II. TINJAUAN PUSTAKA

2.1 Kajian Literatur

Penulis [1] melakukan penyelidikan implementasi VoIP menggunakan *Open*IMS*Core* sebagai infrastruktur jaringannnya. Dimana penulis melakukan pengujian jaringan dengan melakukan perubahan *bandwidth* dan melihat performansi QoS VoIP, yaitu *delay*, *jitter* dan *packet loss* untuk mendapatkan *bandwidth* minimum sehingga performasi VoIP dapat berjalan dengan baik sesuai dengan standar. Dalam uji cobanya *client* menggunakan transmisi kabel LAN dalam mengakses layanan.

Penulis [2] melakukan penyelidikan performansi layanan pesan cepat pada jaringan implementasi *Open*IMS*Core*. Dalam penelitiannya penulis melakukan perubahan *bandwidth* menggunakan standar ITU-T seri V, IDSL, dan HDSL. Pada setiap *bandwidth*, parameter QoS *delay* diukur sebannyak tiga kali percobaan melalui SIP *MESSAGE* yang diperoleh. Hasil *delay* ketiga percobaan dirata-rata untuk melihat nilai *delay* pada *bandwidth* tersebut. Standar minimum fungsi IM pada V.32 (9,6 Kbps) sedangkan untuk performansi yang optimal didapatkan pada standar V.92 (56 Kpbs). Dalam uji cobanya *client* menggunakan transmisi kabel LAN dalam mengakses layanan.

2.2 IP *Multimedia Subsystem* (IMS)

IMS merupakan suatu teknologi komunikasi yang dapat menyatukan sistem komunikasi *wireless* dan *wired* dalam suatu jaringan yang *real time*, ekstensibel, dan dapat memberi layanan multimedia secara interaktif. IMS didesain sehingga mampu menyediakan layanan aplikasi *streaming* (suara, *video*, gambar) yang lebih kompetitif, memiliki mobilitas yang lebih besar, dan serta memiliki layanan yang lebih baik. IMS juga didesain untuk mampu bekerja tanpa dibatasi area maupun domain yang ada. Prinsip kerja jaringan IMS adalah menggunakan sesi untuk menangani setiap layanan yang diminta oleh pengguna [3].

2.2.1 Konsep IMS

IMS merupakan suatu konsep yang didesain dan ditentukan oleh 3GPP yang merupakan suatu persetujuan kerjasama dari berbagai macam badan standar yang dibentuk pada Desember 1998.

IMS meningkatkan kemampuan mode *packet switched* (PS) jaringan bergerak (seperti G.729, 3G) dengan mendukung jasa dan aplikasi berbasis IP melalui protokol *Session Initiation Protocol* (SIP). Secara efektif, IMS menyediakan suatu arsitektur pemersatu yang mendukung cakupan yang luas dari jasa berbasis IP di atas jaringan paket dan CS, memanfaatkan perbedaan teknologi akses *wireless* dan *fixed*.

IMS dirancang untuk menyediakan sejumlah kemampuan kunci yang diperlukan untuk memungkinkan jasa baru IP melalui jaringan bergerak. Bidang yang baru dari jasa IP ini harus mempertimbangkan kompleksitas multimedia, batasan

jaringan, pengaturan mobilitas dan pengaturan akan banyak munculnya aplikasi-aplikasi. Walaupun IMS telah dirancang untuk jaringan bergerak**,** namun dapat juga digunakan untuk menyediakan jasa untuk jaringan *fixed*.

IMS menggunakan protokol SIP untuk negosiasi sesi multimedia dan sesi manajemen. IMS adalah suatu jaringan SIP bergerak penting, yang dirancang untuk mendukung fungsionalitas jaringan bergerak. Di dalamnya tersedia *routing*, lokasi jaringan, dan fungsi-sungsi pengalamatan. Berbeda dengan domain CS dan PS, domain IMS memungkinkan jenis sesi media apapun untuk dibuat. Dan juga mengijinkan penyedia jasa untuk melakukan kemampuan mengkombinasikan jasa dari domain CS dan PS di sesi yang sama. Kemampuan ini membuka sejumlah jasa yang baru dan inovatif untuk *user-touser* dan *multi-user* seperti peningkatan jasa suara, *video telephony*, *chat*, *push to talk* dan konferensi multimedia, semua ini didasarkan pada konsep suatu sesi multimedia.

- *Mobility Management*

Infrastruktur IMS memungkinkan jasa sebuah perangkat komunikasi bergerak IP untuk menemukan pengguna lain di jaringan dan kemudian melakukan sesi dengan pengguna tersebut. Komponen kunci IMS yang memungkinkan manajemen mobilitas adalah *Call Session Control Function* (CSCF) dan *Home Subscriber Service* (HSS). HSS memegang semua data pelanggan dan memperbolehkan pemakai untuk menemukan dan berkomunikasi dengan *end user* yang lain. Fungsi utama CSCF adalah sebagai *proxy*, yang membantu *setup*, mengatur sesi dan meneruskan pesan ke jaringan IMS.

- *Quality of Service (QoS)*

IMS menyediakan suatu solusi yang efektif dan distandardisasi untuk operator yang ingin menerapkan jasa IP *mobile* yang *real-time* tanpa bergantung pada kerja transmisi yang baik dan menghasilkan ketidakpuasan pelanggan. Komunikasi IP *mobile* yang *real-time* itu sulit karena adanya *bandwidth* yang berubah-ubah, yang mempengaruhi transmisi paket IP melalui jaringan. Mekanisme *Quality of Service* (QoS) telah dikembangkan dalam rangka mengalahkan isu itu dan menyediakan beberapa bentuk tingkat jaminan transmisi sebagai pengganti '*best effort'*. QoS memastikan bahwa unsur-unsur kritis dari transmisi IP seperti *transmision rate*, g*ateway delay* dan *error rate* dapat diukur, ditingkatkan dan dijamin. Para pemakai bisa menetapkan tingkatan mutu yang mereka perlukan tergantung pada jenis jasa dan keadaan pemakai.

- Eksekusi Layanan, Pengendalian , dan Interaksi

Di suatu susunan jasa bergerak yang kompleks dimana operator telah membuat sejumlah besar jasa, tentu saja multak bagi operator untuk bisa mengendalikan jasa dan interaksi antar berbagai komponen layanan. IMS menyediakan solusi bagi tantangan ini dengan menyediakan jasa yang efisien, sesuai dengan kemampuan yang ditetapkan.

- Pihak ketiga pembuat *Interfaces*

IMS menyediakan arsitektur yang distandardisasi untuk memungkinkan kelanjutan dari pengembangan jasa IP. Berbagai jasa IMS dapat dikembangkan dengan tidak tergantung dan pada waktu yang sama menggunakan fitur umum dari infrastruktur IMS [4].

2.2.2 Arsitektur IMS

Lapisan jaringan IMS terbagi menjadi 3, yaitu *Layer Server* Aplikasi (menyediakan *end user logic*), *Layer Session Control* (terdapat CSCF yang mengatur sesi registrasi hingga komunikasi data), *Layer Transport* dan *Endpoint* (untuk menginisiasi dan mengakhiri pensinyalan SIP). IMS mampu menanggulangi inefisiensi *softswitch* dengan cara membangkitkan multi layanan dalam satu session. Yang berperan sentral dalam hal ini adalah protokol SIP dengan 3 *server* berbeda : S-CSCF, P-CSCF, dan ICSCF. Masing-masing CSCF memiliki tugas yang berbeda-beda. Secara umum semua jenis CSCF memiliki peran selama sesi registrasi dan sesi pembentukan. Gambar 2.1 menunjukan arsitektur IP *Multimedia Subsistem*.

1. *Proxy*-CSCF

P-CSCF merupakan titik awal dari pengguna dalam IMS. Hal ini berarti semua trafik pensinyalan SIP dari UE akan dikirimkan ke P-CSCF. Demikian pula dengan pengakhiran pensinyalan SIP dari jaringan dikirimkan melalui P-CSCF ke UE. P-CSCF bertanggung jawab untuk menjaga rahasia keamanan untuk pensinyalan SIP. Hal ini dilakukan selama proses registrasi SIP seperti UE dan P-CSCF melakukan negosiasi IPSec.

2. *Interrogating*-CSCF

I-CSCF merupakan titik kontak dalam sebuah jaringan operator untuk semua koneksi yang ditentukan untuk pelanggan dari jaringan tersebut.

Terdapat beberapa tugas dari I-CSCF diantaranya:

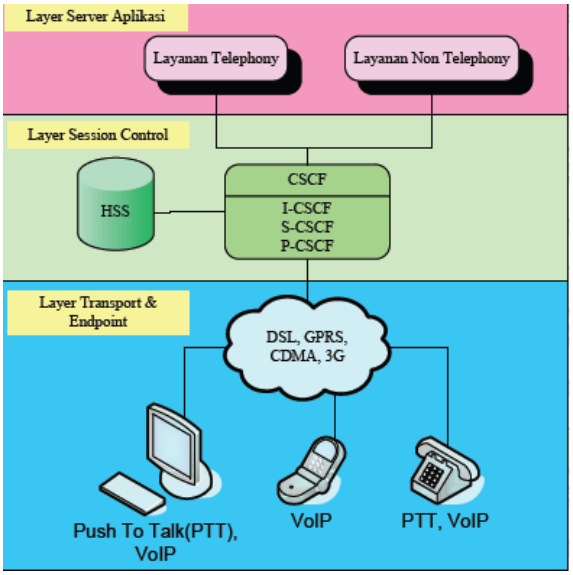
a. Memperoleh nama tujuan selanjutnya dari *Home Subscriber Server* (HSS)

b. Menentukan S-CSCF berdasarkan kemampuan yang diterima dari HSS.

c. Merutekan permintaan yang masuk untuk menentukan S-CSCF atau application *server*.

3. *Serving*-CSCF

S-CSCF merupakan titik fokus dari IMS yang bertanggungjawab untuk menangani proses registrasi, membuat keputusan perutean, menjaga sesi, dan menyimpan profile layanan. Ketika seorang *user* mengirimkan permintaan registrasi, hal tersebut akan di teruskan ke S-CSCF, S-CSCF akan mengunduh data otentifikasi dari HSS. Berdasarkan data otentifikasi yang diberikan HSS, dilakukan pembangkitan pertukaran informasi dengan UE. Setelah menerima respon dan memverifikasi, S-CSCF menerima registrasi dan memulai pengawasan status registrasi. Setelah proses tersebut berlangsungpengguna dapat memulai dan menerima layanan dari IMS. Selain itu S-CSCF mengunduh profile layanan dari HSS sebagai bagian dari proses registrasi.

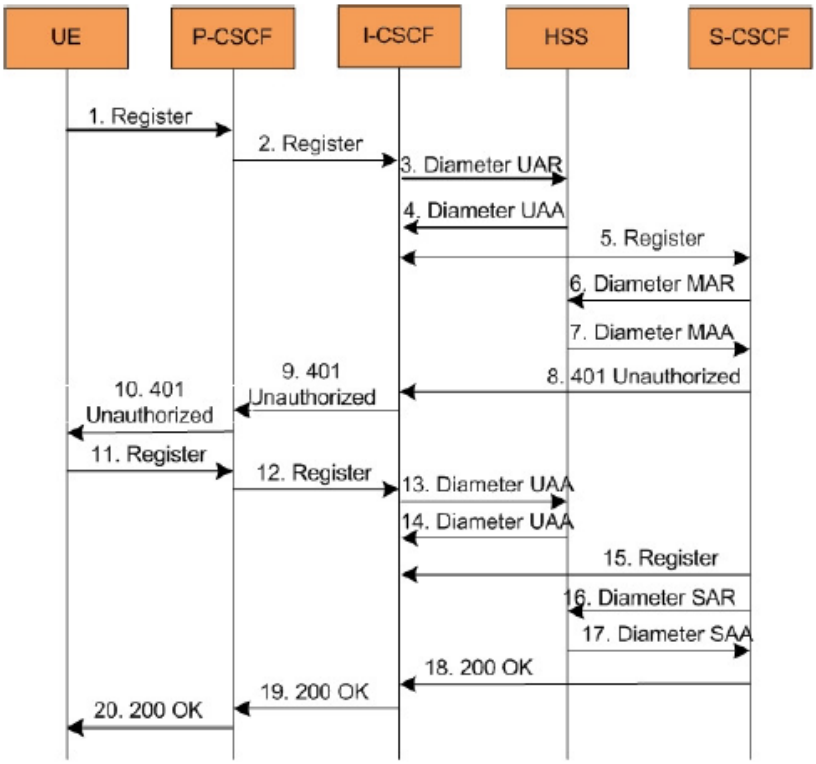


Gambar 2.1. Arsitektur IP Multimedia Subsistem [3]

2.2.3 Pensinyalan Pada IMS

SIP digunakan sebagai protokol pensinyalan dalam lingkungan IMS. Protokol SIP didefinisikan dalam RFC 3261 [5] yang memiliki fungsi registrasi, pembangunan sesi, manajemen sesi, dan mengatur partisipan. termasuk menciptakan, memodifikasi, dan mengakhiri sesi dengan satu atau lebih partisipan.

Pensinyalan SIP merupakan metode utama yang digunakan untuk registrasi pengguna dan sesi kontrol dalam arsitektur IMS. CSCF merupakan sever inti pensinyalan dalam arsitektur IMS. keduanya bekerja sebagai SIP *registrar* dan *stateful* SIP *proxy server*. Gambar 2.2 menggambarkan prosedur pensinyalan dalam inti IMS.



Gambar 2.2. Alur Pensinyalan Pesan Registrasi [6]

Prosedur pensinyalan dimulai dengan SIP REGISTER pengguna meminta untuk dikirimkan ke P-CSCF. Karena *bandwidth* antarmuka udara yang terbatas, pesan dikompresi sebelum dikirim oleh pengguna dan didekompresi di P-CSCF. Jika beberapa S-CSCF ada di *home network* pengguna, sebuah I-CSCF perlu disebarkan untuk memilih S-CSCF yang melayani sesi pengguna. Dalam hal ini, P-CSCF menyelesaikan pengalamatan dari *home* I-CSCF pengguna dengan menggunakan *home domain name* pengguna dan meneruskan REGISTER ke I-CSCF [7].

Setelah I-CSCF mengirimkan *User Authorization Request* (UAR) ke HSS, yang mengembalikan alamat S-CSCF, lalu I-CSCF memilih satu S-CSCF dan meneruskan pesan REGISTER. S-CSCF mengirimkan sebuah pesan *Multimedia Authentication Request* (MAR) ke HSS untuk mengunduh data otentifikasi pengguna. S-CSCF juga menyimpan *Uniform Resource Indicator* (URI) di dalam HSS. kemudian HSS memberi respon pesan *Multimedia Authentication Answer* (MAA). S-CSCF membentuk sebuah respon SIP 401*Unauthorised* dengan sebuah pertanyaan yang harus dijawab oleh UE. Setelah pertanyaan dijawab, maka otentifikasi berhasil. S-CSCF mengirimkan sebuah *Server Assignment Request* (SAR) untuk memberitahu HSS bahwa pengguna telah terdaftar dan HSS dapat mengunduh profil pengguna. HSS memberi balasan dengan jawaban *Server Assignment Answer* (SAA). Akhirnya, S-CSCF mengirimkan 200 pesan OK untuk menginformasikan UE proses registrasi sukses.

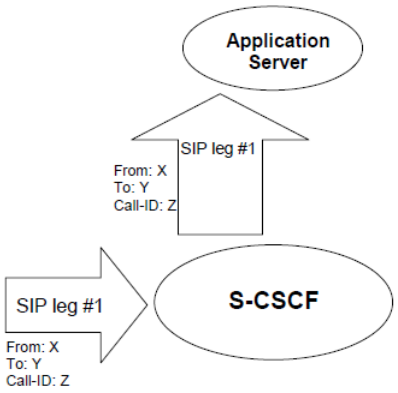
2.2.4 Layanan IMS

IMS memiliki beberapa layanan yang dapat digunakan diantaranya [4]:

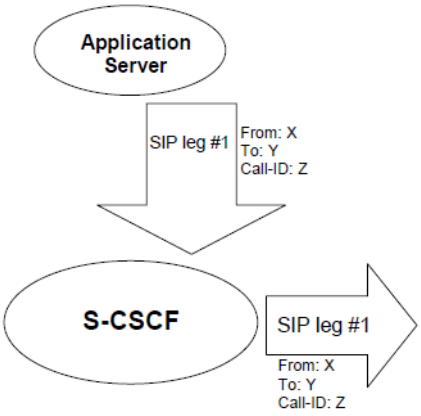
1. *Push to Talk over Cellular (PoC)*
2. *Real Time Video Sharing*
3. Aplikasi Interaktif
4. Layanan Pesan Cepat
5. Pesan Suara
6. *IMS enabled Voice and Video Telephony*, IMS memungkinkan panggilan suara dan video dibawa ke suatu jaringan inti paket (VoIP).
7. *Video-conferencing*

2.2.5 *Application Server* (AS)

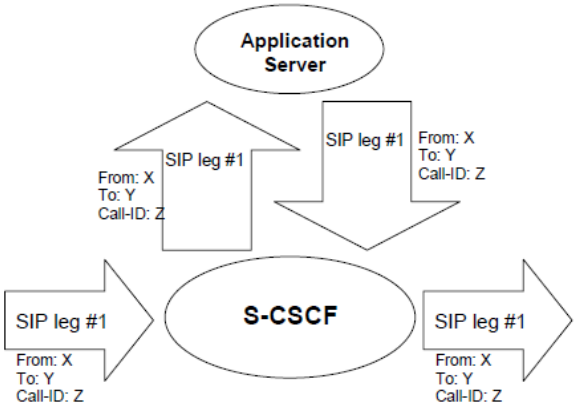
Application Server (AS) merupakan sebuah *server* SIP yang menjalankan layanan pada IMS. AS dapat bertindak sebagai *SIP User Agent Client* (UAC), *Sip User Agent Server* (UAS) atau SIP *Back to Back User Agent* (B2BUA). AS menjadi antarmuka S-CSCF yang meneruskan permintaan ke AS spesifik. Apabila terdapat beberapa AS dalam IMS, HSS berisi kriteria filter yang akan memilih AS berdasarkan konten tertentu dari permintaan yang masuk [8]. Antarmuka dari S-CSCF ke AS disebut IMS *Centralised Services* (ICs) dan didefinisikan dalam 3GPP TS.23.228 [9]. Gambar 2.3 sampai 2.7 menunjukan peran yang berbeda dari sebuah AS yang terjadi dan hasil yang berbeda dalam pensinyalan IMS [10].



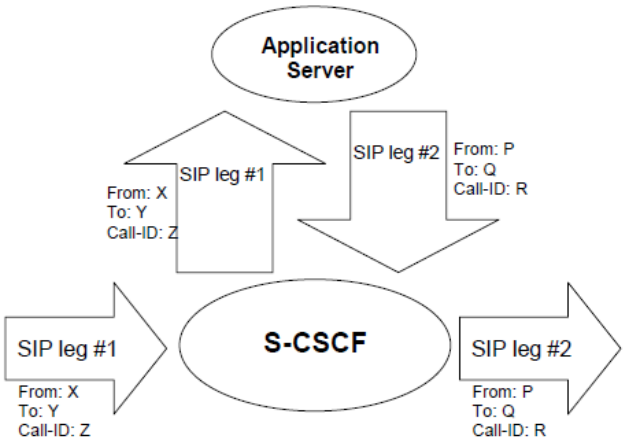
Gambar 2.3. *Application Server* Bertindak Sebagai *Terminating* UA atau *Redirected Server*



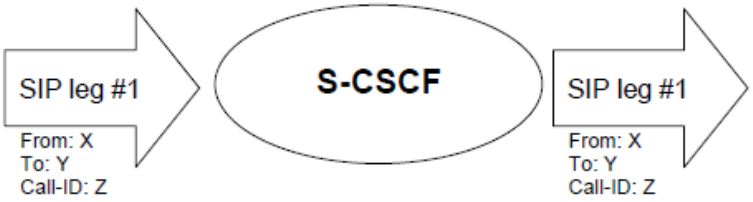
Gamba 2.4. *Application Server* Bertindak Sebagai *Originating* UA



Gambar 2.5. *Application Server* Bertindak Sebagai SIP *Proxy*



Gambar 2.6. *Application Server* Melakukan Kontrol *Third Party Call* yang Bertindak Menginisiasi B2BUA



Gambar 2.7. Sebuah SIP melewati S-CSCF tanpa *Application Server*

2.2.6. Home Subscriber Server (HSS)

Home Subscriber Server (HSS) adalah database pengguna yang menyimpan profil pengguna, dan menunjukkan keaslian dan kewenangan pengguna termasuk identifikasi pengguna, kontrol informasi, dan lokasi pengguna. Bagian ini serupa dengan *Home Location Register* (HLR) dan *Autenthication Center* (AUC) pada *Code Division Multiple Access* (CDMA) maupun *Global System for Mobile* (GSM) [11].

HSS dalam peran dasarnya merupakan sebuah database terpusat untuk entitas IMS, PS, dan CS. HSS merupakan entitas jaringan inti yang berperan sangat penting dalam melakukan otentifikasi pengguna, otorisasi dan manajemen sesi. HSS bersama dengan CSCF melengkapi fungsi *Control Layer* pada IMS untuk berlangganan dan manajemen sesi. HSS mendukung antarmuka yang berbeda dari jaringan IMS, jaringan PS / GPRS, jaringan CS/G.729, dan jaringan IP pada umumnya.

2.3 *Session Initation Protocol* (SIP)

Session Initation Protocol (SIP) merupakan sebuah *application–layer control protocol* yang dapat membentuk, memodifikasi, dan mengakhiri sesi multimedia seperti panggilan telepon internet [12].

2.3.1. Arsitektur Komponen SIP

Secara umum jaringan SIP terdiri dari dua komponen dasar, yaitu SIP *user agent* dan SIP *network server*. SIP *user* agent merupakan komponen dari *client* yang memulai dan menjawab panggilan.

Arsitektur SIP terdiri dari beberapa elemen fungsional, diantaranya:

1. *User* Agent

SIP *user* agent (UA) merupakan sebuah perangkat akhir yang dapat memulai dan menerima panggilan SIP. UA bisa berupa laptop, telepon genggam atau sebuah perangkat akhir yang dapat digunakan sebagai mesin penjawab. SIP mendukung arsitektur *peer-to-peer* dan *client server*. UA bertindak sebagai *client*, dimana

antara *client* satu dan yang lainnya dapat saling berhubungan satu sama lain dan dapat melakukan suatu sesi.

2. *User* Agent *Server*

Dalam model *client server*, ketika mengirim permintaan atau menerima tanggapan, SIP UA bertindak sebagai *client*, disebut dengan UAC. Penerima SIP UA bertindak sebagai *server* (menerima permintaan dan mengirimkan tanggapan) disebut dengan UAS. UAC dan UAS merupakan sebuah entitas yang terdapat pada setiap SIP *user* agent.

3. *Back* to *Back User* Agent (B2BUA)

Ketika entitas SIP bertindak sebagai UAC dan UAS maka entitas SIP tersebut dapat di sebut sebagai *Back*-to-*Back User* Agent (B2BUA). B2BUA memproses permintaan untuk memutuskan bagaimana permintaan panggilan akan dijawab.

4. Proxy *Server*

SIP proxy *server* merupakan komponen utama dalam infrastruktur SIP. Merupakan entitas pertama yang menerima semua permintaan panggilan keluar dari SIP UA. Merutekan permintaan yang melintas dan menempatkan permintaan ke *server* terdekat dengan tujuan SIP UA. Umumnya terdapat dua SIP proxy *server*, satu berada disisi pemanggil dan lainnya berada di sisi penerima panggilan.

5. Registrar

Registrar SIP merupakan tempat penyimpanan informasi lokasi dari UA. Registrar menerima permintaan registrasi dari UA dan menempatkan informasi tersebut dalam lokasi penyimpanan data. Ketika lokasi atau perangkat yang di gunakan

UA berubah, UA akan mengirimkan pesan permintaan SIP kepada registrar. SIP proxy *server* juga memanfaatkan informasi lokasi yang tersimpan dalam registrar untuk mengetahui lokasi dari UA.

6. Redirect *Server*

Server redirect merespon permintaan SIP dengan sebuah alamat dimana pesan SIP akan diarahkan. *Server* redirect memetakan alamat tujuan ke satu atau lebih alamat [13].

2.3.2. Pesan SIP

Pesan yang terdapat dalam SIP terbagi dalam dua format, diantaranya:

1. *Request*, merupakan pesan yang dikirim dari *client* ke *server*, dimana *request* berisi mengenai operasi perminataan yang ingin dilakukan oleh *client.* Pesan *request* dari SIP dibagi menjadi enam jenis, diantaranya:
   a. *INVITE* : menunjukan bahwa *user* atau layanan sedang diundang untuk bergabung dalam suatu sesi.
   b. ACK: menunjukan bahwa *client* telah menerima respon akhir dari suatu *request INVITE*.
   c. OPTION: digunakan untuk menanyakan suatu *server* mengenai kemampuan yang dimilikinya.
   d. BYE: menunjukan bahwa *client* mengindikasikan *server* jika sesi akan segera diakhiri.
   e. CANCEL: menunjukan bahwa *request* yang dberikan akan dibatalkan.
   f. REGISTER: menunjukan bahwa *client* mendaftarkan informasi kontak.

2. *Response*, merupakan pesan yang dikirim dari *server* ke *client*, dimana *response* berisi status balasan dari operasi permintaan yang ingin dilakukan oleh *client*. Pesan *response* berisi kode status dan keterangan mengenai kondisi dari pesan *request* yang diberikan. Terdapat tiga digit angka kode status dalam pesan *response* SIP. Terdapat enam jenis pesan *response* SIP, diantaranya [5]:
   a. 1xx: *Provisional*, pesan telah berhasil diterima dan melanjutkan proses dari *request* yang diberikan.
   b. 2xx: *Success*, tindakan yang ingin di lakukan telah berhasil diterima, dimengerti, dan disetujui.
   c. 3xx: *Redirection*, tindakan lebih lanjut perlu dikakukan untuk melengkapi *request* yang diberikan.
   d. 4xx: *Client Error*, *request* berisi sintak yang salah sehingga tidak dapat dikenali oleh *server*, dan *server* tidak dapat memprosesnya.
   e. 5xx: *Server Error*, *server* gagal untuk memproses permintaan yang sah, dalam hal ini terdapat masalah pada *server*.
   f. 6xx, *Global Failure*, *request* tidak dapat di lakukan pada *server* manapun.

2.4. Jaringan *Wireless*

Teknologi utama yang digunakan untuk membangun sebuah jaringan nirkabel murah yang saat ini banyak digunakan adalah protokol 802.11, yang dikenal oleh banyak kalangan sebagai Wi-Fi.

Terdapat banyak protokol dalam standar 802.11, dan tidak semua berhubungan secara langsung dengan protokol radio itu sendiri. Terdapat tiga standar jaringan nirkabel yang saat ini banyak digunakan dan mudah diaplikasikan, diantaranya:

a. 802.11 b, disahkan oleh IEEE pada tanggal 16 September 1999. Terdapat jutaan perangkat yang mendukung 802.11b sejak dikeluarkan tahun 1999. 802.11b menggunakan mekanisme transmisi yang disebut sebagai *Direct Sequence Spread Spectrum* (DSSS). Memiliki kecepatan maksimum 11 Mbps, dengan kecepatan penggunaan data aktual hingga 5 Mbps.
b. 802.11g merupakan protokol jaringan nirkabel yang datang belakangan dipasaran jaringan nirkabel karena belum disahkan hingga bulan juni 2013. Walaupun protokol 802.11g datang terlambat, 802.11g merupakan protokol jaringan nirkabel yang banyak digunakan sebagai fitur standar pada hampir semua laptop dan telepon gengam. Menggunakan mekanisme transmisi *Orthogonal Frequency Division Multiplexing* (OFDM). Memiliki kecepatan data maksimum 54 Mbps dan bisa turun menjadi 11 Mbps.
c. 802.11 a merupakan protokol jaringan nirkabel yang juga disahkan pada tanggal 16 September 1999, 802.11 a menggunakan modulasi OFDM. Protokol ini memiliki kecepatan data maksimum 54 Mbps, dengan aktual *throughput* sampai dengan 27 Mbps. Protokol 802.11a beroperasi pada pita *industrial, scientific and medical* (ISM) antara 5.745 dan 5.805 Ghz dan pada pita *Unlicensed National Information Infrastructure* (UNII) antara 5.150 dan 5.320 Ghz. Hal ini menyebabkan protokol 802.11a tidak cocok dengan 802.11b atau 802.11g, dan dengan frekuensi yang lebih tinggi berarti memiliki jangkauan yang lebih pendek dibandingkan dengan 802.11b/g dengan daya pancar yang sama [14].

2.5. *Internet* Protocol (IP)

Internet Protocol (IP) merupakan protokol *network layer* yang berisi informasi pengalamatan dan beberapa informasi kendali untuk memungkinkan paket di salurkan dalam sebuah jaringan. IP merupakan sebuah protokol lapisan jaringan utama dalam TCP/IP. Seiring dengan TCP, IP mewakili inti dari protokol internet. IP bekerja dengan baik pada jaringan komunikasi LAN maupun WAN.

IP memiliki dua tanggungjawab utama, yaitu menyediakan mode hubungan *connectionless*, mengirimkan data melalui sebuah jaringan, dan menyediakan fragmentasi dan *reassembly* dari datagram untuk mendukung data link dengan ukuran *maximum-transmission unit* (MTU) yang berbeda [15].

2.6. Real Time Protocol

Protokol RTP merupakan suatu protokol yang berfungsi untuk membawa suatu data *real time* baik yang dikirimkan secara *unicast* ataupun *multicast* melalui jaringan berbasis IP [16]. RTP didesain untuk membawa informasi waktu bersama dengan data yang dikirimkan sehingga RTP digunakan untuk membawa data *audio* maupun *video streams*. RTP terdiri dari suatu data dan bagian kendali yang disebut dengan *Real Time Control Protocol* (RTCP) [17].

2.7. *Quality of Service* (QoS)

Quality of Service menunjukkan kemampuan sebuah jaringan untuk menyediakan layanan yang lebih baik lagi bagi layanan trafik yang melewatinya. QoS

merupakan sebuah *system* arsitektur *end to end* dan bukan merupakan sebuah *feature* yang dimiliki oleh jaringan.

QoS suatu *network* merujuk ke tingkat kecepatan dan keandalan penyampaian berbagai jenis beban data di dalam suatu komunikasi. Dalam QoS meliputi beberapa hal yaitu: *throughput*, *delay*, *jitter*, *delta max* (*latency*) dan *packet Loss*.

Throughput

Throughput adalah kemampuan sebenarnya suatu jaringan dalam melakukan pengiriman data. Biasanya *throughput* dikaitkan dengan *bandwidth* karena *throughput* memang bisa disebut dengan *bandwidth* dalam kondisi yang sebenarnya. Sementara *throughput* sifatnya adalah dinamis tergantung trafik yang sedang terjadi. Semakin besar *bit-rate* maka akan semakin besar pula *throughput* nya, semakin besar nilai *throughput* nya akan menunjukkan semakin bagus pula kemampuan jaringan dalam mentransmisikan *file*.

$$Throughput\ (Kbps) = \frac{Jumlah\ paket\ diterima}{Waktu\ simulasi} \times besar\ 1\ paket\ data \quad ....(2.1)$$

Delay

Delay (waktu tunda) merupakan *interval* waktu yang dibutuhkan oleh suatu paket data saat data mulai dikirim dan keluar dari proses antrian dari titik sumber awal hingga mencapai titik tujuan.

$$Delay\ (seconds) = Waktu\ paket\ akhir - Waktu\ paket\ awal \quad ...(2.2)$$

Jitter

Jitter merupakan variasi *delay* antar paket yang terjadi pada jaringan IP. Besarnya nilai *jitter* akan sangat dipengaruhi oleh variasi beban trafik dan besarnya tabrakan antar paket (*congestion*) yang ada dalam jaringan IP. Semakin besar beban trafik

di dalam jaringan akan menyebabkan semakin besar pula peluang terjadinya *congestion* dengan demikian nilai *jitter*-nya akan semakin besar. Semakin besar nilai *jitter* akan mengakibatkan nilai QoS akan semakin turun. Untuk mendapatkan nilai QoS jaringan yang baik, nilai *jitter* harus dijaga seminimum mungkin. Jitter pada paket RTP dapat dihitung menggunakan perhitungan jitter yang dideskripsikan dalam RFC 3550 [18].

Perkiraan dari variansi satistik pada waktu antarkedatangan paket data RTP, diukur dalam satuan *timestamp* dan dinyatakan sebagai sebuah bilangan bulat. *Interarrival jitter* (J) didefinisikan sebagai rata-rata deviasi dari selisih (D) jarak paket dari pengirim dibandingkan dengan pengirim.

$$D(i,j) = (Rj - Ri) - (Sj - Si) = (Rj - Sj) - (Ri - Si) \ldots\ldots \quad (2.3)$$

Si,Sj = RTP *timestamp* dari paket i dan j,

Ri = waktu kedatangan dalam RTP *timestamp* dari paket i dan j

Persamaan 2.3 menunjukan perhitungan selisih (D) dari dua paket i dan j. Karena paket RTP datang secara kontinyu pada penerima maka selisih (D) dari pasangan paket (paket n dan paket n-1) juga dihitung secara kontinyu. Sehingga *interarrival jitter* dapat dihitung dengan persamaan:

$$J(i) = J(i-1) + \frac{(|D(i-1,i)| - j(i-1))}{16} \ldots. \quad (2.4)$$

Packet Loss

Packet loss dapat disebabkan oleh sejumlah faktor, mencakup penurunan *signal* dalam media jaringan, melebihi batas saturasi jaringan, paket yang *corrupt* dan kesalahan *hardware* jaringan. Dalam hal kerugian paket, penerima akan meminta *retransmission* atau pengiriman secara otomatis *resends* walaupun segmen telah

tidak diakui. Walaupun TCP dapat memulihkan dari kerugian paket, *retransmitting* paket yang hilang menyebabkan *throughput* yang berhubungan dengan koneksi dapat berkurang. *Retransmission* ini menyebabkan keseluruhan *throughput* yang berhubungan dengan koneksi menurun jauh. *Packet Loss* (paket hilang) terjadi ketika satu atau lebih paket data yang melewati satu jaringan gagal mencapai tujuannya. *Packet loss* ini merupakan persentase dari rasio perbandingan jumlah paket yang diterima terhadap jumlah paket yang dikirim [19].

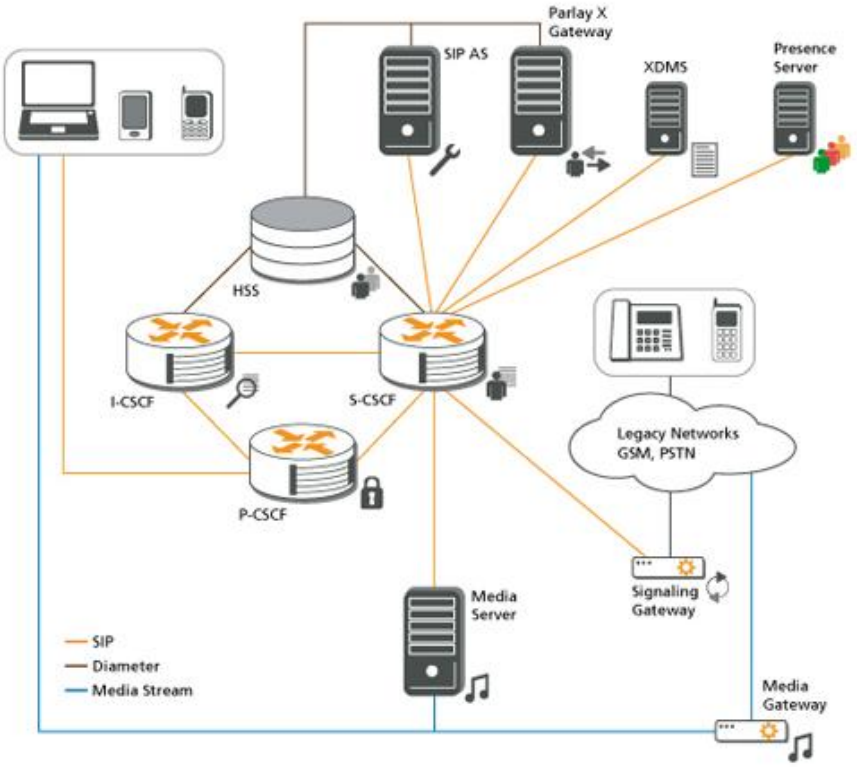
$$Persentase\ Packet\ Loss\ =\ \frac{Jumlah\ paket\ dikirim - Jumlah\ paket\ sampai}{Jumlah\ paket\ dikirim} \times 100\% \text{.....(2.4)}$$

***Delta max* (*latency*)**

Delta max atau disebut *latency* merupakan waktu yang dibutuhkan untuk mengirimkan suatu paket dari satu titik ke titik yang lainnya. Misalnya suatu jaringan membutuhkan waktu 24 ms untuk mengirimkan pesan dari ujung ke ujung, artinya jaringan tersebut memiliki *delta max* 24 ms.

2.8. *OPEN* IMS

*Open*IMS adalah perangkat lunak yang dikembangkan oleh FOKUS (Institut Jerman) Desember 2006. FOKUS mengimplementasikan beberapa komponen secara terintergrasi seperti CSCFs, HSS, *Application Servers*, dan lainnya yang dapat dilihat pada gambar 2.9 [20].



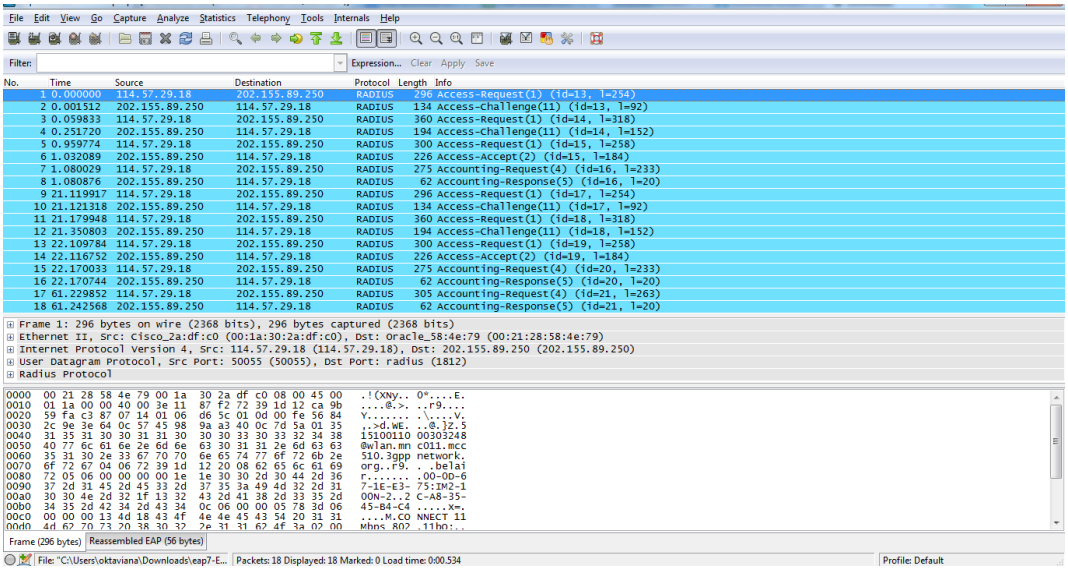
Gambar 2.8. *Testbed Open* IMS [21]

2.9. PROTOCOL ANALYZER WIRESHARK

Wireshark adalah salah satu dari sekian banyak tool *Network Analyzer* yang banyak digunakan oleh *network administrator* untuk menganalisa kinerja jaringannya. *Wireshark* banyak disukai karena antarmukanya yang menggunakan *Graphical User Interface* (GUI) atau tampilan grafis.

Wireshark mampu menangkap paket-paket data/informasi yang ada dalam jaringan. Semua jenis paket informasi dalam berbagai format protokol pun akan dengan mudah ditangkap dan dianalisa. Karenanya tak jarang peralatan ini juga dapat dipakai untuk *sniffing* (memperoleh informasi penting seperti kata sandi email atau akun lain) dengan menangkap paket-paket yang ada di dalam jaringan

dan menganalisanya. Peralatan ini tidak mengenal antarmuka modem dan hanya dapat bekerja dalam jaringan melalui LAN/*Ethernet Card* yang ada di PC. [22]



Gambar 2.9. Contoh hasil *Capture* Wireshark