

# IMPLEMENTASI LAYANAN *INSTANT MESSAGING* BERBASIS *IP MULTIMEDIA SUBSYSTEM* MENGGUNAKAN *VIRTUAL SERVER*

Fakkar Robbi Radhian<sup>\*)</sup>, Yuli Christyono, and Sukiswo

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro,  
Jl. Prof. Sudharto, SH, Kampus UNDIP Tembalang, Semarang 50275, Indonesia

<sup>\*)</sup>E-mail: [radhian89@gmail.com](mailto:radhian89@gmail.com)

## Abstrak

IMS (IP Multimedia Subsystem) didefinisikan oleh 3GPP (3<sup>rd</sup> Generation Partnership Project) sebagai subsistem baru yaitu suatu teknologi jaringan yang memungkinkan pemusatan data, suara, dan gambar melalui suatu infrastruktur berbasis IP. IM (Instant Messaging) adalah teknologi yang memungkinkan para pengguna dalam jaringan untuk mengirimkan pesan singkat secara langsung pada saat yang bersamaan menggunakan teks, gambar, atau pengiriman berkas kepada pengguna lainnya yang sedang terhubung ke jaringan yang sama. Pada tugas akhir ini akan diimplementasikan dan dianalisis layanan Instant Messaging berbasis IMS menggunakan virtual server. Dari implementasi ini akan dianalisis dari tinjauan Quality of service yaitu delay, packet loss, jitter dan throughput diharapkan mampu memberikan informasi tentang parameter kelayakan QoS pada jaringan IMS untuk layanan Instant Messaging. Dari hasil pengujian dan analisis dapat diketahui bahwa IMS dapat diimplementasikan pada jaringan lokal menggunakan server virtual. Nilai delay, jitter, throughput, dan packet loss sudah sesuai dengan standar dari ITU-T G.1010. Selain itu diketahui pula bahwa kualitas layanan IMS tergantung penggunaan jaringan yang digunakan.

*Kata Kunci* :IMS, Instant Messaging, dan QoS.

## Abstract

IMS (IP Multimedia Subsystem) defined by 3GPP (3<sup>rd</sup> Generation Partnership Project) As the new subsystem is network technology that enables the centralization of data, sound and images via an IPbased infrastructure. IM (Instant Messaging) is a technology that allows the users of the network to send brief messages directly at the same time using text, images or sending files to other users who are connected to the same network. The final project will be implemented and analyzed the IMS-based Instant Messaging service using virtual server. The implementation will be analyzed from the review Quality of service among others his delay, packet loss, jitter and throughput is expected to provide information about the QoS parameters on the network requirements of IMS for the Instant Messaging service. Testing and analysis of the results can be known that IMS can be implemented on a local network using a virtual server. The value of delay, jitter, packet loss, and throughput is in compliance with the standard of ITU-T G.1010. Additionally note that IMS service quality depends on the use of network load used.

*Keyword*:IMS, Instant Messaging, and QoS.

## 1. Pendahuluan

IMS (IP *Multimedia Subsystem*) didefinisikan oleh 3GPP (3<sup>rd</sup> *Generation Partnership Project*) sebagai subsistem baru yaitu suatu teknologi jaringan yang memungkinkan pemusatan data, suara, dan gambar melalui suatu infrastruktur berbasis IP. IM (*Instant Messaging*) adalah teknologi yang memungkinkan para pengguna dalam jaringan untuk mengirimkan pesan singkat secara langsung pada saat yang bersamaan menggunakan teks, gambar, dan pengiriman berkas kepada pengguna lainnya yang sedang terhubung ke jaringan yang sama. Kelebihan IMS dibandingkan

teknologi layanan multimedia yang sudah ada adalah adanya pemisahan elemen server sehingga beban *server* menjadi lebih ringan dan keamanan *server* yang lebih baik dikarenakan adanya fungsi *proxy* untuk menyembunyikan identitas pengguna dan menyembunyikan topologi jaringan server.

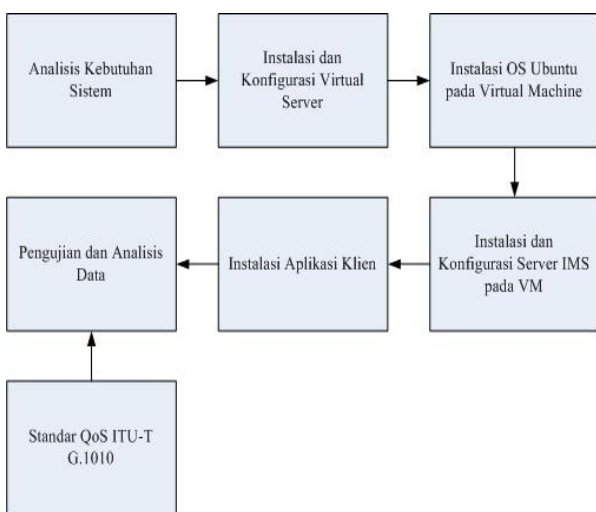
Pada penelitian sebelumnya telah dilakukan implementasi Asterisk sebagai *server* SIP dalam melayani komunikasi VoIP di lingkungan Universitas Diponegoro dan referensi jurnal dari IEEE yaitu yang berjudul *VoIP Quality Optimization in IP Multimedia Subsystem (IMS)* yang telah memodelkan IMS dengan menggunakan aplikasi *simulator* Opnet.

Berdasarkan penelitian tersebut, penulis mencoba membangun layanan *instat messaging* berbasis *IPMultimedia Subsystem* menggunakan *virtual server*. Sistem ini memungkinkan pengguna yang berada dalam lingkungan kampus Teknik Elektro Fakultas Teknik Universitas Diponegoro untuk mendapatkan layanan pengiriman pesan instan, panggilan video, panggilan suara, dan pengiriman berkas melalui sistem berbasis IMS. Selama perangkat pengguna masih berada di dalam wilayah sinyal *WiFi*, pengguna dapat menghubungi pengguna lain melalui sambungan tunggal.

Tujuan dari pembuatan tugas akhir ini adalah membangun suatu sistem untuk layanan *instant messaging* berbasis IMS menggunakan *virtual server* di lingkungan Jurusan Teknik Elektro Fakultas Teknik Universitas Diponegoro kemudian mempelajari dan menganalisa kualitas layanan yang telah diimplementasikan.

## 2. Metode

Dalam tugas akhir ini dibuat sebuah sistem implementasi layanan instant messaging berbasis IP Multimedia Subsystem berbasis virtual server. Layanan yang berbasis pada jaringan *IP Multimedia Subsystem* (IMS) dalam hal ini menggunakan Open IMS core. Pada jaringan ini terdapat 2 komponen utama, yaitu *IMS core* dan *IMS klien*. Dimana fungsi dari *IMS core* adalah memproses semua permintaan dari klien dan meneruskannya ke klien yang lain yang di implementasikan berbasis virtual server yang di sesuaikan dengan konfigurasi jaringan Lokal dari Fakultas Teknik Jurusan Teknik Elektro Universitas diponegoro. Untuk memudahkan proses perancangan dan implementasi diperlukan diagram proses seperti yang terlihat pada Gambar 1



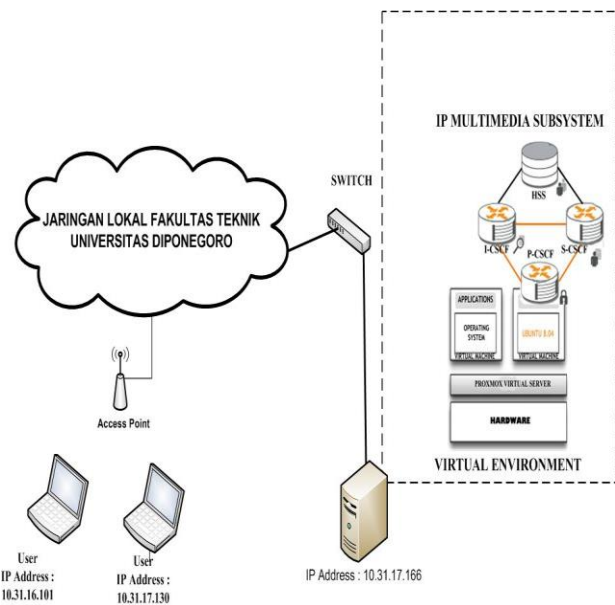
Gambar 1. Diagram alir Tugas Akhir

## 2.1 Perancangan IMS Core

Perancangan layanan Instant Messaging berbasis IP Multimedia Subsystem menggunakan virtual server ini secara garis besar alurnya adalah sebagai berikut

1. Klien akan melakukan instalasi aplikasi klien.
2. Klien akan melakukan registrasi kepada *IMS core*.
3. *IMS core* akan memeriksa apakah klien tersebut sudah terdapat pada database *Home Subscriber Server* (HSS) atau tidak, jika sudah maka proses registrasi berhasil dan klien diperbolehkan melakukan permintaan selanjutnya.
4. Selanjutnya klien akan melakukan SIP permintaan untuk salah satu layanan pesan instan, panggilan video, dan panggilan suara yang tersedia.
5. *IMS core* akan menerima permintaan dari klien dan meneruskannya ke klien lain yang ditujukan untuk menerima pesan atau panggilan.

Perancangan sistem ini diperlukan konfigurasi dari beberapa komponen penting, yaitu *IMS core* dan aplikasi klien, dan registrasi pengguna-pengguna baru. Untuk menghemat penggunaan perangkat keras dan memudahkan dalam implementasi dan pengendaliannya maka untuk komponen *IMS Core* digunakan perangkat lunak Proxmox sebagai *virtual server* dan berada dalam satu personal computer. Sedangkan klien berada dalam personal computer yang berbeda. Arsitektur dari sistem terdapat pada Gambar 2



Gambar 2. Arsitektur sistem

## 2.1 Implementasi Sistem

*Proxmox VE* (*Virtual Environment*) yang bersifat *open source* berfungsi untuk menjalankan mesin virtual berbasis KVM berupa sistem operasi *Linux* yang nantinya digunakan sebagai *server* pada jaringan *IP Multimedia Subsystem* (IMS) dengan layanan pengiriman pesan, panggilan video, panggilan suara, dan pengiriman data berupa *file*. *Proxmox*

melakukan suatu virtualisasi, yang merupakan sebuah teknik untuk membuat *server* komputer secara virtual, atau lebih mudahnya di dalam komputer fisik terdapat beberapa komputer di dalamnya yang menggunakan sistem operasi bermacam-macam yang tidak harus sama dengan *server* induk

## 2.2 Instalasi dan Konfigurasi Proxmox VE (Virtual Environment)

Proses pemasangan dan pengaturan proxmox hanya bisa dilakukan untuk *personal computer* dengan *processor* 64 bit, pada proses instalasi, proxmox akan menggunakan seluruh ruang penyimpanan yang ada, pembuatan partisi akan ditangani secara otomatis dan ruang penyimpanan yang digunakan diformat total, maka perlu diperhatikan untuk mem-*backup* dahulu data sebelum memulai instalasi. Proses virtualisasi di Proxmox VE dilakukan melalui penjelajah web dengan mengakses alamat IP dari Proxmox VE yang sudah dikonfigurasi sebelumnya, pada tugas akhir ini alamat IP Proxmox VE adalah 10.31.17.163 dan akan dibuat virtualisasi berupa sistem operasi Linux yang nantinya digunakan sebagai *server* pada jaringan *IP Multimedia Subsystem* (IMS).

## 2.3 Instalasi IMSCore

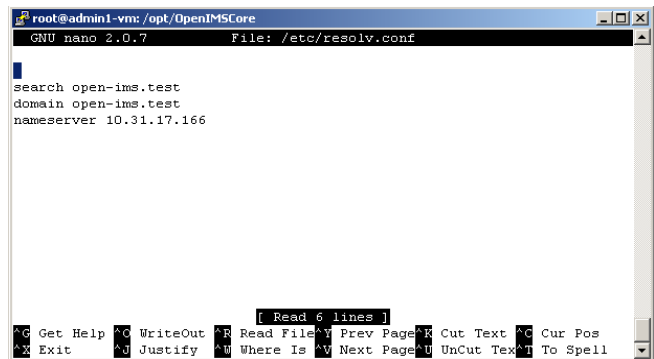
IMS *core* berfungsi untuk menerima permintaan dari klien, memeriksa apakah klien tersebut sudah terdaftar dan mempunyai hak untuk mengakses layanan yang dia minta atau tidak, meneruskan permintaan dari klien ke *application server*, dan selanjutnya mengembalikan jawaban dari *application server* ke klien.

Perancangan layanan *instant messaging* berbasis IMS menggunakan OpenIMSCore sebagai aplikasi dari *server* yang sifatnya bebas pakai. Instalasi IMS Core dilakukan menggunakan *virtual server Proxmox* yang telah di instal dan dikonfigurasi sebelumnya.

## 2.4 Konfigurasi Domain Name System IMSCore

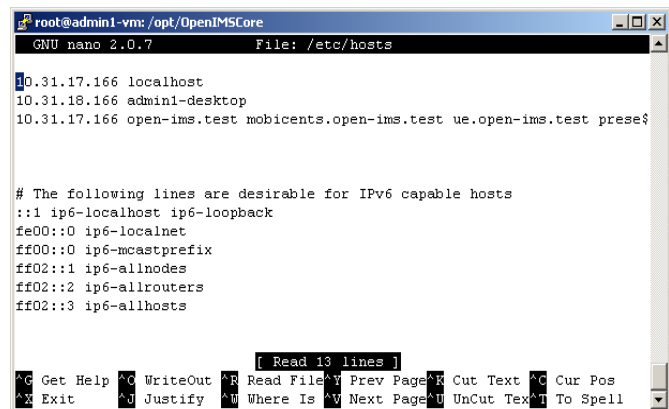
Konfigurasi DHCP dan DNS dapat dilakukan berdasarkan IP server yang hendak digunakan dan juga bergantung kepada keberadaan DNS, Langkahnya adalah sebagai berikut

1. Penggantian DNS Open IMS Core dari *localhost* ke spesifik IP. Untuk mengubah Open IMS Core ke spesifik IP maka kita perlu mengganti semua IP 127.0.0.1 dengan IP sistem, dalam sistem ini digunakan IP 10.31.17.166 dan mengganti DNS menjadi open-ims.test sesuai dengan nama sistem. Pengaturan dan konfigurasi DNS ini membutuhkan perintah – perintah dibawah ini  
#sudo nano /etc/resolv.conf



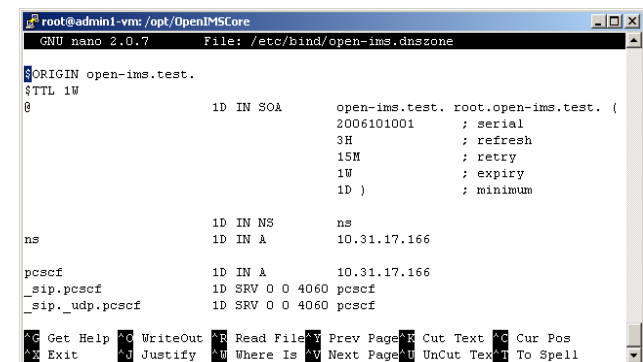
Gambar 3. Tampilan file *Resolv.conf* yang diubah ke spesifik IP

#sudo nano /etc/hosts



Gambar 4. Tampilan alamat IP dan DNS dalam *fileHosts*

#sudo nano /etc/bind/open-ims.dnszone



Gambar 5. Tampilan Pengaturan IP dan DNS file *open - ims.dnszone*

2. Pengaturan DNS dan alamat IP dari *node-node* dengan memanfaatkan *scriptconfigurator.sh* bawaan dari *Open IMS Core*. Hal ini karena pada prinsipnya IMS dijalankan oleh setiap titik / *node* yang saling berhubungan. Karena itu setiap titik diharuskan mempunyai IP dan DNS yang sama. Diperlukan perintah – perintah di bawah ini

```
#cd /opt/Open IMS Core/
#./configurator.sh pcsf.cfg icscf.cfg
icscf.xml scscf.cfg scscf.xml
#./configurator.sh
ser_ims/cfg/icscf.sql
#./configurator.sh
FHoSS/deploy/DiameterPeerHSS.xml
#./configurator.sh
FHoSS/deploy/hss.properties
#./configurator.sh
FHoSS/scripts/hss_db.sql
#./configurator.sh
FHoSS/scripts/userdata.sql
```

Setelah menjalankan perintah tersebut maka pengguna akan diminta memasukkan alamat IP dan DNS. Alamat IP yang digunakan adalah 10.31.17.166 dan DNS nya adalah *open-ims.test*.

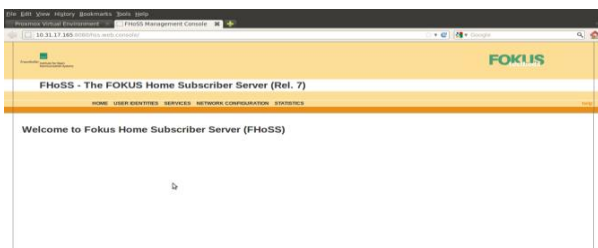
3. Pengaturan DNS dan alamat IP pada setiap titik dan file akan berefek pada aplikasi *bind* yang berguna untuk mensinkronisasi DNS dan alamat IP, karena itu di perlukan langkah *restart bind*. Perintah pengaturan yang diperlukan adalah sebagai berikut  
`#sudo /edit/init.d/bind9 restart`  
`#dig open-ims.test`

Pada *answer section* harus tertulis spesifik IP 10.31.17.166

4. Pengaturan IP dan DNS berpengaruh terhadap database yang digunakan oleh setiap *node*, karena itu diperlukan pengaturan ulang terhadap database. Perintah yang digunakan adalah sebagai berikut

```
#mysql -uroot -p </opt/Open IMS
Core/ser_ims/cfg/icscf.sql
# mysql -uroot -p </opt/Open IMS
Core/FHoSS/scripts/hss_db.sql
# mysql -uroot -p </opt/Open IMS
Core/FHoSS/scripts/userdata.sql
```

5. Pengaturan laman web dari FHoSS yang merupakan web server dari sistem yang berfungsi mengatur database pengguna dan mensinkronkan dengan alamat jaringan IMS yang digunakan. Alamat IP yang digunakan dari web server adalah 10.31.17.165:8080. Tampilan laman dari FHoSS terdapat pada Gambar 6



Gambar 6. Tampilan awal FHoSS

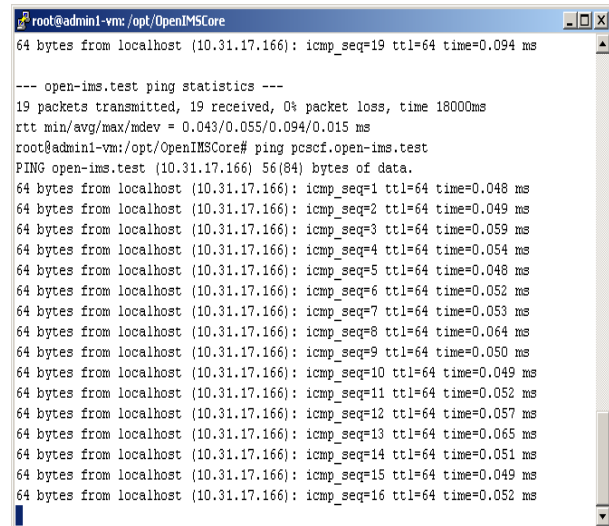
### 3. Hasil dan Analisa

#### 3.1 Pengujian Komponen IMS

Pengujian komponen IMS adalah menguji sambungan server IMS dan komponen didalamnya apakah sudah

sesuai dengan *Domain Name Server* yang ditentukan yaitu *open-ims.test* dengan alamat IP 10.31.17.166

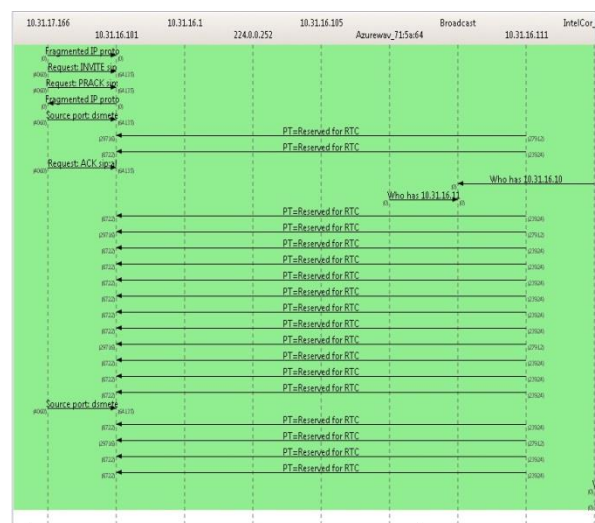
1. Pengujian dengan melakukan perintah ping (*packet internet groper*). Pengujian ini ditujukan untuk menguji sambungan server IMS seperti terlihat pada Gambar 7



Gambar 7. Tampilan pengujian ping server

Dari Gambar 7 terlihat PCSCF sudah terintegrasi dengan *localhost* sehingga sudah siap dipakai sebagai server IMS.

2. Pengujian Sambungan Tunggal Antar Pengguna dan Server. Pengujian sambungan antara pengguna dan server bertujuan untuk mengetahui apakah jalur komunikasi dasar telah dibangun dengan baik. Proses pensinyalan protokol dan pengiriman paket data berupa tampilan grafik bisa diperoleh dengan menggunakan aplikasi Wireshark. Gambar 8 menunjukkan proses pensinyalan selama panggilan dilakukan.



Gambar 8. Tampilan pengujian sambungan tunggal antar pengguna

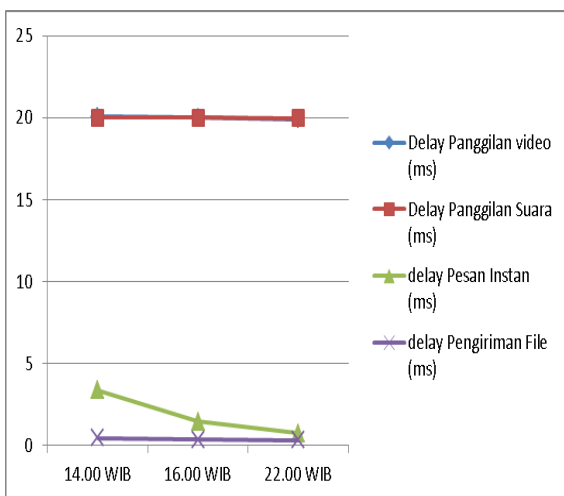
Dari Gambar 8 diketahui ada dua alamat IP pengguna, yaitu 10.31.16.101 dan 10.31.16.111. Satu alamat server yaitu 10.31.17.166. Antara alamat IP 10.31.16.101 dan 10.31.16.111 terjadi pertukaran sinyal dan pengiriman paket data. Pada mulanya perangkat pengguna dengan alamat IP 10.31.16.101 mengirimkan sinyal SIP *INVITE* ke alamat IP 10.31.17.166 yang berada di dalam lingkungan *server*. *Server* kemudian mengirimkan status permintaan *PRACK* yang berarti otentifikasi diterima namun dalam lingkup lingkungan terbatas. Perangkat pengguna dengan alamat 10.31.16.111 lalu mengirimkan sinyal panggilan kepada perangkat pengguna 10.31.6.101 dan kemudian diteruskan kepada *server*. *Server* mengenali alamat dari pengguna yang telah terdaftar lalu mengirimkan sinyal *ACK* yang berisi informasi otentifikasi.

### 3.2 Pengujian Kualitas Layanan

Pengujian kualitas layanan *instant messaging* berbasis *IP Multimedia Subsystem* menggunakan *virtual server* dilakukan dengan menerapkan suatu rangkaian skenario panggilan. Hal ini dilakukan untuk menguji kualitas jaringan yang digunakan. Beberapa skenario panggilan tersebut adalah sebagai berikut

1. Panggilan video tunggal antar pengguna
2. Panggilan suara tunggal antar pengguna
3. Pengiriman pesan instan antar pengguna
4. Pengiriman data berupa *file* antar pengguna

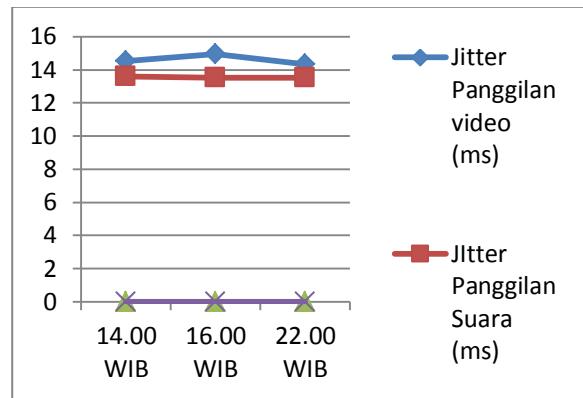
Panggilan dilakukan pada waktu – waktu tertentu yaitu pada pukul 14.00 WIB, pada pukul 16.00 WIB, dan pada pukul 22.00. *Wireshark* mencatat arus panggilan setiap kali skenario panggilan dilakukan kemudian mengetahui nilai dari parameter *Quality of Service* (QoS) yaitu nilai *delay*, *jitter*, *packet loss* dan *Throughput*. Hasil perbandingan nilai *delay* pada layanan dapat dilihat pada Gambar 9



Gambar 9. Grafik perbandingan nilai *delay*.

Dari grafik pada Gambar 9 dapat diambil kesimpulan bahwa *delay* pada layanan masih termasuk dalam kategori baik menurut standar dari ITU-T G.1010. Nilai *delay* semakin turun di waktu sore dan malam hari dikarenakan beban jaringan semakin sedikit di waktu selain jam kerja.

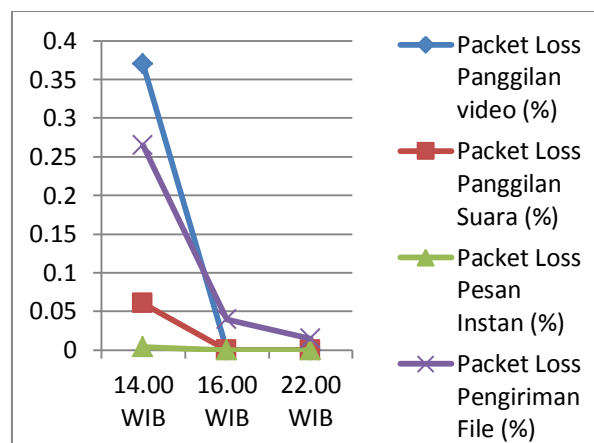
Hasil perbandingan nilai *Jitter* pada layanan dapat dilihat pada Gambar 10



Gambar 10. Grafik perbandingan nilai *Jitter*.

Dari grafik pada Gambar 10 dapat diambil kesimpulan bahwa *jitter* pada layanan panggilan suara dan panggilan video berada di luar standar ITU-T G1010. Standar *jitter* menurut ITU-T G1010 adalah <1ms. Hal ini dikarenakan adanya tabrakan antar paket data yang ada pada jaringan lokal yang digunakan. Nilai *Jitter* semakin turun di waktu sore dan malam hari dikarenakan beban jaringan semakin sedikit di waktu selain jam kerja.

Hasil perbandingan nilai *Packet loss* pada layanan dapat dilihat pada Gambar 11



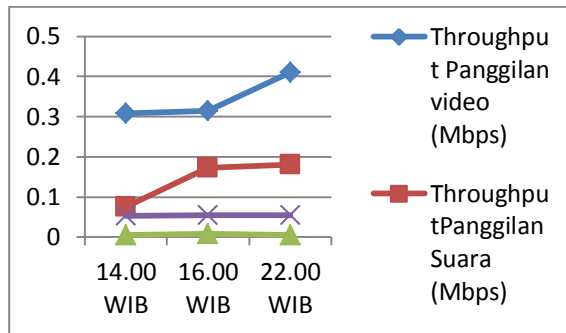
Gambar 11. Grafik perbandingan nilai *packet loss*.

Dari grafik pada Gambar 11 dapat diambil kesimpulan bahwa *jitter* pada layanan masih termasuk dalam kategori baik menurut standar dari ITU-T G.1010. Nilai grafik *packet*



loss semakin turun dikarenakan penggunaan beban jaringan yang semakin sedikit di sore dan malam hari.

Hasil perbandingan nilai *Throughput* pada layanan dapat dilihat pada Gambar 12



Gambar 12. Grafik perbandingan nilai *Throughput*.

Dari grafik pada Gambar 12 dapat diambil kesimpulan bahwa *throughput* pada layanan masih termasuk dalam kategori baik menurut standar dari ITU-T G.1010. Nilai grafik *throughput* semakin naik dikarenakan penggunaan beban jaringan yang semakin rendah pada sore dan malam hari.

#### 4. Kesimpulan

Dari hasil implementasi layanan *Instant Messaging* berbasis *IP Multimedia Subsystem* berbasis *virtual server* dan pengujian dapat diambil beberapa kesimpulan sebagai berikut. Sistem Layanan *Instant Messaging* berbasis *IP Multimedia Subsystem* dapat diimplementasikan menggunakan server virtual dan dapat bekerja di lingkungan jaringan kampus Teknik Elektro Universitas Diponegoro. Sistem layanan yang dibangun telah memenuhi sebagian besar parameter kelayakan, parameter QoS yang digunakan adalah standar dari ITU-T G1010. Kemudian Penggunaan terbaik layanan adalah ketika jumlah pengguna menurun, dalam pengujian didapatkan waktu terbaik adalah saat malam hari. Selain itu Kinerja dari layanan di pengaruhi oleh penggunaan beban jaringan yang ada pada sistem, dalam hal ini menggunakan jaringan Fakultas Teknik Universitas Diponegoro. Kinerja dan konfigurasi dari perangkat router dan perangkat nirkabel *wi-fi* sangat mempengaruhi dari kinerja layanan, hal ini dikarenakan dalam menghubungkan pengguna memerlukan sambungan menggunakan kabel dan nirkabel. Adapun saran yang dapat diberikan untuk menjadi masukan pada penelitian lebih lanjut adalah Sistem *IP Multimedia Subsystem* sebaiknya diimplementasikan dengan menggunakan perangkat yang berdiri sendiri yang secara khusus bekerja mendukung kinerja yang terbaik. Pengguna sebaiknya menggunakan *softphone* berbasis android atau windows phone 8 yang terintegrasi dengan layanan IMS untuk

melakukan komunikasi yang lebih mudah. Sebaiknya dilakukan penambahan fitur yang ada pada IMS, dapat ditambahkan fitur *charging set* dan *call session control function* yang lebih baik. Pada server IMS, dapat ditambahkan fitur *video streaming*, dan IPTV (*Internet Protocol Television*).

#### Referensi

- [1]. --. *IMS, IP Multimedia Subsystem*. <http://www.3gpp.org/Technologies/Keyword-Acronyms/article/ims>, 27 Agustus 2013.
- [2]. --. Open Source Ims Core. <http://www.openimscore.org/>, diakses pada 29 Agustus 2013.
- [3]. --. OSIMS The FOKUS Open Source IMS. [http://www.fokus.fraunhofer.de/en/fokus\\_tesbeds/open\\_ims\\_playground/components/osims/index.html](http://www.fokus.fraunhofer.de/en/fokus_tesbeds/open_ims_playground/components/osims/index.html), diakses pada 28 Agustus 2013.
- [4]. Godbole, Achyut S. *Data Communications and Networks*. 2003. Singapura : McGraw Hill Co.Ltd.
- [5]. Ismail, Nanang. Yusep R. Armein Z.R *Ip Multimedia Subsystem (IMS) Muculnya Peluang Bisnis Baru*. 2006. Bandung : Konferensi Nasional Teknologi Informasi & Komunikasi untuk Indonesia, Institut Teknologi Bandung.
- [6]. Kurniawan, Agus. *Network Forensics Panduan Analisis dan Investigasi Paket Data Jaringan menggunakan Wireshark*. 2012. Yogyakarta : ANDI.
- [7]. Monfort, Jean-Yves. *Basic Requirement to Quality of Service (IP centric)*. 2003. Swiss : Workshop on Standardization E-Health
- [8]. Nanda, S., and T. Chiueh, A Survey On Virtualization Technologies, Brookhaven, New York: Stony Brook University, Research Preliminary Exam Report, 2005.
- [9]. Poikselka, Miikka. Georg Mayer. *The IMS: IP Multimedia Concepts and Services*. 2009. United Kingdom : John Wiley and sons, Ltd.
- [10]. Proxmox Server Solution, *Proxmox Network Model*, Vienna: Proxmox ServerSolution GmbH, 2013, [http://pve.proxmox.com/wiki/Network\\_Model](http://pve.proxmox.com/wiki/Network_Model), 29 Agustus 2013.
- [11]. Proxmox Server Solution, *Proxmox Storage Model*, Vienna: Proxmox Server Solution GmbH, 2013, [http://pve.proxmox.com/wiki/Storage\\_Model](http://pve.proxmox.com/wiki/Storage_Model), diakses pada 29 Agustus 2013.
- [12]. Proxmox Server Solution, *Proxmox Virtual Enviroment Datasheet*, Vienna: Proxmox Server Solution GmbH, 2013, Tersedia pada: <http://www.proxmox.com/downloads/proxmox-ve/misc/116-proxmox-ve-datasheet/download>, 29 Agustus 2013.
- [13]. Rittinghouse, John W and James F. Ransome. *Instant Messaging Security*. . United States of America : Elsevier Digital Press.
- [14]. Russel, Travis. *THE IP MULTIMEDIA SUBSYSTEM (IMS) Session Control and Other Network Operations*. 2008. United States of America : The McGraw-Hill Companies.
- [15]. Siop, Mamadou. Boghe IMS Client. <https://code.google.com/p/boghe/>, diakses pada 29 Agustus 2013
- [16]. Sofana, Iwan. *Cisco CCNA dan Jaringan Komputer*. 2012. Bandung : Informatika.
- [17]. Szigeti, Tim. Christina Hattingh. *End-to-End QoS Network Design: Quality of Service in LANs, WANs, and VPNs*. 2004. Amerika Serikat : Cisco Press