut ke jaringan *transport* dan *delay hardware* yang bersifat tetap. *Delay* pada *codec audio* G.711 sebesar 0.75 *ms*, sehingga *delay codec* ini dapat dihitung dengan persamaan 2.11 berikut :

$$t_{codec} = 2 \times t_{audio} = 2 \times 0.75 = 1.5\ ms$$

- ***Delay* Total**

Delay total merupakan besarnya *delay* yang diperlukan untuk mengirimkan data dari sisi pengirim ke sisi penerima. *Delay* total dapat dihitung dengan Persamaan 2.12 sebagai berikut:

$$t_{tot} = t_{proces} + t_{prop} + t_{trans} + t_w$$

Dengan:

$t_{jaringan}$ = *delay* jaringan (ms)

t_{proces} = *delay* proses (ms)

t_{prop} = *delay* propagasi (ms)

t_{trans} = *delay* transmisi (ms)

t_w = *delay* antrian (ms)

- ***Delay* Proses**

Delay proses adalah waktu yang dibutuhkan untuk memproses paket data dan untuk menentukan kemana data tersebut akan diteruskan. *Delay* proses berupa *delay* enkapsulasi dan *delay* dekapsulasi. *Delay* enkapsulasi dan *delay* dekapsulasi data melalui jaringan LAN untuk menambahkan *header* pada masing-masing layer dan perubahan format data.

Ketika data VoIP (*payload audio*) dikirim dari sumber melewati layer aplikasi menuju layer *transport*, data akan diubah menjadi segmen. *Message* data ketika melewati *transport* data akan mengalami penambahan *header* RTP dan *header* UDP, sehingga dapat dihitung dengan Persamaan 2.13 :

$$W_{message} = W_{data} + Header_{RTP} = 80\ byte + 12byte = 92\ byte$$

Dari layer *transport* segmen ditambahkan dengan *header* UDP karena menggunakan aplikasi yang bersifat *real time*.

$$W_{segmen} = W_{message} + Header_{UDP} = 92\ byte + 8\ byte = 100\ byte$$

Dari *layer transport* data dibawa menuju *layer network*, yaitu IP untuk diubah menjadi *datagram* dengan penambahan *header* IP. Dengan menggunakan Persamaan 2.14 diperoleh:

$$W_{datagram} = W_{segmen} + Header_{IP} = 100\ byte + 20\ byte = 120\ byte$$

Kemudian paket data akan dilewatkan melalui PPTP *tunnel*. Paket yang melalui PPTP *tunnel* adalah paket data yang telah terenkripsi. Sebelum paket melalui proses *tunneling*, akan mengalami penambahan *header* PPP, GRE, dan IP yang didapat dari PPTP *server*, diperoleh :

$$W_{datagram} = W_{datagram} + Header_{PPP} + Header_{GRE} + Header_{IP}$$

$$= 120\ byte + 4\ byte\ + 12\ byte\ + 20\ byte$$
$$= 156\ byte$$

Pada segmen *datagram* tidak melebihi MTU *Ethernet*, maka *datagram* IP tidak difragmentasi. Dengan menggunakan Persamaan 2.15 maka *frame-frame* tadi dienkapsulasi dengan rumus :

$$W_{frame} = W_{datagram} + Header_{Ethernet} = 156\ byte + 18\ byte = 174\ byte$$

Maka besar *delay* enkapsulasi dengan kecepatan layer *ethernet* adalah 100 Mbps adalah:

$$t_{enc} = \frac{W_{frame\ total}}{c_{Ethernet}} \times 8 = \frac{174}{10^8} \times 8 = 0.01392 \times 10^{-3}\ s$$

Besar *delay* dekapsulasi dengan kecepatan layer *ethernet* adalah 100 Mbps adalah:

$$t_{dec} = \frac{W_{frame\ total}}{c_{Ethernet}} \times 8 = \frac{174}{10^8} \times 8 = 0.01392\ \times 10^{-3}\ s$$

Sehingga didapatkan *delay* proses dengan Persamaan 2.18 :

$$t_{proc} = t_{enc} + t_{dec}$$
$$= (0.01392\ + 0.01392) \times 10^{-3}\ s$$
$$= 0.02784\ ms$$

- ***Delay* propagasi**

Delay propagasi adalah jumlah waktu yang dibutuhkan oleh data untuk berpropagasi pada media transmisi. Dengan jarak terjauh 20 m, maka besar *delay* propagasi pada media *wired* dari *host* sumber menuju *host* tujuan dapat dihitung dengan Persamaan 2.19 :

$$t_{prop} = \frac{L_k}{v_{prop}} = \frac{20}{2 \times 10^8} = 0.0001\ ms$$

- ***Delay* transmisi**

Delay transmisi adalah waktu yang dibutuhkan untuk meletakkan semua data pada media transmisi, dipengaruhi oleh ukura paket dan kapasitas media transmisi. *Delay* ini hanya terjadi sekali saja di sumber informasi. Besar *delay* transmisi sesuai dengan persamaan 2.20:

$$t_{trans} = \frac{w}{c} \times 8 = \frac{174}{100 \times 10^6\ bps} \times 8 = 0.0139\ ms$$

Besarnya *delay* transmisi dipengaruhi oleh kapasitas media transmisi, semakin besar kapasitas media transmisi maka *delay* transmisi akan semakin kecil.

- ***Delay* antrian**

Delay antrian adalah waktu yang dibutuhkan data selama berada dalam antrian untuk ditransmisikan. *Delay* ini disebabkan oleh waktu proses yang diperlukan oleh *router* di dalam menangani paket di sepanjang jaringan.

Pada analisis aplikasi VoIP, menggunakan model antrian M/M/1 dengan disiplin antrian FIFO. Untuk nilai waktu pelayanan dapat dihitung dengan persamaan 2.21 :

$$\mu = \frac{c}{w} = \frac{100 \times 10^6}{174 \times 8} = 71839.1\ paket/s$$

Besarnya kecepatan kedatangan dengan faktor utilisasi 0.9 dengan model antrian M/M/1, maka *delay* antrian dapat dihitung sebagai berikut :

$$\lambda_w = \mu\rho = 71839.1 \times 0.9 = 64655.2$$

Sehingga besarnya *delay* antrian adalah

$$t_w = \frac{\rho/\mu}{(1-\rho)} = \frac{\frac{0.9}{71839.1}}{(1-0.9)} = 0.01253 \text{ ms}$$

Jadi besar *delay end to end* adalah

$$t_{end-to-end} = 2\ (t_{codec} + t_{proces} + t_{prop} + t_{trans} + t_w)$$
$$= 2\ (1.5 + 0.02784 + 0.0001 + 0.0139 + 0.01253)$$
$$= 3.1087 \text{ ms}$$

Maka besar *delay end to end* aplikasi VoIP melalui jaringan *wired* dan jaringan *wireless* ditunjukkan pada Tabel 4.7 berikut:

Tabel 4.7 Hasil Perhitungan *Delay end to end* VPN

Delay end to end (ms)	
Wired	*Wireless*
3.1087	3.1086

(**Sumber:** Perhitungan)

4.4.2 Analisis *Packet Loss*

Packet loss adalah hilangnya paket data yang dikirimkan *host* sumber. *Packet loss* dapat dihitung dengan Persamaan 2.24 :

1. Probabilitas *Packet loss* VoIP

Perhitungan *packet loss* VoIP dipengaruhi oleh BER standar untuk VoIP dengan codec G.711 yakni 1% (10^{-2}) (www.voip-info.org), secara matematis dituliskan sesuai dengan persamaan berikut:

$$\rho_{voip} = \rho_b = 10^{-2}$$

2. Probabilitas *packet loss* jaringan

Perhitungan *packet loss* jaringan VoIP untuk tipe modulasi QPSK ½ sesuai dengan persamaan berikut:

$$\rho_{QPSK\ 1/2} = P_{QPSK\ 1/2} = 0.57305 \times 10^{-5}$$

Maka probabilitas *packet loss* total untuk modulasi QPSK ½ adalah

$$\rho_{total} = 1 - \left[\left(1 - \rho_{voip}\right)\left(1 - \rho_{QPSK\frac{1}{2}}\right)\right]$$

$$= 1 - [(1 - 10^{-2})(1 - 0.57307 \times 10^{-5})]$$

$$= 0.0101$$

Besar *packet loss* aplikasi VoIP melalui jaringan *wired* dan jaringan *wireless* ditunjukkan pada Tabel 4.8 berikut:

Tabel 4.8 Hasil Perhitungan *Packet Loss* VPN

Packet Loss (%)	
Wired	*Wireless*
0.0101	0.0101

(**Sumber:** Perhitungan)

4.4.3 Analisis *Throughput*

Dalam analisis *throughput* menunjukkan jumlah data yang diterima dengan benar pada penerima setelah melewati media transmisi pada data *link layer* dari *client to client*. Nilai *a* dapat dihitung dengan Persamaan 2.26 :

$$\alpha = \left(1 + \frac{2t_{prop+}2t_{trans}}{t_{trans}}\right) = \left(1 + \frac{2 \times 0.0001 \times 10^{-3} + 2 \times 0.0139 \times 10^{-3}}{0.0139 \times 10^{-3}}\right) = 3.014$$

Dengan nilai *packet loss* yang telah dicari di persamaan sebelumnya, maka nilai *throughput* dapat diketahui dengan Persamaan 2.25 berikut :

$$\lambda = \frac{1}{t_v} = \frac{(1-\rho)}{t_{trans}[1+(\alpha-1)\rho]}$$

$$\lambda = \frac{1}{t_v} = \frac{(1-0.0101)}{0.0139 \times 10^{-3}[1+(3.014-1)0.0101]}$$

$$= 69.79\ kbps$$

Dengan cara yang sama diperoleh nilai *throughput* dengan media transmisi *wireless* yang ditunjukan pada Tabel 4.9 berikut :

Tabel 4.9 Hasil Perhitungan *Throughput*

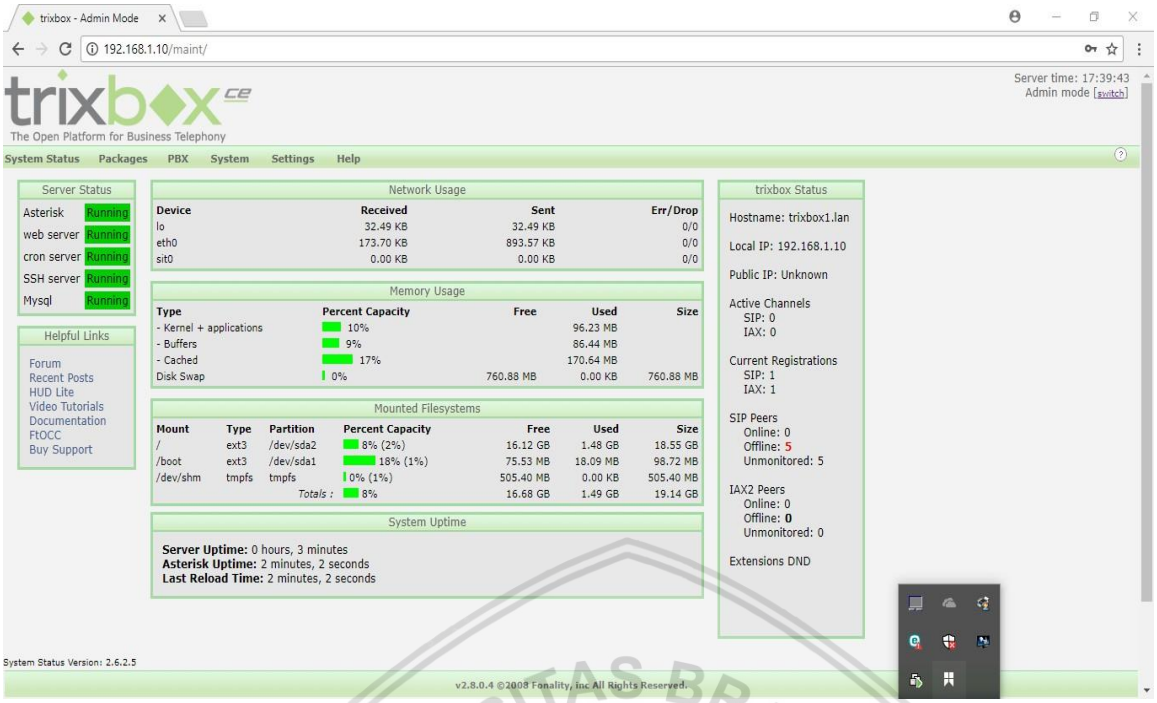
Throughput (kbps)	
Wired	*Wireless*
69.79	69.80

(**Sumber:** Perhitungan)

4.5 Analisis Simulasi VoIP

Voice Over Internet Protocol (VoIP) merupakan teknologi yang mampu mengubah suara analog (suara manusia) menjadi paket data kemudian melalui jaringan data *public internet* paket data dilewatkan. Protokol VoIP digunakan adalah *Session Initiation Protocol* (SIP). Protokol SIP memiliki karakteristik *client-server*, untuk itu *request* diberikan oleh *client* dan *request* ini diberikan ke *server*. Kemudian, *server* mengolah *request* dan memberikan tanggapan terhadap *request* tersebut ke *client*.

Pada skripsi ini digunakan aplikasi TrixBox sebagai *server* VoIP. Pada masing-masing konfigurasi jaringan yang dibuat, baik jaringan tanpa menggunakan VPN dan jaringan menggunakan VPN akan diimplementasikan penggunaan aplikasi tersebut. Gambar 4.4 di bawah ini adalah gambaran aplikasi TrixBox.



Gambar 4.4 Tampilan GUI TrixBox

Terdapat tiga parameter yang diambil dalam pengambilan data yaitu *delay end to end*, *throughput*, dan *packet loss*. Parameter tersebut dianggap dapat mewakili untuk unjuk kerja dari VoIP. Nilai-nilai parameter tersebut dapat diperoleh dengan menggunakan *software* Wireshark. Perhitungan dilakukan sebanyak 5 kali agar memberikan nilai yang lebih akurat.

4.5.1 Analisa *Delay end to end*

Delay end to end adalah waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. Hal ini dapat dipengaruhi oleh jarak, media fisik, *congesti* atau juga waktu proses yang lama. Pengambilan parameter *delay end to end* dilakukan dengan cara *client* melakukan panggilan terhadap *client* lain. Kemudian disaat yang bersamaan pada salah satu *client* melakukan *capture* data atau penangkapan paket-paket yang masuk dengan aplikasi Wireshark. Data hasil pengujian ditunjukkan pada Tabel 4.10.

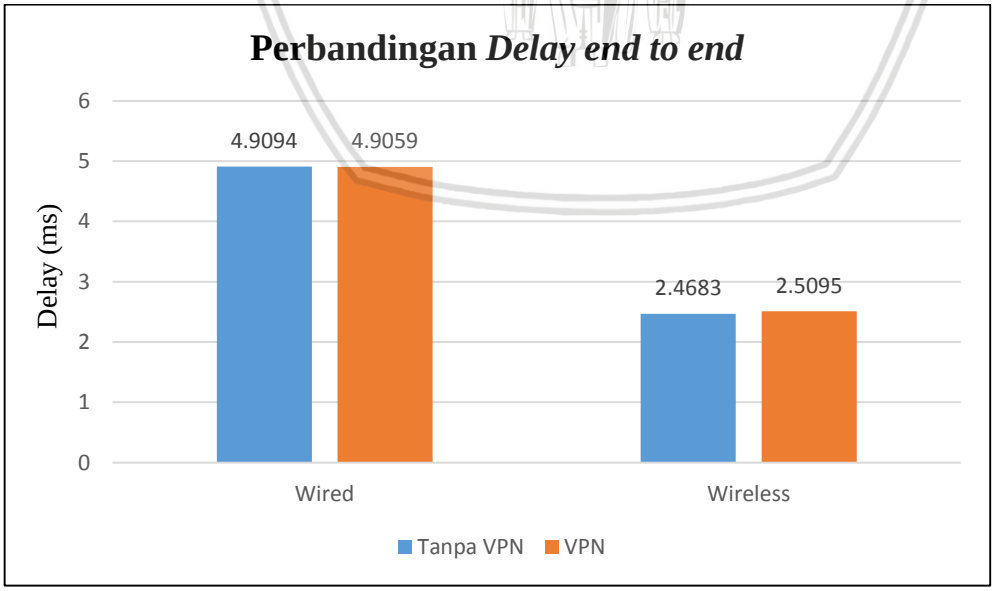
Tabel 4.10 Hasil Pengukuran *Delay end to end*

Pengujian	***Delay end to end* (ms)**			
	Wired		***Wireless***	
	Tanpa VPN	**VPN**	**Tanpa VPN**	**VPN**
1	4.9182	4.8897	2.4616	2.4682
2	4.9090	4.8357	2.4688	2.6903
3	4.8814	4.9377	2.4722	2.4665
4	4.9175	4.9299	2.4712	2.4635
5	4.9209	4.9365	2.4679	2.4590
Rata-rata	**4.9094**	**4.9059**	**2.4683**	**2.5095**

(**Sumber:** Simulasi)

Dari hasil perhitungan rata-rata *delay end to end* pada Tabel 4.10, dapat dilihat dari pengujian 1 sampai 5 nilai *delay* untuk jaringan tanpa VPN pada media *wired* dan *wireless* tidak mengalami perubahan yang signifikan dan cenderung stabil. *Delay* jaringan tanpa VPN dan VPN pada media *wired*, sedikit lebih besar dari *delay* pada media *wireless*.

Untuk diagram perbandingan nilai *delay end to end* pada dua model konfigurasi jaringan dan dua media transmisi berbeda dapat dilihat pada Gambar 4.5.



Gambar 4.5 Diagram perbandingan *Delay end to end*

Berdasarkan nilai indeks *delay* dalam standar TIPHON, untuk jaringan pada *wired* tanpa VPN dengan nilai rata-rata *delay* 4.9094 ms masuk dalam kategori “Sangat Bagus” dan

jaringan *wired* VPN dengan nilai rata-rata *delay* 4.9059 ms masuk dalam kategori "Sangat Bagus". Sedangkan untuk jaringan pada *wireless* tanpa VPN dengan nilai rata-rata *delay* 2.4683 ms masuk dalam kategori "Sangat Bagus" dan jaringan *wireless* VPN dengan nilai rata-rata *delay* 2.5095 ms masuk dalam kategori "Sangat Bagus". Dengan demikian dapat disimpulkan bahwa perbedaan konfigurasi keamanan jaringan dan media transmisi yang digunakan tidak terlalu berpengaruh karena sama-sama masuk dalam kategori "Sangat Bagus".

4.5.2 Analisa *Packet Loss*

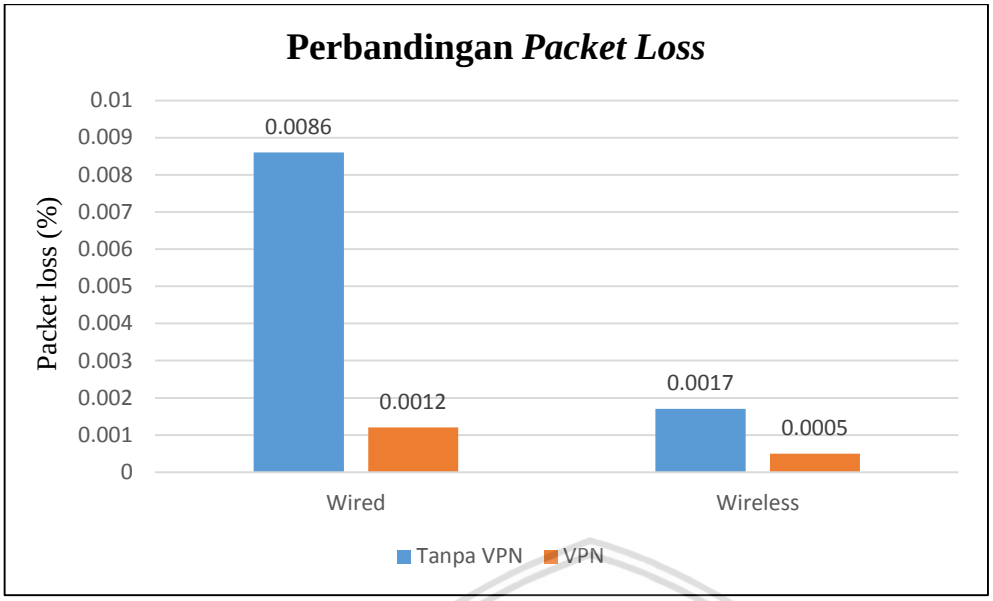
Packet Loss adalah banyaknya paket yang hilang pada suatu jaringan paket yang disebabkan oleh tabrakan (*collision*) dan penuhnya kapasitas jaringan. Pengambilan parameter *packet loss* dilakukan dengan cara *client* melakukan panggilan terhadap *client* lain. Kemudian disaat yang bersamaan pada salah satu *client* melakukan *capture* data atau penangkapan paket-paket yang masuk dengan aplikasi *Wireshark*. Data hasil pengujian parameter *packet loss* ditunjukkan pada Tabel 4.11.

Tabel 4.11 Hasil Pengukuran *Packet Loss*

Pengujian	***Packet Loss* (%)**			
	Wired		***Wireless***	
	Tanpa VPN	**VPN**	**Tanpa VPN**	**VPN**
1	0.0009	0.0027	0	0.0015
2	0	0	0.0017	0
3	0	0.0021	0.0041	0
4	0	0	0	0
5	0.0034	0.0008	0.0029	0,0010
Rata-rata	**0.0086**	**0.0012**	**0.0017**	**0.0005**

(**Sumber:** Simulasi)

Dari Tabel 4.11 berdasarkan perhitungan rata-rata persentase *packet loss* dapat dilihat untuk jaringan tanpa VPN ataupun jaringan VPN pada media *wired* dan *wireless* sama-sama memiliki nilai *packet loss* yang kecil, sehingga perbedaan konfigurasi jaringan dan media transmisi tidak terlalu berpengaruh pada besarnya nilai *packet loss*.



Gambar 4.6 Diagram perbandingan *Packet Loss*

Dari diagram di atas dapat dilihat bahwa ada perbedaan pada media *wired* dan *wireless*, namun sangat kecil. Berdasarkan nilai indeks *throughput* dalam standar TIPHON, untuk jaringan pada *wired* tanpa VPN dengan persentase nilai rata-rata *packet loss* 0.0086% masuk dalam kategori "Bagus" dan jaringan *wired* VPN dengan persentase nilai rata-rata *packet loss* 0.0012% masuk dalam kategori "Bagus". Sedangkan untuk jaringan pada *wireless* tanpa VPN dengan persentase nilai rata-rata *packet loss* 0.0017% masuk dalam kategori "Bagus" dan jaringan *wireless* VPN dengan persentase nilai rata-rata *packet loss* 0.0005% masuk dalam kategori "Bagus". Dengan demikian dapat disimpulkan bahwa perbedaan konfigurasi keamanan jaringan dan media transmisi yang digunakan tidak terlalu berpengaruh karena sama-sama masuk dalam kategori "Bagus".

4.5.3 Analisa *Throughput*

Throughput merupakan kecepatan transfer data rata-rata dari suksesnya paket yang dikirim per detiknya, pada umumnya menggunakan satuan *kilo bit per second* (kbps). Pengambilan parameter *throughput* dilakukan dengan cara *client* melakukan panggilan terhadap *client* lain. Kemudian disaat yang bersamaan pada salah satu *client* melakukan *capture* data atau penangkapan paket-paket yang masuk dengan aplikasi Wireshark. Data hasil pengujian parameter *throughput* ditunjukkan pada Tabel 4.12.

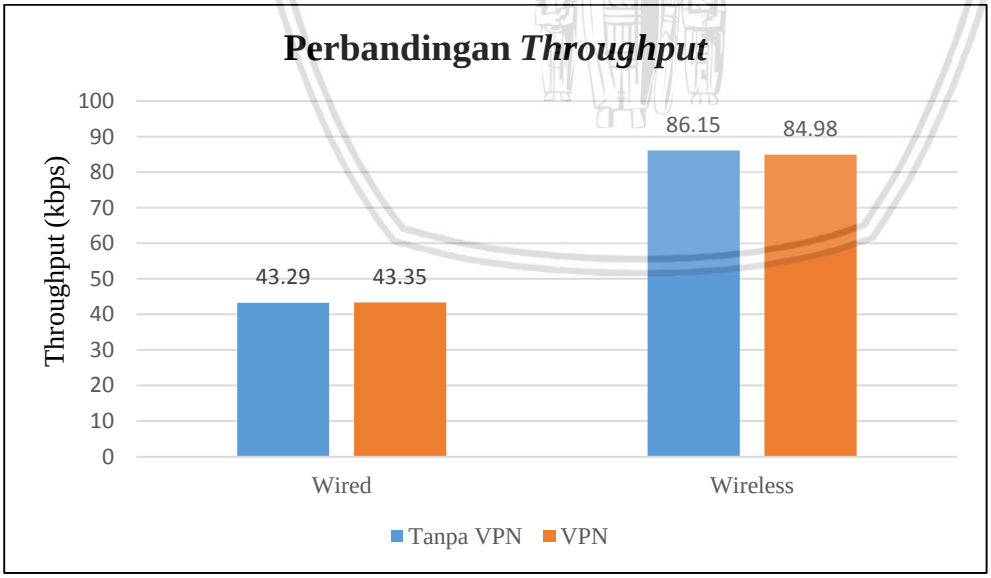
Tabel 4.12 Hasil Pengukuran *Throughput*

Pengujian	***Throughput* (kbps)**			
	Wired		***Wireless***	
	Tanpa VPN	**VPN**	**Tanpa VPN**	**VPN**
1	43.19	43.22	86.23	86.43
2	43.24	43.66	86.24	79.41
3	43.52	43.19	86.05	86.32
4	43.38	43.44	86.11	86.29
5	43.14	43.25	86.15	86.45
Rata-rata	**43.29**	**43.35**	**86.15**	**84.98**

(**Sumber:** Simulasi)

Berdasarkan hasil rata-rata pada Tabel 4.12, didapatkan bahwa terdapat pengaruh perbedaan media transmisi yang digunakan terhadap nilai *throughput*. Nilai *throughput* pada media *wireless* lebih besar dibandingkan nilai *throughput* pada media *wired*.

Untuk diagram perbandingan nilai *throughput* pada dua model konfigurasi jaringan dan dua media transmisi berbeda dapat dilihat pada Gambar 4.7.



Gambar 4.7 Diagram perbandingan *Throughput*

Berdasarkan nilai indeks *throughput* dalam standar TIPHON, untuk jaringan pada *wired* tanpa VPN dengan persentase nilai rata-rata *throughput* 43.29% masuk dalam kategori "Sedang" dan jaringan *wired* VPN dengan persentase nilai rata-rata *throughput* 43.35%

masuk dalam kategori "Sedang". Sedangkan untuk jaringan pada *wireless* tanpa VPN dengan persentase nilai rata-rata *throughput* 86.15% masuk dalam kategori "Sangat Bagus" dan jaringan *wireless* VPN dengan persentase nilai rata-rata *throughput* 84.98% masuk dalam kategori "Sangat Bagus". Dengan demikian dapat disimpulkan bahwa jaringan yang menggunakan media *wireless* dengan konfigurasi tanpa VPN memiliki nilai *throughput* yang lebih baik yaitu dengan kategori "Sangat Bagus".

4.6 Perbandingan Hasil Perhitungan dengan Simulasi

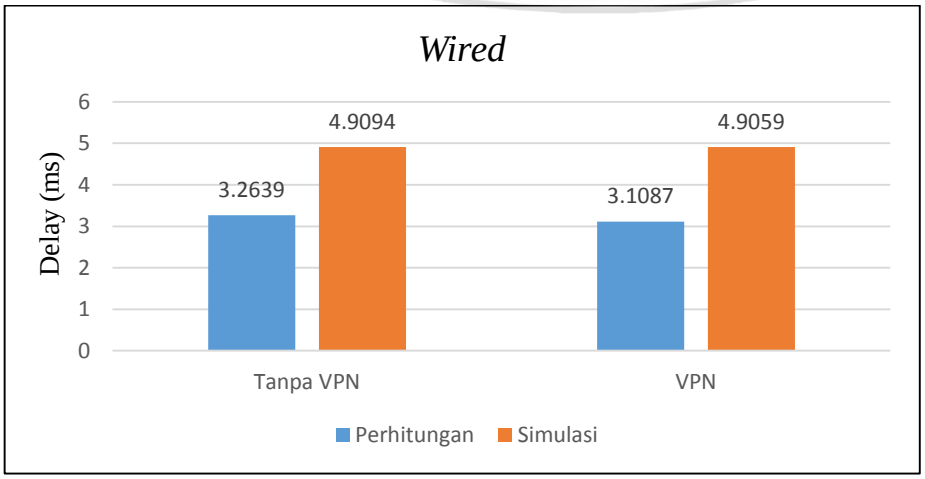
4.6.1 *Delay end to end*

Pada Tabel 4.11 dan Tabel 4.12 menunjukkan perbedaan besarnya nilai *delay end to end* antara hasil analisis perhitungan dengan simulasi. Perbedaan yang timbul antara hasil analisis perhitungan dengan hasil simulasi terjadi dikarenakan pada proses simulasi dilakukan pada rentang waktu simulasi tertentu, sehingga metode *error-control* yang terdapat pada jaringan LAN dapat terlihat, dimana metode tersebut tidak dapat terlihat jika analisis menggunakan perhitungan.

Tabel 4.13 Perbandingan nilai *delay end to end wired* perhitungan dan simulasi

***Delay end to end* (ms)**			
Wired			
Tanpa VPN		**VPN**	
Perhitungan	**Simulasi**	**Perhitungan**	**Simulasi**
3.2639	4.9094	3.1087	4.9059

(**Sumber:** Simulasi)

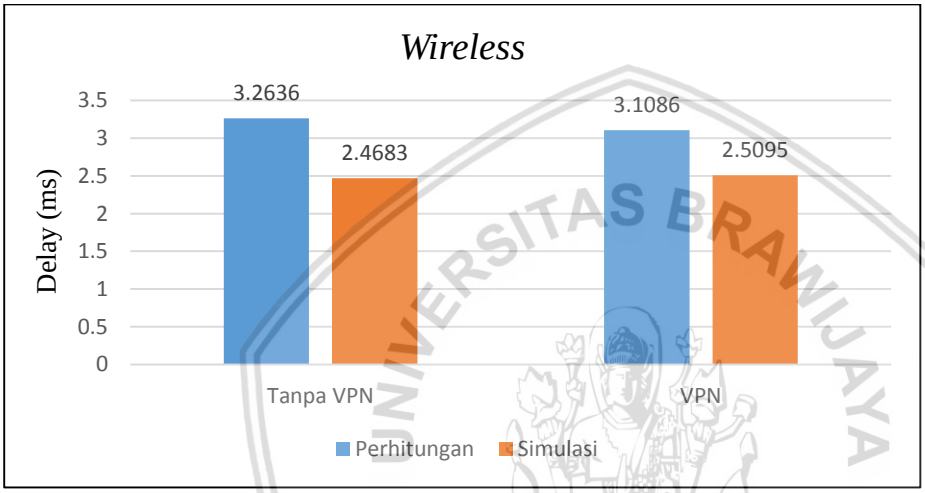


Gambar 4.8 Perbandingan *delay end to end wired* hasil perhitungan dan simulasi

Tabel 4.14 Perbandingan nilai *delay end to end wireless* perhitungan dan simulasi

***Delay end to end* (ms)**			
Wireless			
Tanpa VPN		**VPN**	
Perhitungan	**Simulasi**	**Perhitungan**	**Simulasi**
3.2636	2.4683	3.1086	2.5095

(**Sumber:** Simulasi)



Gambar 4.9 Perbandingan *delay end to end wireless* hasil perhitungan dan simulasi

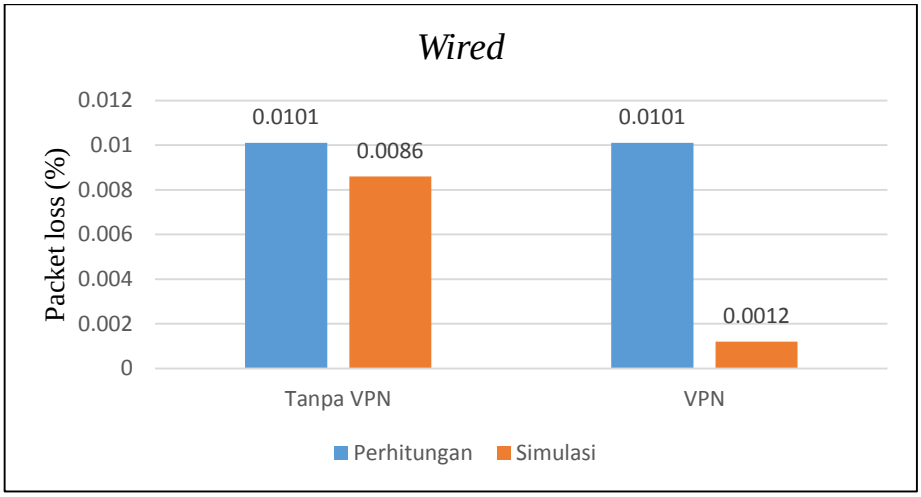
4.6.2 *Packet Loss*

Pada Tabel 4.15 dan Tabel 4.16 menunjukkan perbedaan besarnya nilai packet loss antara hasil analisis perhitungan dengan simulasi.

Tabel 4.15 Perbandingan nilai *packet loss wired* perhitungan dan simulasi

***Packet loss* (%)**			
Wired			
Tanpa VPN		**VPN**	
Perhitungan	**Simulasi**	**Perhitungan**	**Simulasi**
0.0101	0.0086	0.0101	0.0012

(**Sumber:** Simulasi)

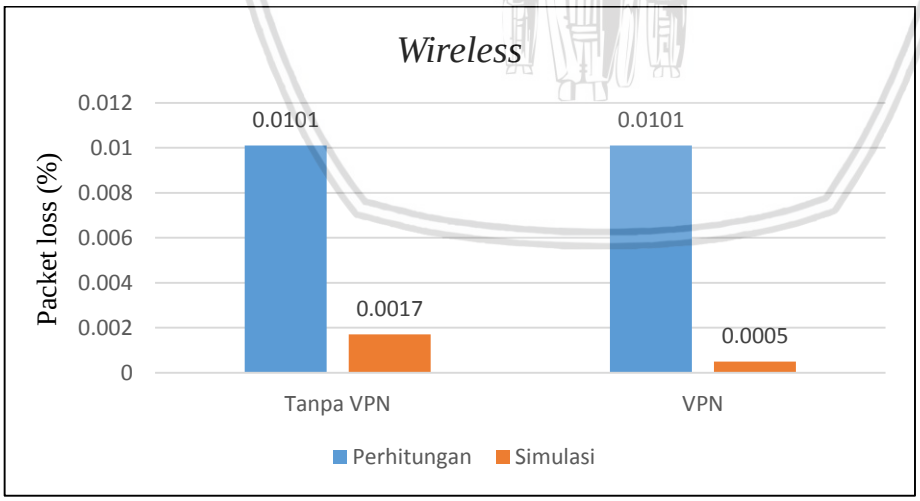


Gambar 4.10 Perbandingan *packet loss wired* hasil perhitungan dan simulasi

Tabel 4.16 Perbandingan nilai *packet loss wireless* perhitungan dan simulasi

***Packet loss* (%)**			
Wireless			
Tanpa VPN		**VPN**	
Perhitungan	**Simulasi**	**Perhitungan**	**Simulasi**
0.0101	0.0017	0.0101	0.0005

(**Sumber:** Simulasi)



Gambar 4.11 Perbandingan *packet loss wireless* hasil perhitungan dan simulasi

4.6.3 *Throughput*

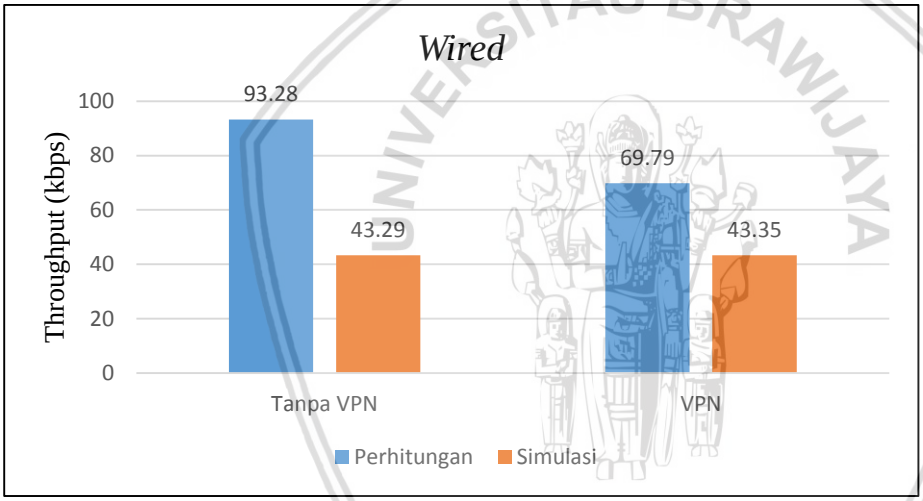
Pada Tabel 4.17 dan Tabel 4.18 menunjukkan perbedaan besarnya nilai throughput antara hasil analisis perhitungan dengan simulasi. Perbedaan yang timbul antara hasil

analisis perhitungan dengan hasil simulasi terjadi dikarenakan pada proses simulasi dilakukan pada rentang waktu simulasi tertentu, sehingga metode *error-control* yang terdapat pada jaringan LAN dapat terlihat, dimana metode tersebut tidak dapat terlihat jika analisis menggunakan perhitungan.

Tabel 4.17 Perbandingan nilai *throughput wired* perhitungan dan simulasi

***Throughput* (kbps)**			
Wired			
Tanpa VPN		**VPN**	
Perhitungan	**Simulasi**	**Perhitungan**	**Simulasi**
93.28	43.29	69.79	43.35

(**Sumber:** Simulasi)

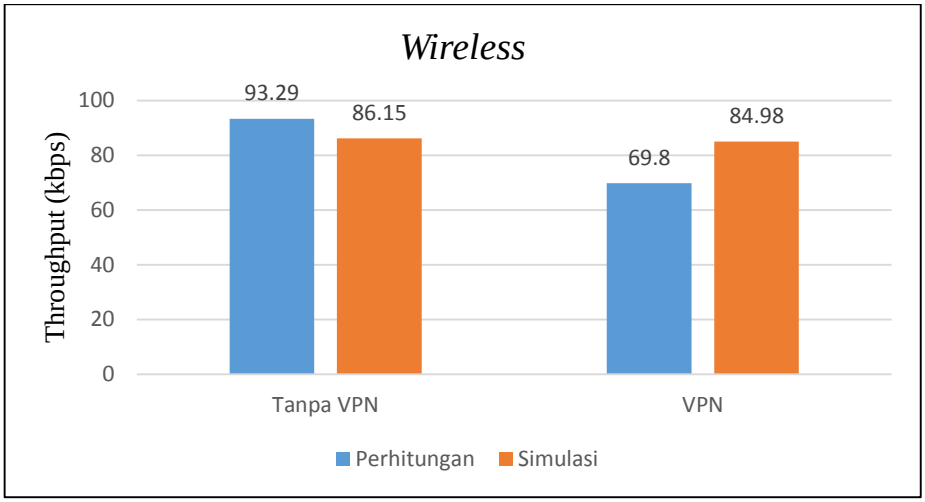


Gambar 4.12 Perbandingan *throughput wired* hasil perhitungan dan simulasi

Tabel 4.18 Perbandingan nilai *throughput wireless* perhitungan dan simulasi

***Throughput* (kbps)**			
Wireless			
Tanpa VPN		**VPN**	
Perhitungan	**Simulasi**	**Perhitungan**	**Simulasi**
93.29	86.15	69.80	84.98

(**Sumber:** Simulasi)



Gambar 4.13 Perbandingan *throughput wireless* hasil perhitungan dan simulasi

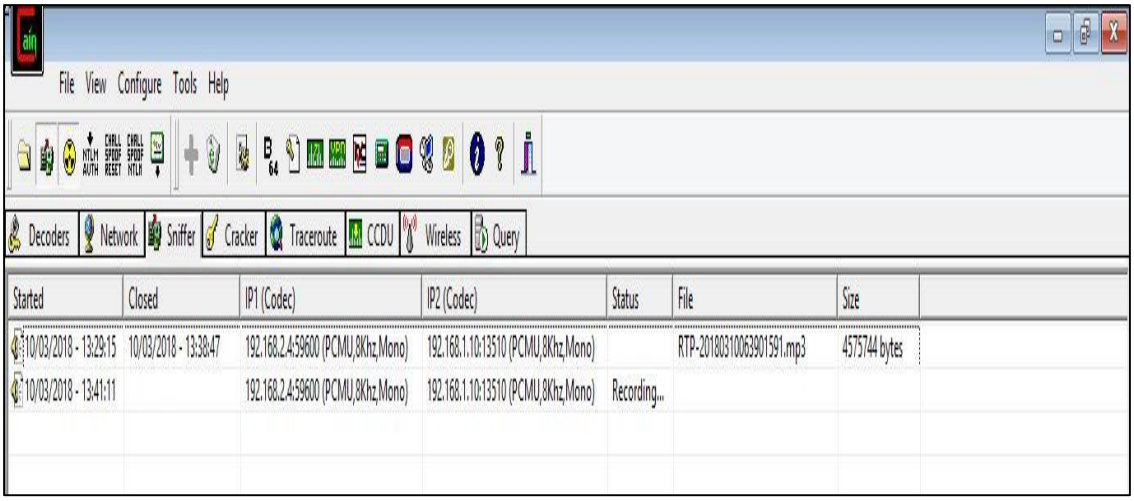
Dari hasil yang ditunjukkan Gambar 4.8, 4.9, 4.10, 4.11, 4.12, dan 4.13 dapat diketahui bahwa :

- Terdapat perbedaan nilai parameter pada hasil simulasi dengan hasil perhitungan karena faktor C (kecepatan proses) yang dimana kecepatan proses pada perhitungan digunakan asumsi standar yang sudah ada, sedangkan pada simulasi kecepatan proses yang terjadi sesuai aspek perangkat yang digunakan.
- Perbedaan data yang terjadi adalah karena pada sisi pengamatan Wireshark, menghitung QoS berdasarkan per paket data, sedangkan pada perhitungan berdasarkan keseluruhan paket data.

4.7 Pengujian Keamanan VoIP

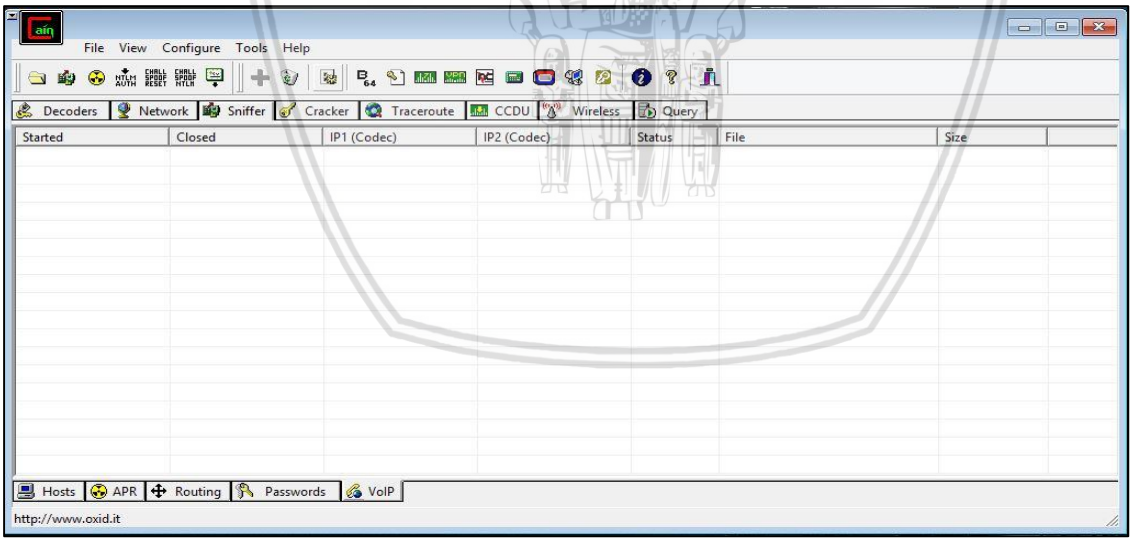
Pengujian keamanan VoIP pada penelitian ini dilakukan dengan teknik *sniffing*. *Sniffing* dilakukan tanpa merubah data atau paket apapun pada jaringan. *Sniffer* akan melakukan penyadapan terhadap komunikasi *client-server* yang sedang berjalan pada jaringan lokal, dimana pada saat *client* 2 sedang melakukan komunikasi dengan *client* 3 yang melewati *server* terlebih dahulu.

Proses *sniffing* dilakukan dengan menggunakan *software* Cain and Abel. Berikut adalah gambar hasil pengujian keamanan pada jaringan VoIP tanpa VPN dan dengan VPN.



Gambar 4.14 Sniffing VoIP tanpa VPN

Pada Gambar 4.14, *sniffer* dapat menyadap komunikasi antara *client* 2 dan *client* 3 pada jaringan VoIP tanpa keamanan VPN. Kelemahan komunikasi menggunakan VoIP adalah data *payload* tidak diproteksi sehingga ketika dikirimkan dan ditangkap maka akan dengan mudah data tersebut disadap pihak lain, ini terbukti setelah data *payload* VoIP disadap sehingga komunikasi antara *client* 2 dan *client* 3 dapat didengar kembali. Hal ini membuktikan bahwa komunikasi VoIP belum aman.



Gambar 4.15 Sniffing VoIP dengan VPN

Pada Gambar 4.15, jaringan VoIP telah diberikan konfigurasi keamanan dengan menggunakan VPN. Hasilnya *payload* tidak terdeteksi sehingga data *payload* tidak dapat disadap. Hasil penelitian ini menunjukkan bahwa komunikasi VoIP sangat mudah disadap tetapi dengan menambahkan konfigurasi VPN, komunikasi VoIP menjadi aman karena terdapat *tunnel* dan data dienkripsi sehingga pihak lain tidak dapat menyadapnya.

BAB V
PENUTUP

5.1 Kesimpulan

1. Kualitas performansi VoIP jaringan tanpa VPN dan VPN pada media *wired* dan *wireless* adalah sesuai dengan standar TIPHON untuk *delay end to end* masuk dalam indeks *delay* 0−150 ms atau "Sangat Bagus", kemudian untuk *packet loss* masuk dalam indeks 0−3 % atau "Bagus", dan *throughput* masuk dalam indeks 50-25 % atau "Sedang" untuk media *wired* sedangkan untuk media *wireless* 100−75 % atau "Sangat Bagus".
2. Kualitas layanan VoIP berdasarkan parameter *delay end to end, packet loss,* dan *throughput* memiliki perbedaan nilai untuk hasil simulasi dan hasil perhitungan. Perbedaan nilai parameter terjadi karena faktor kecepatan proses dari sisi simulasi dan perhitungan.
3. Dengan menggunakan VPN maka komunikasi akan lebih aman karena adanya autentikasi *server* dengan *client* dan paket yang melalui jaringan VPN akan di enkripsi sehingga proses komunikasi tidak dapat disadap.

5.2 Saran

1. Diharapkan untuk penelitian selanjutnya tidak hanya menggunakan PC / laptop namun juga digunakan sebuah *handphone* untuk menggunakan fasilitas VoIP ini.
2. Perlu dilakukan pengujian terhadap jumlah maksimum panggilan yang mampu diatur oleh *server* VoIP.

DAFTAR PUSTAKA

Amrulloh, Hafid. 2010. *Integrasi Jaringan VoIP dengan Jaringan PABX antara Kantor Cabang Surabaya dengan Kantor Pusat Jakarta PT. Wijaya Karya melalui VPN*. ITS.

Gonzalfes, Flavio E. 2007. *How to Build and Configure a PBX with Open Source Software*.

Lazuardi, Novri. 2008. *Perencanaan Jaringan Komunikasi VoIP (Voice Over Internet Protocol) menggunakan Asterisk SIP (Session Initiation Protocol)*. USU.

Markus, Fielner. 2006. *Open VPN Building and Integrating Virtual Network*. Packet Publishing Ltd, Birmingham.

Purbo, W. Onno & Raharja, Anton. 2010. *VoIP Cookbook Building your own Telecommunication Infrastructure*.

Sugeng, Winarno. 2008. *Membangun Telepon Berbasis VoIP*. Penerbit Informatika Bandung.

Schwartz, Mischa. 1987. *Computer-Communication Network Design and Analysis*. USA : Addison Wesley Pub.

Yuniati, Yetti. 2014. *Analisa Perancangan Sever VoIP (Voice Over Internet Protocol) dengan Open Source Asterisk dan VPN (Virtual Private Network) Sebagai Pengaman antar Client*. Universitas Lampung.

Zakaria, M. Isnan. 2015. *Analisis Keamanan dan Performansi VoIP berbasis GNU Linux Trixbox pada Jaringan Wifi*. UIN Sunan Kalijaga.

http://www.mikrotik.co.id/artikel_lihat.php?id=43 Diakses 2 Maret 2017

https://technet.microsoft.com/ptpt/library/cc779919(v=ws.10).aspx#w2k3tr_vpn_how_niuh Diakses 2 Maret 2017

https://www.cisco.com/c/en/us/support/docs/voice/voice-quality/7934-bwidth consume.html Diakses 3 Maret 2017

http://sourcedaddy.com/windows-xp/the-tcp-ip-protocol-framework.html Diakses 3 Maret 2017