

ANALISA INTEROPERABILITY SERVER VOIP/SIP DENGAN OPEN IMS

Riski Munawir Utomo Yusuf*)

*) Jurusan Telekomunikasi, Politeknik Elektronika Negeri Surabaya
Institut Teknologi Sepuluh Nopember (ITS) Surabaya
Kampus PENS-ITS, Keputih, Sukolilo, Surabaya
Email: rizkimunawir@gmail.com

ABSTRAK

Saat ini dengan berkembangnya network generasi terbaru (Next-Generetaion) IMS menjadi sangat penting, dan akan menjadi kunci peranan dalam perkembangan infrastruktur berbasis IP. Hanya saja saat ini jaringan tersebut masih dalam tahap pengembangan, dan masih membutuhkan waktu untuk peng-upgrade-an dari 3G mobile menjadi jaringan 3GPP. Oleh karena itu, proyek akhir ini akan mempelajari dan meneliti solusi-solusi interoperability antara openIMS dengan implementasi open source VoIP/SIP. OpenIMS (Open Source IMS) dikembangkan oleh FOKUS (institusi Jerman) pada desember 2006. Sistem ini masih banyak yang meragukan untuk diimplementasikan, oleh karena itu sangatlah penting untuk dievaluasi. Evaluasi diawali dengan melakukan pengujian terhadap jaringan dari OpenIMS, lalu dengan menggunakan perangkat lunak wireshark dilakukan evaluasi terhadap paket-paket data dan proses pensinyalan untuk membuktikan fungsi-fungsi dari komponen-komponen OpenIMS apakah berjalan sesuai dengan spesifikasi 3GPP v10.

Dalam proyek akhir ini, bagian terpenting adalah menemukan solusi terbaik interoperability antara OpenIMS dengan SIP/VoIP yang sudah ada. Beberapa parameter yang dievaluasi antara lain tetang QoS VoIP dari OpenIMS (rekomendasi ITU-T) dan fungsi dari masing-masing komponen dalam openIMS. Dikarenakan hal itu, pokok permasalahan dibagi menjadi dua bagian. Bagian yang pertama adalah membangun sebuah lingkungan kerja dari openIMS dan melakukan pengujian kinerja dari sistem tersebut. Bagian yang kedua adalah melakukan analisa sistem tersebut saat melakukan komunikasi antara sesama jaringan atau klien yang menggunakan protokol IMS dan komunikasi dengan jaringan atau klien yang menggunakan protokol SIP. Dari hal tersebut diharapkan dapat ditemukan solusi terbaik untuk pengimplementasian OpenIMS.

Kata kunci : IMS, OpenIMS, SIP, SIP Signaling

1.PENDAHULUAN

Dengan berkembangnya generasi terbaru dari struktur jaringan atau sering disebut dengan *next-generation network* (NGN), muncullah beberapa standar-standar baru untuk komunikasi data baik data yang berupa gambar, suara, video dan format-format data lainnya. Salah satu yang paling terkenal adalah VoIP (Voice Over IP), khususnya dalam perkembangannya di dunia bisnis. Secara berangsur-angsur hal ini akan menggeser teknologi komunikasi yang masih bersifat tradisional dengan teknologi komunikasi modern yang lebih murah, efektif dan efisien. Selain itu dibandingkan dengan teknolgi sebelumnya, teknologi modern atau sering disebut dengan teknologi VoIP memiliki keunggulan dalam memproses data dan suara yang lebih kompleks dan dapat diteruskan menuju berbagai aplikasi yang berbeda pada *user*. Bahkan dengan protokol VoIP yang telah mengalami sedikit modifikasi dan didukung dengan penerapan solusi yang tepat dapat dibentuk sebuah komabilitas dan interoperabilitas yang cukup handal dalam sebuah komunikasi.

IMS atau sering disebut *IP Multimedia SubSystem* merupakan salah satu perkembangan teknologi NGN. Bahkan telah menjadi faktor utama dalam perkembangan dari NGN. IMS juga memberikan beberapa fitur seperti QoS, keamanan jaringan, manajemen grup, dan pengiriman pesan suara. IMS mempermudah operator sebuah jaringan untuk mengembangkan sebuah layanan-layanan baru yang lebih handal bila dibandingkan dengan layanan yang ada pada GSM yang masih terbilang cukup standar dan sangat terbatas. IMS ini memiliki arsitektur yang standar

tapi mampu menggabungkan layanan multimedia antara dunia seluler dengan dunia IP, tanpa harus merubah protokol standar yang telah digunakan oleh keduanya. Hal ini telah banyak dijelaskan pada 3rd Gneration Partnership Project (3GPP). Keuntungan yang didapat dengan menggunakan IMS ini antara lain untuk sebuah provider, jaringan IMS ini memungkinkan untuk mempercepat dan mempermudah dalam pembuatan dan pengembangan sebuah layanan baru sehingga dapat mengurangi bahkan menekan pengeluaran dari provider tersebut, sehingga dapat meningkatkan nilai ekonomis dari suatu provider. Sedangkan pada sisi *end-user*, IMS menawarkan sebuah komunikasi baru secara *real-time* yang sangat fleksibel dan dapat diatur sedemikian rupa sehingga sesuai dengan keinginan dari *user* tersebut dan hanya dengan menggunakan satu terminal IMS. Selain itu komunikasi dengan IMS dapat menghubungkan berbagai macam jaringan dengan protokol yang berbeda bahkan dengan perangkat komunikasi yang berbeda pula. Dengan berbagai fitur yang dimiliki IMS ini, IMS dapat menjadi inti (*core*) dari perkembangan *Next-Generation Network*.

2.LANDASAN TEORI

Pada proyek akhir ini beberapa landasan teori yang digunakan antara lain IMS, OpenIMS, VoIP/SIP, OpenSer

2.1 IMS

IMS merupakan sebuah jaringan yang didefinisikan oleh 3rd Gneration Partnership Project (3GPP) sebagai sebuah sistem baru yaitu sebuah infrastruktur jaringan bergerak yang memungkinkan pemusatan bebagai macam data dan teknologi ke dalam sebuah infrastruktur yang

berbasis IP. IMS dirancang untuk menjembatani teknologi telekomunikasi yang sudah ada atau tradisional dengan teknologi internet yang sedang berkembang. Jaringan IMS memungkinkan untuk digunakannya dalam layanan *real-time*, layanan berbasis multimedia bergerak seperti video teleponi, tele-konferensi dan lainnya. Dalam jaringan ini mengedepankan tentang mutu layanan (QoS) dan manajemen mobilitas yang handal.

Beberapa alasan mengapa IMS dibutuhkan adalah IMS dapat memberikan layanan-layanan dengan mengacu pada QoS (Quality of Service). Selama ini penggunaan domain *packet switch* hanya dapat menghasilkan layanan terbaik tanpa ada parameter QoS didalamnya. Pernyataan tersebut dapat dicontohkan seperti sebuah komunikasi yang menggunakan VoIP, keadaan dari komunikasi tersebut sewaktu-waktu dapat berubah secara drastis. Suatu saat komunikasi dapat berlangsung dengan suara yang terdengar sangat jelas dan bagus, tetapi di lain hal itu dapat berubah menjadi komunikasi yang buruk yaitu suara yang tidak dapat didengarkan, sehingga informasi tidak dapat di terima dengan baik. Jaringan IMS memungkinkan sebuah operator untuk dapat mengatur QoS setiap *user* yang menggunakan jasa layanannya, sehingga setiap *pengguna* bisa mendapatkan layanan terbaik dalam berkomunikasi. Selain itu adalah operator dapat mengatur secara rinci tentang biaya layanan multimedia yang diberikan pada setiap pengguna, sehingga operator dapat membuat parameter-parameter khusus tentang kapan harus menggunakan standar-standar biaya tertentu sesuai dengan jenis layanan yang digunakan oleh *user*.

Arsitektur dari IMS ini kurang lebih sama dengan SIP hanya terdapat dari perbedaan nama dari masing-masing komponen.

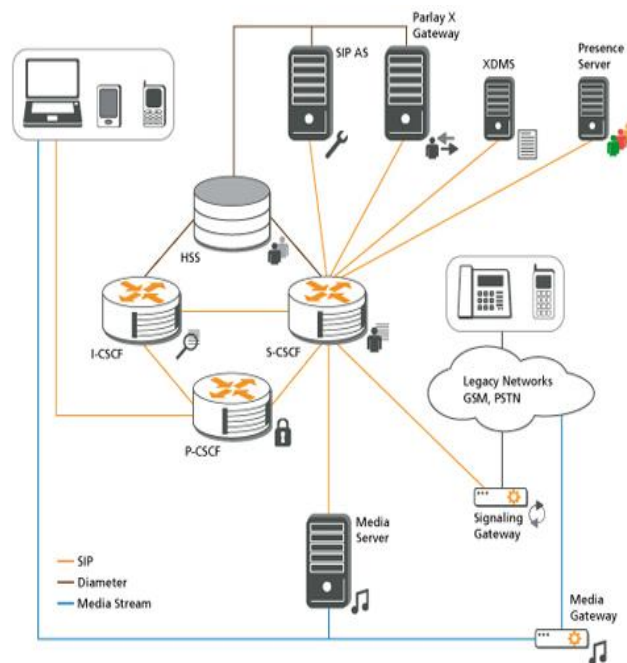
1. Home Subscriber Servers (HSS) Subscriber Location Functions(SLF)
2. Call/Session Control Functions(CSCF)
3. Application Servers(AS)
4. Media Resource Functions(MRF),yang terdiri atas Media Resource Function Controllers(MRFC) dan Media Resource Function Processes(MRFP)
5. Breakout Gateway Control Functions(BGCF) PSTN Gateway, yang terdiri atas Signaling Gateway(SGW), Media Gateway Controller Function(MGCF) dan Media Gateway(MGW)

2.2 OpenIMS

OpenIMS merupakan proyek *open source* milik Fraunhofer Institute FOKUS yang dikembangkan khusus untuk membangun sebuah jaringan IMS. Proyek OpenIMS ini bertujuan untuk melayani jaringan IMS yang sudah ada pada lingkungan *open source* yang mengedepankan fleksibilitas dan kemudahan untuk mengembangkannya. OpenIMS ini digunakan sebagai media untuk melakukan tes terhadap jaringan IMS sehingga dapat ditemukan solusi atau skenario terbaik untuk pengembangan jaringan IMS.

Komponen yang terdapat dalam Open IMS ini merupakan implementasi dari komponen yang ada pada jaringan IMS seperti HSS, CSCF, PSCF, Media Gateway dan komponen IMS lainnya yang di gabungkan menjadi sebuah lingkungan yang baru. OpenIMS juga

menyediakan beberapa layanan yang lain seperti Open Service Acces(OSA)/Parlay dan Web Service/Parlay X. Gambaran tentang lingkungan dari Open IMS dapat dilihat pada gambar 1. Perbedaan antara IMS dengan Open IMS adalah SIP2IMS hanya terdapat pada OpenIMS saja sedangkan pada jaringan IMS tidak terdapat komponen tersebut. SIP2IMS ini digunakan oleh IMS untuk berkomunikasi dengan sistem yang menggunakan SIP sebagai protokolnya.



Gambar 1 Arsitektur OpenIMS

2.3 VoIP/SIP

VoIP atau Voice Over Internet Protocol merupakan sebuah teknologi komunikasi yang menggunakan Internet Protokol melalui sistem komunikasi tradisional atau sering disebut sistem analog. VoIP bekerja dengan merubah sinyal suara menjadi sebuah sinyal digital yang dapat di transmisikan melalui internet. Pada penerima sinyal tersebut akan dikembalikan menjadi sinyal analog sesuai dengan yang dikirimkan. Terdapat banyak layanan yang dapat ditangani oleh VoIP antara lain melakukan panggilan melalui komputer atau melakukan panggilan telepon melalui jaringan IP baik secara langsung maupun dengan menggunakan tambahan perangkat untuk sinkronisasi data.

SIP merupakan protokol pensiyalan untuk mengatur, memulai dan mengakhiri sesi video dan suara yang ada dalam jaringan paket data. SIP ini dalam melakukan komunikasi yang melibatkan satu atau beberapa *user* yang berbeda dalam satu komunikasi dan dapat juga menggunakan metode komunikasi unicast maupun multicast. Dengan menduplikasi berbagai macam protokol-protokol lain yang digunakan dalam internet seperti HTTP dan SMTP, SIP adalah protokol berbasis text dan masih sangat memungkinkan untuk diperluas kegunaannya. Pengembangan tersebut dapat berupa kemampuan SIP untuk mengatur fitur-fitur dan layanan seperti *call-control*, mobilitas, interoperability, dengan tetap memanfaatkan sistem telepon yang kini sudah ada dan sudah dipergunakan secara luas.

Jaringan SIP dibangun dari empat bagian SIP, keempat bagian tersebut memiliki fungsi-fungsi khusus dan dalam komunikasi SIP, keempat bagian tersebut dapat berposisi sebagai *client* (melakukan permintaan), *sebagai server* (merespon permintaan) ataupun berposisi pada keduanya yaitu *client* dan *server*. Keempat komponen tersebut yaitu,

1. *User Agent (UA)* – End User
2. *Proxy Server* – Gerbang menuju jaringan SIP
3. *Redirect Server* – Memetakan lokasi user dan meneruskan permintaan
4. *Registrar Server* – Memproses registrasi user

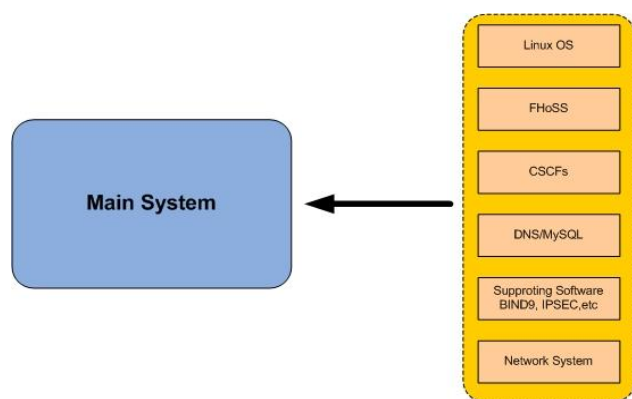
2.3 OpenSER

OpenSER merupakan sebuah perangkat lunak (sistem) yang menggunakan protokol SIP untuk berkomunikasi. OpenSER ini dapat bertindak sebagai proxy server, router, *user agent* registration server. Sistem ini merupakan sistem yang tidak berbayar dan terlisensi dibawah GNU general Public License. Dibandingkan dengan asterisk yang juga merupakan kombinasi IP PBX dengan SIP server, OpenSER yang lebih fokus pada proses pensinyalan SIP saja dapat bekerja lebih cepat dibandingkan asterisk yang menggunakan sistem B2BUA (back to back *user agent*). OpenSER mampu melayani jumlah panggilan yang cukup besar, dan menerapkan sistem *load balancing* yang cukup efektif. Untuk melakukan komunikasi dengan PSTN dibutuhkan sebuah media yang sering disebut dengan SIP gateway.

Terdapat beberapa perubahan dalam inti dari openSER ini, dengan ditimbahnya beberapa fitur baru openSER muncul dengan dua buah nama baru yaitu openSIPS dan Kamailio. Meskipun berbeda nama tetapi dari segi fungsi utama yaitu sebagai sebuah sever SIP antara ketiga perangkat lunak tersebut sama. Saat ini hanya openSIPS dan kamailio yang masih terus dikembangkan.

3 METODOLOGI

3.1 Perencanaan Sistem



Gambar 2 Blok Diagram Sistem

Sistem yang dibuat dalam proyek akhir ini akan bekerja pada sebuah *testbed* (media untuk melakukan pengujian). Oleh karena itu sebelum sistem dibentuk terdapat sebuah

media *testbed* yang harus dipersiapkan. Seperti yang telah dijelaskan pada bab sebelumnya tentang OpenIMS, maka perlu dilakukan instalasi komponen-komponen openIMS dan pendukungnya. Media *testbed* tersebut seperti digambarkan pada gambar 3.1

Cara kerja dari sistem ini diawali dengan membangun sebuah *testbed* atau media pengujian yang nantinya akan di isi dengan sistem yang akan diuji, dalam hal ini adalah OpenIMS. Media pengujian yang akan dibangun adalah sebuah perangkat komputer yang telah terinstall sebuah sistem operasi open source dengan distro debian kernel 2.6.26. pada perangkat pengujian tersebut juga ditambahkan beberapa perangkat lunak tambahan untuk mendukung kinerja dari sistem utam antara lain server DNS menggunakan bind9, server database menggunakan MySQL, Java Environment menggunakan Sun Java JDK 1.6 dan beberapa perangkat lunak lainnya yang dibutuhkan selama proses kompilasi dari OpenIMS. Setelah media pengujian terbentuk proses selanjutnya adalah membangun sistem utama yaitu OpenIMScore. Proses diawali dengan melakukan kompilasi terhadap kode openIMS sehingga menjadi sebuah sistem yang dapat dijalankan atau dengan cara membuat paket openIMS yang nantinya akan diinstall pada media pengujian. Setelah seluruh sistem telah terbentuk dengan baik, proses selanjutnya adalah melakukan pengujian terhadap performa dari openIMS.

Pengujian dilakukan dalam beberapa tahap yaitu pengujian terhadap fungsi dari openIMS, pengujian terhadap komponen-komponen yang membangun openIMS, pengujian interoperability openIMS dengan SIP, pengujian terhadap integrasi openIMS dengan SIP dan pengujian terhadap performa dari openIMS.

3.2 Pembuatan Sistem

Pembuatan sistem diawali dengan membangun sebuah media test bed tempat dimana sistem utama atau openIMS nanti akan ditempatkan seperti yang telah dibahas sebelumnya. Setelah itu diteruskan dengan membangun openIMS

1. Instalasi FhoSS

Proses instalasi FhoSS ini diawali dengan mengambil file kode yang berada pada <http://svn.berlios.de/svnroot/repos/openimscore>. Kode ini merupakan kode yang telah dikonfigurasi untuk membangun FhoSS dengan fitur yang standar. Instalasi dapat dilakukan dengan dua cara yaitu dengan membuat paket debian yang berasal dari kode tersebut atau hanya melakukan kompilasi dan langsung menggunakannya. Pastikan seluruh perangkat pendukung lainnya telah terinstall terlebih dahulu.

Setelah proses instalasi berhasil, beberapa file konfigurasi yang perlu diperhatikan adalah

1. *DiameterPeerHSS.xml* -- Konfigurasi yang dibutuhkan dalam file ini adalah FQDN, *Acceptor Port* dan beberapa identitas untuk otorisasi.

2. `Hibernate.properties` – Konfigurasi hibernate, salah satunya adalah mengatur konektivitas dengan mysql pada localhost, seperti:
`Hibernate.connection.url=jdbc:mysql:http://127.0.0.1:3306/hssdb`
`Hibernate.connection.username=hss`
`hibernate.connection.password=hss`
3. `Hss.properties` – Berisi konfigurasi yang lebih spesifik seperti alamat yang dirujuk oleh tomcat (localhost) dan tempat direktori dari FhoSS web (/hss.web.console)
4. `Log4j.properties` – Konfigurasi tentang file-file log yang digunakan oleh FhoSS. Berisi tentang tempat menyimpan log yang terjadi pada FhoSS dan level log nya.

Konfigurasi FhoSS dapat diakses melalui media web dengan alamat `http://ip_address:8080`.

2. Instalasi CSCF

Sama seperti FhoSS, kode dari CSCF dapat diambil dari `http://svn.berlios.de/svnroot/repos/openimscore`. Kode ini telah di konfigurasi untuk membangun sebuah sistem CSCFs dengan fitur standar. Sebelumnya telah dibuat direktori `/opt/OpenIMSCore` sehingga hanya perlu dibuatkan sebuah direktori dibawah `OpenIMSCore` dengan nama `ser_ims`. Prose instalasi juga dapat dilakukan dengan membuat paket debian dari kode tersebut atau melakukan kompilasi kode dan langsung menggunakannya.

Setelah seluruh instalasi selesai maka jalankan seluruh komponen tersebut dan pastikan tidak terdapat pesan kesalahan yang muncul pada file log masing-masing komponen.

3. Konfigurasi Klien

Konfigurasi pada sisi klien harus disesuaikan dengan data yang tersimpan pada database FhoSS. Salah satu hal yang penting dan diperhatikan adalah klien harus diarahkan untuk menghubungkan proxy server dari openIMS atau PCSCF pada port 4060 (outbound proxy). Setelah seluruh konfigurasi selesai maka klien telah siap untuk melakukan proses selanjutnya yaitu konfigurasi dan komunikasi dengan serv openIMS atau dengan klien lainnya.

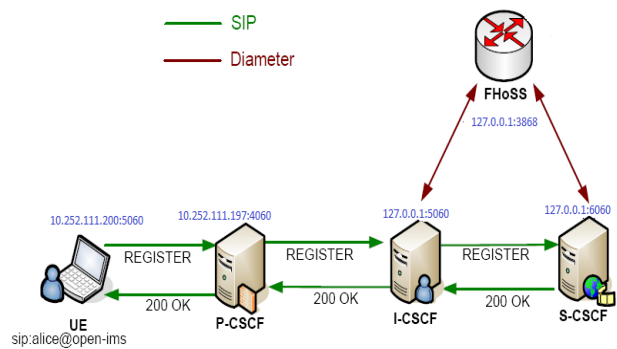
4 PENGUJIAN DAN ANALISA

4.1 Pengujian Fungsi OpenIMS

Pada tahap pengujian ini adalah untuk menguji kerja dari sistem open IMS dalam melayani proses registrasi dan proses komunikasi.

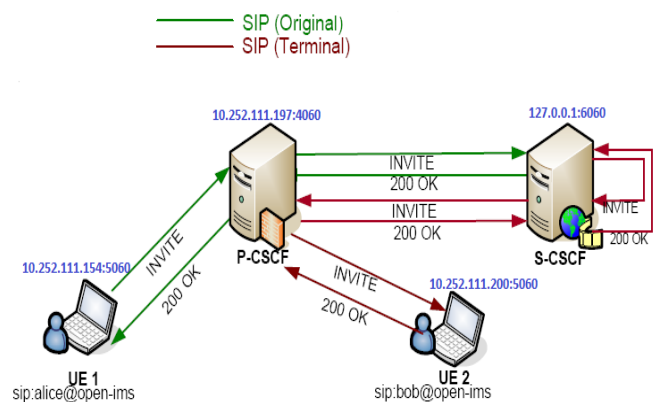
1. Proses Registrasi

Proses registrasi diawali dengan koneksi yang dilakukan oleh user menuju alamat dari PCSCF yang bertindak sebagai *outbound/inbound SIP proxy server*. Proses ini merupakan tahap yang paling penting karena proses pensinyalan dibangun disini. Proses registrasi terminal IMS ini sama dengan proses registrasi pada user server SIP. Secara garis besar proses registrasi digambarkan pada gambar 3



Gambar 3 Proses Registrasi pada openIMS

2. Proses Komunikasi



Gambar 4 Proses Komunikasi pada openIMS

Proses komunikasi diawali dengan permintaan panggilan dari user yang akan diterima PCSCF untuk ditambahkan informasi tentang alamat PCSCF itu agar pesan tersebut nantinya setelah diproses dapat dikembalikan pada PCSCF tersebut. Informasi baru tersebut diteruskan menuju SCSCF yang telah dipilih oleh FhoSS saat proses registrasi untuk melayani permintaan dari klien yang melakukan permintaan panggilan. SCSCF akan melakukan pengecekan terhadap identitas Public user nya dan akan merutekannya melalui PCSCF.

4.2 Pengujian terhadap masing-masing komponen open IMS

Pada tahap ini pengujian akan dilakukan dengan membandingkan dengan standarisasi yang dikeluarkan oleh 3GPP yaitu TS.24.229 V10. Pengujian akan dilakukan untuk mengetahui kesesuaian kinerja openIMS dengan standar yang ada.

4.3 Pengujian Interoperability OpenIMS dengan SIP

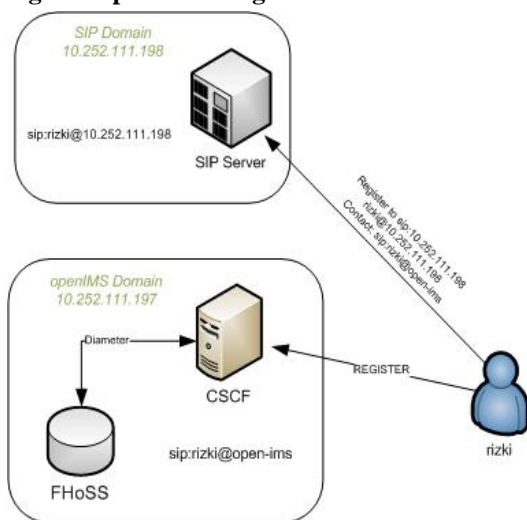
Setelah melakukan beberapa pengujian, ditemukan solusi yang paling memungkinkan untuk diterapkan yaitu hanya digunakan salah satu protokol SIP atau IMS saja, maksudnya adalah pada sisi klien hanya akan dapat berkomunikasi SIP-SIP atau IMS-IMS. Tetapi pada kenyataannya kondisi tersebut sangat kurang apabila diterapkan sebagai solusi enterprise karena setiap klien hanya akan mampu untuk menggunakan salah satu dari

protokol yang ada, SIP atau IMS saja. Oleh karena itu diperlukan sebuah solusi yang mampu untuk menggabungkan keduanya.

Solusi yang dijalankan adalah dengan membangun dua buah SCSCF untuk melayabi perbedaan otentikasi antara klien SIP dan klien IMS. Setelah kedua SCSCF terbentuk dan telah terhubung dengan sistem utama maka antara klien IMS dan klien SIP dapat melakukan registrasi secara bersamaan. Selain itu keduanya telah dapat berkomunikasi secara langsung.

Terdapat permasalahan saat penerapan solusi tersebut yaitu status dari user yang menggunakan SCSCF2 akan beberpa klai mengalami kesalahan, salah satunya adalah berubahnya konfigurasi pada database, hal ini disebabkan oleh penambahan SCSCF dan database secara manual terhadap FhoSS yang telah didesain untuk melayani satu skema otentikasi saja.

4.4 Integrasi OpenIMS dengan SIP



Gambar 5 Registrasi pada solusi “client based”

Pengujian yang dilakukan sebelumnya seluruhnya terjadi dalam satu domain yang sama. Tetapi sebenarnya terdapat domain-domain lain yang sebenarnya harus dihubungkan. Sehingga migrasi antara domain yang berbeda ini sangat dibutuhkan.

Pada tahap migrasi ini diasumsikan seorang user memiliki dua identitas yang mana identitas pertama digunakan untuk registrasi dengan domain IMS sedangkan user yang kedua digunakan untuk registrasi pada domain yang berbeda. Salah satu contoh kasus nya adalah saat terdapat user yang terdaftar dalam domain IMS akan menghubungi user dengan domain selain IMS maka panggilan tersebut harus dapat dilakukan. Oleh karena itu dibutuhkan adanya interoperability antara kedua domain tersebut.

Salah satu solusi yang dapat diterapkan adalah “client based” . Solusi ini membutuhkan sebuah klien yang mampu untuk melakukan registrasi dan menambahkan pesan kepada server yang berisi kontak atau URI lainnya yang dimiliki oleh user tersebut. Saat user tersebut mengaktifkan terminal IMS maka user akan melakukan registrasi sebagai bagian dari domain IMS, tetapi saat user menggunakan klien SIP maka user akan melakukan registrasi dengan server SIP sekaligus mengirimkan

informasi tentang URI lainnya yang dimiliki user tersebut (IMS URI). Saat terdapat panggilan ke user tersebut dalam domain SIP maka server SIP akan bertindak sebagai redirect server dan mengirimkan pesan tentang perubahan lokasi user tersebut yang berada pada domain IMS.

Saat digunakan klien IMS untuk melakukan panggilan dan klien tersebut telah mendapatkan lokasi baru dari “rizki”. Tidak terjadi proses *invite* ulang yang dilakukan oleh klien IMS. Saat menggunakan klien sip yang teregistrasi pada server SIP yang sama, proses *invite* ulang hanya mampu mencapai PCSCF OpenIMS, lalu PCSCF akan mengirimkan pesan *redirect* menuju klien yang dituju. Saat dilakukan pengecekan dengan menggunakan Wireshark terdapat dua sesi dalam satu komunikasi (Cseq 1 dan Cseq2). Hal ini yang menyebabkan tidak dapat terjadinya proses redirect menuju user “rizki” pada domain openIMS, karena dalam satu komunikasi memiliki Call-ID yang sama. Call-ID tersebut merupakan identitas yang merepresentasikan alamat rizki@10.252.111.198 sehingga saat proses telah sampai pada PCSCF, PCSCF tetap akan mengirimkan pesan “300 redirect” menuju alamat Call-ID milik rizki@01.252.11.198.

4.4 Pengujian Performa openIMS

Setelah seluruh sistem telah terbangun dan menerapkan beberapa solusi interoperability IMS dengan SIP. Selanjutnya adalah melakukan pengujian apakah openIMS dapat bekerja dengan baik. Beberapa parameter yang diujikan antar lain,

- Berapa banyak *user* yang dapat dilayani oleh openIMS?
- Apakah sistem akan selalu dapat melayani permintaan dari *user* ?
- Apabila sistem dapat bekerja dengan baik dalam kondisi normal, apakah sistem mampu bekerja dengan baik saat kondisi yang tidak normal seperti terjadi overload?

Pada bagian pengujian ini digunakan SIPp sebagai media pengujian performansi openIMS.

1. Pengujian Registrasi 100 User

Pengujian dilakukan dengan melakukan registrasi 100 user menuju openIMS. Dari pengujian tersebut dapat diketahui bahwa proses pensinyalan yang terjadi saat registrasi 100 user dengan registrasi 1 user adalah sama. Sehingga openIMS mampu untuk menangani registrasi tersebut.

2. Pengujian Performansi OpenIMS

Pengujian selanjutnya adalah dengan melakukan komunikasi antar klien melalui openIMS. Dengan memanfaatkan SIPp akan dilakukan komunikasi secara simultan untuk mengetahui kemampuan openIMS untuk melayani panggilan tersebut. Pengujian yang dilakukan yaitu komunikasi antara klien SIP dengan klien IMS dan komunikasi antara klien SIP dengan klien SIP. Pengujian yang dilakukan adalah melakukan 20 panggilan dimana setiap panggilan terdiri dari 1 proses *invite*. Berikut tabel hasil pengujian tersebut.

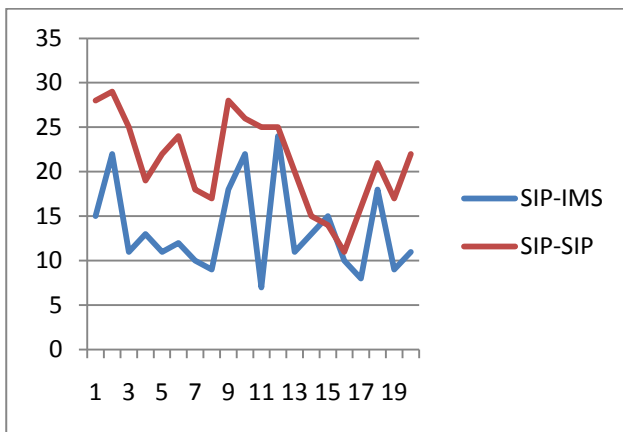
Tabel 1 Komunikasi SIPp – Monster (SIP to IMS)

20 panggilan dengan 1 invite/panggilan	Jumlah	Keterangan
Panggilan Sukses	19	-
Panggilan Gagal	1	600 Busy

Tabel 2 Komunikasi SIPp – X-Lite (SIP to SIP)

20 panggilan dengan 1 invite/panggilan	Jumlah	Keterangan
Panggilan Sukses	20	-
Panggilan Gagal	0	-

Pengujian selanjutnya adalah melakukan 20 panggilan diman setiap panggilan terdiri dari 30 proses invite. Pemilihan nilai 30 adalh karena kemampuan limit panggilan sukses yang mampu dilayani oleh SIPp. Nilai tersebut masih terlalu kecil untuk mengetahui apakah openIMS dapat bekerja lebih baik dibandingkan dengan SIP. Berikut grafik hasil pengujian tersebut



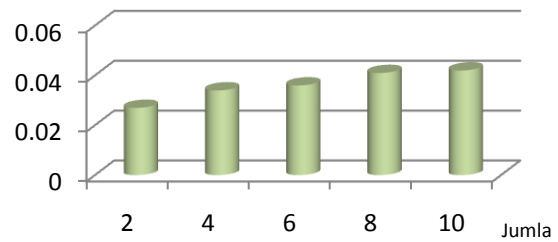
Gambar 6 Grafik Perbandingan Komunikasi SIP-SIP dengan SIP-IMS

Dari grafik tersebut dapat terlihat bahwa performa dari SIP-SIP lebih baik dari performa SIP-IMS.karena untuk dapat melakuakn komunikasi ini perlu ditambahkan SCSCF2 yang ditambahkan secara manual, sehingga komunikasi yang terjadi diantar keduanya menajdi tidak sempurna.Hal ini juga yang mengakibatkan sering terjadi perubahan tentang informasi klien secara random pada database. Tetapi untuk klien IMS hal tersebut lebih baik karena terdaftar dalam SCSCF yang sebenarnya.

Pengujian selanjutnya adalah tentang QoS dari jaringan openIMS, beberapa parameter yang akan diuji adalah *delay*, *jitter* dan *throughput*. Pengujian ini dilakukan dengan beberapa kondisi atau skenario. Skenario yang dibuat adalah pada jumlah panggilan yang dibangun dalam setiap pengujiannya. Jumlah panggilan yang dibuat adalah 2 panggilan, 4 panggilan, 6 panggilan 8 panggilan dan 10 panggilan. Pemilihan jumlah panggilan dalam pengujian ini dipengaruhi oleh keterbatasan perangkat yang dapat digunakan untuk melakukan pengujian.

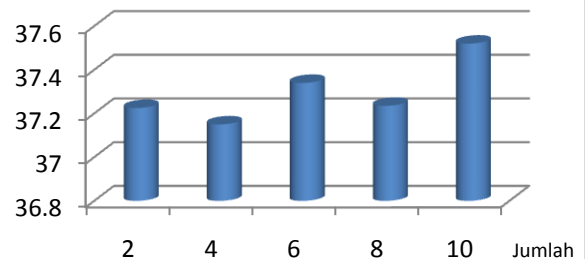
Pengujian selanjutnya adalah untuk mengetahui kemampuan openIMS melayani proses *invite* dalam satu waktu. Yaitu dengan melakukan proses *invite* dengan jumlah *invite* yang bervariasi.

Rata-Rata Delay (ms)



Gambar 7 Diagram Rata-Rata *delay* dalam beberapa skenario komunikasi yang berbeda

Rata-Rata Jitter (ms)



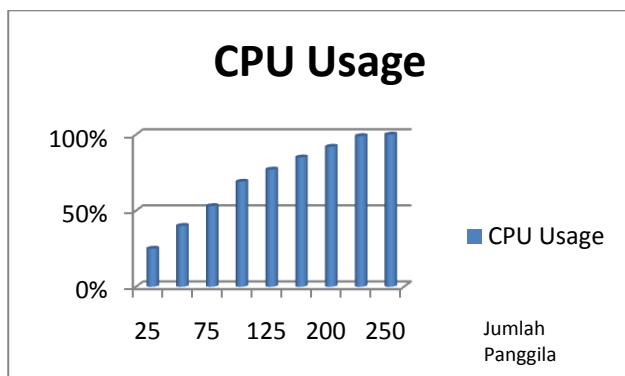
Gambar 8 Diagram Rata-Rata *jitter* dalam beberapa skenario komunikasi yang berbeda

Dari data-data tentang *delay*, *jitter* dan *throughput* dari jaringan openIMS yang diuji ini dapat diketahui beberapa hal mengenai kualitas layanan sebuah jaringan. Nilai rata-rata *delay* yang didapatkan dalam pengujian ini adalah berada pada nilai 0.036ms. Menurut rekomendasi dari ITU-T G.144 tentang *delay* bahwa batas *delay* yang dapat diterima untuk sebuah komunikasi VoIP adalah antara 0 ms -150 ms. *Delay* yang terjadi ini dalam komunikasi ini adalah *packetization delay* yaitu *delay* yang disebabkan oleh pengakumulasian bit *voicetample* menjadi *frame*. Karena media pengujian menggunakan *wireline* jarak pendek maka *delay* media fisik dapat diabaikan karena memiliki nilai yang sangat kecil. *Jitter* merupakan masalah yang terjadi dalam *slow speed links*. Standar *jitter* yang dapat diterima dalam sebuah komunikasi VoIP adalah kurang dari 150ms. *Jitter* ini dapat ditekan dengan memanfaatkan mekanisme *priority buffer*, *bandwith reservation* (RSVP, MPLS) dan koneksi kecepatan tinggi.

Dari tabel tabel 3 dapat diketahui menurunnya performa openIMS dalam melayani jumlah panggilan yang lebih besar dalam tiap detiknya. Hal ini salah satunya dipengaruhi oleh spesifikasi dari perangkat yang digunakan oleh openIMS. Dengan semakin meningkatnya jumlah *invite* tiap detiknya ini juga berpengaruh pada waktu hingga permintaan *invite* tersebut dapat dilayani. Karena dalam sebuah komunikasi openIMS yang menggunakan SIP sebagai media pensinyalannya, akan dibangun jalur komunikasi terlebih dahulu dengan memanfaatkan pensiyalan SIP sebelum komunikasi anatara *user* dapat terjadi.

Tabel 3 Prosentase Resources yang Digunakan untuk Melayani panggilan dalam jumlah tertentu

Panggilan /detik	CPU Usage	Keterangan
25	25%	Seluruh <i>Invite</i> dapat dilayani
50	40%	Seluruh <i>invite</i> dapat dilayani
75	53%	Seluruh <i>Invite</i> dapat dilayani
100	69%	Seluruh <i>Invite</i> dapat dilayani
125	77%	Retransmit <i>Invite</i>
150	85%	Retransmit <i>Invite</i>
200	92%	Retransmit <i>Invite</i>
225	99%	OverLoad
250	100%	OverLoad



Gambar 9 Grafik Resources yang Digunakan untuk Melayni panggilan dalam jumlah tertentu

Dari tabel dan grafik diatas dapat diketahui menurunnya performa openIMS dalam melayani jumlah panggilan yang lebih besar dalam tiap detiknya. Hal ini salah satunya dipengaruhi oleh spesifikasi dari perangkat yang digunakan oleh openIMS. Dengan semakin meningkatnya jumlah *invite* tiap detiknya ini juga berpengaruh pada waktu hingga perminttan *invite* tersebut dapat dilayani. Karena dalam sebuah komunikasi openIMS yang menggunakan SIP sebagai media pensinyalannya, akan dibangun jalur komunikasi terlebih dahulu dengan memanfaatkan pensiyalan SIP sebelum komunikasi antara *user* dapat terjadi.

5. KESIMPULAN

Dari hasil penelitian dan analisa dari sistem yang dibuat maka dapat diambil kesimpulan yaitu:

1. Interoperability openIMS dengan SIP yang dilakukan adalah dengan menambahkan satu SCSCF lagi. Penambahan tersebut digunakan untuk melayani otentifikasi *client* berbasis SIP karena skema otentifikasi antara *client* SIP dan *client* IMS berbeda. Dengan penambahan tersebut komunikasi antar keduanya dapat terjadi.
2. Integarsi SIP dan OpenIMS dilakukan dengan menggunakan solusi "*client base*" yaitu solusi yang memanfaatkan lokasi *client* terhadap *server*. Dalam hal ini digunakan dua maca server yaitu IP PBX dan SIP Server. Perbedaan

antara keduanya adalah pada IP PBX dibutuhkan pembuatan trunk baru yang khusus digunakan untuk komunikasi dengan openIMS sedangkan pada SIP server hanya perlu dilakukan proses routing saja.

3. Pengujian tentang QoS jaringan menunjukkan bahwa nilai *delay* (0.036ms) masih berada pada batas baik menurut rekomendasi ITU-T tentang VoIP. Sama halnya dengan *Jitter* yang bernilai 37,264ms dimana masih berada pada nilai dibawah batas nilai yang dapat diterima dalam komunikasi VoIP. Untuk nilai *throughput* berbanding terbalik dengan *delay*, semakin kecil *delay* maka semakin besar *throughput*nya.
4. Data yang diperoleh dari pengujian beberapa skenario panggilan menunjukkan bahwa kemampuan server untuk dapat melayani seluruh permintaan *invite* adalah pada 100 panggilan/detik dimana server berada telah menggunakan 80% *resource* nya, saat jumlah panggilan dinaikkan maka akan terjadi proses retransmit berulang.

6 DAFTAR PUSTAKA

- [1]. Lian Wu, Anders Aasgaard, "Migration of VOIP/SIP Enterprise Solutions towards IMS", Master Thesis Report, Agder University College, Juni 2006.
- [2]. Fei Yao, Li Zhang, "OpenIMS Express VoIP Solutions", Master Thesis Report, Agder University College, Juni 2007.
- [3]. Rendy Munadi, Efan Najwaini, Asep Mulyana, R.Rumani.M, "Design and Implementation VoIP Service on OpenIMS and Asterisk Servers Interconnected Trough Enum Server, International Journal of Next-Generation Networks (IJNGN), Juni 2010.
- [4]. Fokus Open Source IMS Core, URL : <http://opeimscore.org>, diakses tanggal 4 Februari 2011.
- [5]. SER = Sip Express Router, URL : <http://opensips.org>, diakses tanggal 4 Februari 2011.
- [6]. SIPp, URL : <http://sipp.sourceforge.net/>, diakses tanggal 4 Februari 2011.
- [7]. 3GPPTS 24.229 v10, Internet Protocol Multimedia Call Control based on SIP, Juni 2010.
- [8]. ITU-T G.711, Pulse Code Modulation of Voice Frequencies, Februari 2000.