

## RANCANG BANGUN *MAIL SERVER* DALAM JARINGAN LOKAL MENGGUNAKAN *EMBEDDED SYSTEM*

Moh. Halimi<sup>#1</sup>, Barlian Henryranu P<sup>\*2</sup>, Gembong Edhi Setiawan<sup>\*2</sup>

<sup>1</sup>Mahasiswa, <sup>2</sup>Dosen Pembimbing  
Program Studi Teknik Informatika

Program Teknologi Informasi dan Ilmu Komputer  
Universitas Brawijaya Malang

Email: mohhalimi@gmail.com

### ABSTRACT

Method of mail delivery and electronic documents has become a necessity for everyone to keep up with technology and a server that serves service delivery and retrieval of e-mail. Mail server is a server that most widely used in the company. It is given its own email function that can reduce the cost of correspondence, communication and more efficient than the manual and it's can include attachments that useful as a complementary and additional documents related to the content of the email. Raspberry Pi type B was used for mail server implementation and for user interface of the mail server used Citadel application. This research has two results. First, mail server was capable of serving an email communication until amount of data is 20 MB with average transfer rate value is 505.7 KBps. Second, In once delivery email process using mail server embedded system obtained 100% value of data integrity for each parameter testing. The implementation of the mail server using the embedded system has purpose to ease the workload of mail server's master. The mail server will serve requests from the user who only require an email communication on a local network so it does not need to access the mail server's master.

### ABSTRAK

Metode pengiriman surat maupun dokumen secara elektronik sudah menjadi kebutuhan bagi setiap orang guna mengikuti perkembangan teknologi. *Mail Server* merupakan *server* yang melayani layanan pengiriman dan pengambilan *e-mail*. *Mail server* merupakan salah satu fungsi *server* yang paling banyak digunakan oleh perusahaan maupun instansi. Hal ini mengingat fungsi email sendiri yang bisa mengurangi biaya surat-menyurat, lebih efisien dibandingkan komunikasi manual dan dapat menyertakan *attachment* yang berguna sebagai pelengkap dan dokumen tambahan terkait dengan isi email. Guna meringankan beban kerja dari *mail server* maka dilakukan penelitian mengenai implementasi dan analisa performansi *mail server* menggunakan *embedded system*. Mini komputer yang digunakan adalah Raspberry Pi model B dan sebagai tampilan antar muka pengguna *mail server* menggunakan perangkat lunak Citadel. Hasil yang didapatkan adalah *mail server* mampu melayani komunikasi email hingga besar data keseluruhan 20 MB dengan nilai *transfer rate* rata – rata 505.7 KBps. Dalam satu kali pengiriman email menggunakan *mail server embedded system*, didapatkan nilai integritas data 100% untuk setiap parameter pengujian. *Mail server* ini nantinya

diharapkan dapat melayani permintaan dari pengguna atau *user* yang hanya memerlukan komunikasi email pada jaringan lokal sehingga tidak perlu mengakses *mail server master* yang ada.

**Kata kunci :** *Mail server, Embedded system, Performansi, Raspberry Pi, Citadel*

## 1. Pendahuluan

Seiring dengan kemajuan teknologi, *email* merupakan salah satu media komunikasi sebagai pengganti surat. Menurut Narendro Arifia [NAE-13], dalam tulisannya menyebutkan *electronic mail* (surat elektronik) sering disebut e-mail atau email, merupakan metode *store and forward* dari menulis, mengirim, menerima dan menyimpan surat melalui sebuah sistem komunikasi elektronik. Surat dipertukarkan antar host melalui *Simple Mail Transfer Protocol* dengan software program : *Mail Transport Agents* (MTA). *User* dapat mendownload surat mereka dari server dengan protokol standard seperti POP atau IMAP, atau di lingkungan perusahaan besar dengan protokol khusus seperti *Lotus Notes* atau *Microsoft Exchange Servers*. Surat dapat disimpan pada *client*, pada *server*, atau keduanya. Sebagai penunjang untuk melakukan komunikasi melalui *email* dibutuhkan sebuah *server* yang khusus untuk melayani proses pengiriman surat elektronik yang disebut dengan *mail server*.

Sebagai penunjang untuk melakukan komunikasi melalui *email* dibutuhkan sebuah *server* yang khusus untuk melayani proses pengiriman surat elektronik yang disebut dengan *mail server*. I. E. S. W. Mangunkusumo dalam tulisannya [MIE-13] menyatakan *Mail server* merupakan salah satu fungsi *server* yang paling banyak digunakan di perusahaan. Hal ini mengingat fungsi *email* sendiri yang bisa mengurangi biaya surat-menyurat, lebih efisien dibandingkan komunikasi manual dan dapat menyertakan *attachment* yang berguna sebagai pelengkap dan dokumen tambahan terkait dengan isi email. Tuntutan teknologi juga yang menyebabkan setiap perusahaan maupun universitas harus mempunyai *mail server* sendiri sehingga menjadi lebih efisien.

untuk meminimalkan penggunaan sumber daya saat ini terdapat beberapa sistem yang mampu menjalankan tugas yang bersifat khusus atau spesifik sesuai dengan kebutuhan yang diharapkan. Salah satu sistem yang memiliki fungsi tersebut adalah *embedded system*. Menurut Steve Heath [HSE-03], Sebuah sistem *embedded* adalah sistem berbasis mikroprosesor yang dibangun untuk mengontrol fungsi atau jangkauan fungsi dan tidak dirancang untuk diprogram oleh pengguna akhir dengan cara yang sama pada penggunaan PC .

pada kesempatan ini penulis akan membangun serta menganalisa sistem *mail server* berbasis *embedded system* dengan menggunakan *raspberry pi*. Penelitian ini bertujuan untuk menghasilkan sebuah *mail server* untuk komunikasi surat elektronik guna membantu meringankan beban kerja *master server*. Parameter yang digunakan untuk menganalisa performansi *mail server* adalah nilai *transfer rate* dan besar kapasitas data yang bisa dikirimkan melalui *mail server* tersebut.

## Rumusan Masalah

Berdasarkan latar belakang yang dipaparkan maka rumusan masalah yang perlu diperhatikan adalah :

1. Bagaimana perancangan dan implementasi *mail server* pada *Embedded System*.
2. Pengujian ditekankan pada *transfer rate*, besar data dan integritas data yang dapat dikirimkan menggunakan *mail server* pada *embedded system*.

## Batasan Masalah

Batasan masalah dapat diklasifikasikan sebagai berikut :

1. Pembahasan difokuskan mengenai analisa performansi *mail server* menggunakan *embedded system* yang berkaitan dengan kecepatan pengiriman data dan besar data yang mampu dikirimkan.
2. Perancangan *mail server* menggunakan perangkat lunak Citadel.
3. Perangkat keras yang digunakan dalam penelitian ini adalah mini komputer *raspberry pi* model B.
4. Jumlah pengguna yang terdaftar pada *mail server* adalah 50 pengguna.

## Tujuan

Tujuan penulisan tugas akhir ini adalah untuk merancang dan menganalisa *mail server* pada *embedded system* menggunakan *raspberry pi* guna membantu performansi *mail server master*.

## Manfaat

Penelitian mengenai *mail server* pada *embedded system* mempunyai beberapa manfaat yakni mengurangi beban kerja dari *mail server master* dan hanya memerlukan konsumsi daya yang sedikit. Hasil penelitian ini diharapkan dapat dijadikan sebagai bahan referensi dan tambahan pengetahuan mengenai *mail server* menggunakan *embedded system Raspberry Pi*.

## 2. KAJIAN PUSTAKA DAN DASAR TEORI

### A. Kajian Pustaka

Salah satu penelitian yang membahas mengenai implementasi perangkat citadel sebagai *mail server* dilakukan oleh Fahmi Rajab Febrian [RFF-11], berjudul “Building a Mail Server Using Mail Citadel Transfer Agent on Linux Ubuntu 10:10 and accessing With Mozilla Thunderbird”. Penelitian ini membahas tentang implementasi Citadel sebagai *mail server* pada sistem operasi Linux dan menggunakan aplikasi Mozilla Thunderbird pada sisi pengguna. Hasil dari penelitian ini adalah sebuah layanan komunikasi *email* pada sebuah perusahaan, dimana hanya pihak admin yang berwenang untuk menentukan pengguna yang berhak menggunakan layanan ini.

Penelitian lain membahas mengenai performansi dari *mail server* dilakukan oleh Jun Wan dan Yiming Hu [WJY-04], berjudul “A Performance Study on Internet Server Provider Mail Servers”. Penelitian ini membahas tentang performansi dari sebuah *mail server*. Masalah yang dibahas pada penelitian ini dibagi pada tiga hal, pertama adalah mengamati *session* dari protokol SMTP dan POP mengenai respon *server* pada awal koneksi ketika mengakses sebuah *mail server*. Kedua adalah mengenai pengaruh dari perangkat I/O atau *Input/output* terhadap total waktu transfer ketika proses permintaan *email* terutama pada *server* dengan pengguna yang besar. Ketiga adalah hal yang dapat membuat sebuah *server* menjadi kelebihan beban. Penelitian ini menghasilkan beberapa kesimpulan yakni pada *session* protokol SMTP dan POP ketika proses koneksi awal menuju *mail server* membutuhkan waktu yang lama. Kesimpulan selanjutnya adalah *latency* dari perangkat I/O memiliki kontribusi sebesar 40-55% terhadap proses transfer data pada proses permintaan *email*. Terakhir adalah sebuah kelompok *email* dapat dengan mudah membuat *server* penerima jarak jauh menjadi kelebihan beban.

## B. Dasar Teori

### 1. Email

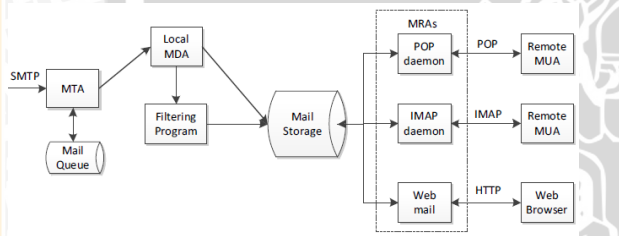
*Electronic mail* atau *email* adalah metode pengiriman surat melalui jaringan internet. Sistematika penggunaan *email* tidak jauh beda dengan surat konvensional. Surat terdiri dari nama pengirim, alamat yang dituju dan isi surat. Perbedaan menggunakan *email* terletak pada media pengiriman yang digunakan yakni menggunakan jaringan internet [RRA-06].

Sebelum pengguna dapat memanfaatkan *email*, terlebih dahulu seorang pengguna harus mendaftarkan dirinya pada sebuah *server* atau *administrator*. Setelah mendaftarkan dirinya pengguna akan mendapatkan *account* yang digunakan untuk masuk pada sebuah layanan *email* dimana pengguna dapat menulis, menerima dan mengirimkan *email* [RRA-06]. Kelebihan yang dimiliki oleh *email* adalah menghemat biaya karena *email* tidak perlu menggunakan kertas sebagai media

penulisannya. *Email* hanya memerlukan sebuah komputer yang tersambung dalam jaringan internet.

## 2. Mail Server

Sebuah *mail server* terdiri dari beberapa komponen yakni MTA (*Mail Transfer Agent*), MDA (*Mail Delivery Agent*) dan MRA (*Message Retrieval Agent*). MTA bertindak sebagai pengirim dan penerima pesan yang melalui sebuah *mail server* melalui SMTP. *Email* yang diterima oleh MTA akan disimpan dalam sebuah antrian kemudian MDA dipanggil untuk mengirimkan dan menyimpan *email* sesuai alamat penerima yang dituju. Ketika pengguna ingin membaca *email* dia dapat masuk atau *log in* menggunakan akun yang sudah terdaftar. MRA dirancang sebagai MUA (*Message User Agent*) dimana pengguna dapat mengakses akunnya dari jarak jauh melalui internet. Sementara IMAP membuat akses yang mungkin sehingga pengguna dapat mengakses *email* yang disimpan dalam sebuah server [LMJ-11].



**Gambar 2.1** Arsitektur mail server

Sumber : [LMJ-11]

## 3. Citadel

Citadel merupakan perangkat dari SpamExile yang menyediakan kita solusi untuk semua persyaratan email yang kita butuhkan, teknologi email menyatu dengan layanan SpamExile yang unik. SpamExile menggantikan semua teknologi yang ada pada saat ini dan merevolusi industri anti-spam. Menghentikan spam pada sumber yang benar, menghentikan *spam* yang berasal dari pengirim spam sejak awal terdeteksi spam. Citadel secara dinamis membangun blacklist server sendiri dan memeriksa sumber yang tidak diketahui terhadap server RBL (*Real-time Block Lists*) pada setiap sambungan eksternal. Server yang sudah terdaftar yang mencoba untuk koneksi ke citadel server akan ditolak secara

aktif, karena semua pengguna citadel menerima update otomatis melalui topologi jaringan komunitas [EXS].

## 4. Embedded System

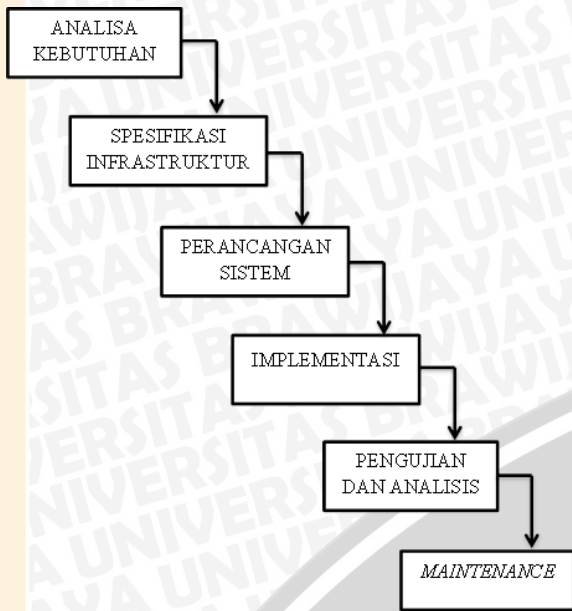
Raj kamal [KAR-08], dalam bukunya yang berjudul *Embedded System* menerangkan bahwa *Embedded system* adalah sebuah sistem memiliki perangkat lunak tertanam pada *hardware* atau perangkat keras dan dapat didedikasikan untuk sebuah aplikasi atau menjadi bagian tertentu dari sebuah aplikasi. *Embedded system* juga dapat berfungsi sebagai bagian dari sebuah sistem yang besar. Perangkat lunak dapat ditanamkan pada ROM (*Read Only Memory*) atau sebuah *removable disk* seperti *flashdisk* dan *SD card memory*. *Embedded system* memiliki tiga komponen utama yakni:

1. *Hardware* atau perangkat keras yang memiliki fungsi seperti layaknya komputer pada umumnya
2. Melekatkan perangkat lunak aplikasi utama ke dalam flash atau ROM dan aplikasi perangkat lunak memproses beberapa tugas.
3. Memiliki sistem operasi real time (RTOS), yang mengawasi tugas software aplikasi yang berjalan pada perangkat keras dan mengatur akses ke sumber daya sistem sesuai dengan prioritas dan waktu kendala tugas dalam sistem.

## 3. METODOLOGI PENELITIAN DAN PERANCANGAN

### A. Metodologi Penelitian

Bab ini menjelaskan langkah-langkah yang akan ditempuh dalam penyusunan skripsi, yaitu perancangan, implementasi dan pengujian dari aplikasi perangkat lunak yang akan dibuat. Kesimpulan dan saran disertakan sebagai catatan atas aplikasi dan kemungkinan arah pengembangan aplikasi selanjutnya.



**Gambar 3.1** Diagram Alir Metode Penelitian

Penjelasan :

1. Analisa Kebutuhan

Analisa kebutuhan dilakukan dengan cara mengumpulkan dan mempelajari segala macam informasi yang berhubungan dengan *citadel mail server*, mini komputer *raspberry pi* dan segala hal yang berhubungan dengan hal – hal mengenai *mail server* pada *embedded system*.

2. Spesifikasi Infrastruktur

Pada tahap spesifikasi infrastruktur ini dilakukan proses pemilihan perangkat yang akan digunakan dalam implementasi *mail server* pada *embedded system*. Spesifikasi infrastruktur meliputi pemilihan perangkat lunak dan perangkat keras yang akan digunakan.

3. Perancangan Sistem

Tahap ini dilakukan guna merancang sistem yang akan dibuat berdasarkan hasil dari analisa kebutuhan mengenai *mail server* pada *embedded system*. Setelah menentukan jenis kebutuhan yang diperlukan, tahap selanjutnya adalah melakukan perancangan *mail server* pada *embedded system* menggunakan perangkat mini komputer Raspberry Pi.

4. Implementasi

Dalam tahap ini, dilakukan implementasi berdasarkan studi literatur dan analisa kebutuhan yang telah dibuat

pada tahap sebelumnya. Implementasi *mail server* meliputi:

- a. Instalasi sistem operasi *raspbian wheezy* sebagai *user interface* dalam menggunakan perangkat *raspberry pi*.
- b. Instalasi dan konfigurasi perangkat *citadel* yang digunakan sebagai *mail server*.

5. Pengujian dan Analisis

Pengujian pada sistem dilakukan untuk menunjukkan bahwa sistem mampu bekerja sesuai dengan spesifikasi kebutuhan. Pada tahap ini dilakukan pengujian mengenai performansi dari *mail server* dengan dua skenario. Skenario pertama dilakukan guna mengetahui nilai *transfer rate* dari *mail server* dengan menggunakan aplikasi sebagai alat bantu uji. Skenario kedua berfungsi untuk mendapatkan nilai integritas data yang dikirimkan menggunakan *mail server* pada *embedded system*.

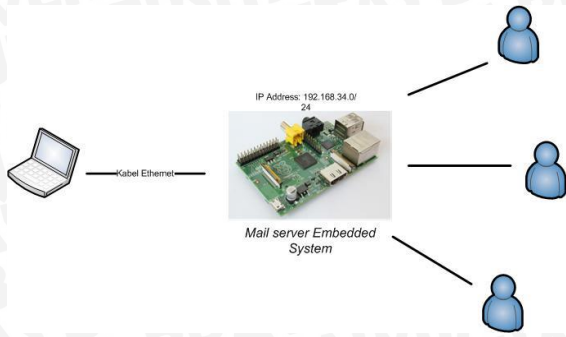
6. Maintenance

Pada tahap ini dilakukan proses *maintenance* atau pemeliharaan pada sistem. Pemeliharaan yang dilakukan adalah proses perubahan konfigurasi dari sistem guna mendapatkan sebuah sistem yang dikehendaki. Tahap ini berjalan seiring dengan proses implementasi *mail server* pada *embedded system*. Proses yang dilakukan yakni mengenai perubahan konfigurasi *webcit* yang berfungsi sebagai tampilan antar muka pengguna pada *web browser* dan proses konfigurasi *IP address* dari mini komputer Raspberry Pi.

**B. Perancangan**

**1. Topologi Perancangan**

Komputer penguji digunakan untuk mengakses *mail server embedded system* sehingga dapat melihat tampilan antar muka dari *mail server* guna mempermudah proses pengujian. *Mail server* yang ditanamkan pada mini komputer Raspberry Pi diakses menggunakan kabel *Ethernet* pada skala jaringan lokal.



Gambar 3.2 Topologi Perancangan

#### 4. IMPLEMENTASI

##### A. Spesifikasi Lingkungan Infrastruktur

Spesifikasi mengenai perangkat keras dan perangkat lunak yang digunakan dalam perancangan *mail server embedded system* menggunakan mini komputer *raspberry pi* adalah sebagai berikut.

Tabel Error! No text of specified style in document..1 Spesifikasi Lingkungan Perangkat Keras

Perangkat keras	Spesifikasi
CPU / Processor	ARM1176JZFS 700 Mhz
Memory (RAM)	512 Mb
Storage	SDHC Card 8 Gb
Network Interface	Ethernet LAN

Tabel Error! No text of specified style in document..2 Spesifikasi Lingkungan Perangkat Lunak

Perangkat Lunak	Spesifikasi
Sistem Operasi	Linux Raspbian wheezy
Perangkat Lunak <i>mail server</i>	Citadel

##### B. Implementasi Raspberry Pi

Tahapan penelitian selanjutnya yang dilakukan adalah proses implementasi pada perangkat *raspberry pi* yang digunakan sebagai *mail server*. Langkah pertama yang dilakukan adalah instalasi sistem operasi pada perangkat *raspberry pi*. Sistem operasi yang digunakan adalah Linux *raspbian wheezy*. *Raspbian wheezy* digunakan karena sistem operasi ini *compatible* atau sesuai digunakan pada perangkat

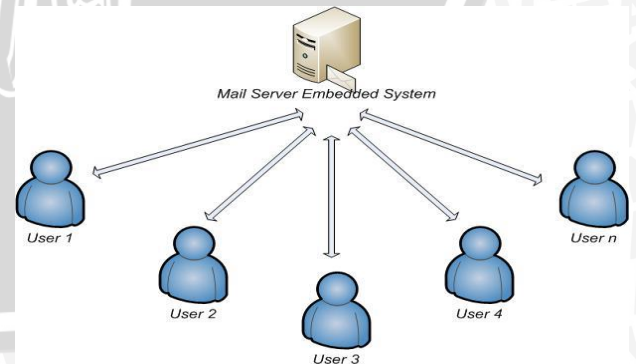
mini komputer *raspberry pi*. Sistem operasi digunakan untuk menampilkan GUI atau tampilan tatap muka pengguna guna proses instalasi dan konfigurasi *mail server*. Berikut adalah tampilan sistem operasi *raspbian wheezy* menggunakan aplikasi *tightvnc*.

##### C. Implementasi Citadel

Citadel adalah aplikasi yang digunakan sebagai *mail server* sehingga pengguna dapat memanfaatkannya sebagai sarana komunikasi *email*. Citadel mempunyai tampilan tatap muka atau *user interface* berbasis *web*, sehingga pengguna dapat mengakses *mail server* dengan menggunakan *web browser* yang dimiliki. Citadel merupakan aplikasi *open source* sehingga kita dapat mengunduh dan menggunakannya secara gratis. Selain itu kita juga dapat melakukan konfigurasi pada aplikasi ini sesuai dengan kebutuhan.

##### D. Implementasi Ifrastruktur

Setelah melalui tahap analisa kebutuhan dan perancangan, maka tahap selanjutnya adalah tahap implementasi sistem. Implementasi yang dilakukan juga mengacu pada kedua tahap tersebut. *Mail server* akan difungsikan sebagai komunikasi *email* oleh pengguna yang berada pada sebuah jaringan lokal.



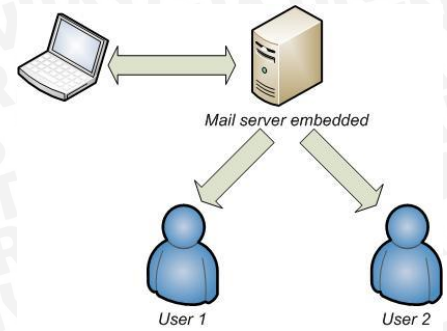
Gambar 4.1 Skema implementasi

Skenario pada gambar diatas dapat dijelaskan sebagai berikut :

1. Pengguna yang akan melakukan komunikasi *email* kepada pengguna lain yang berada pada

jaringan lokal dapat mengakses *mail server embedded system* melalui *browser* sesuai yang dimiliki oleh pengguna.

2. Setiap pengguna diwajibkan mendaftarkan diri untuk memperoleh sebuah akun sebelum dapat memanfaatkan fasilitas *mail server embedded system*.
3. Setelah masing – masing pengguna telah memperoleh akun, maka pengguna dapat masuk ke *mail server embedded system* dan dapat menggunakannya sebagai sarana komunikasi *email* dengan pengguna lain pada jaringan lokal.



Gambar 5.1 Skenario Pengujian pertama

Berikut adalah hasil pengujian pada skenario pertama:

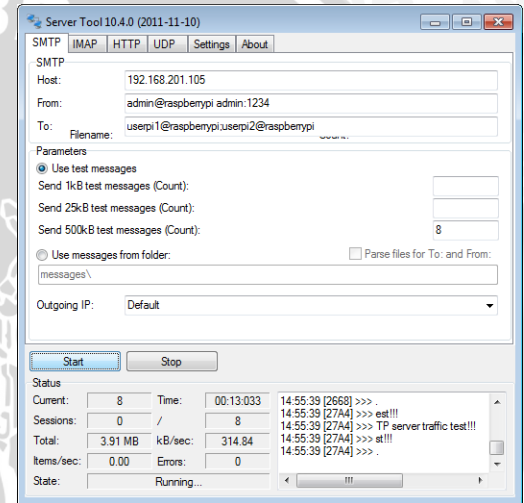
## 5. PENGUJIAN DAN ANALISIS

### A. Pengujian

Tahap pengujian *mail server* ini melalui dua skenario pengujian. Skenario pengujian dibedakan berdasarkan parameter yang digunakan. Pada skenario pertama menggunakan beberapa parameter yakni besar data, *transfer rate* dan waktu pengiriman. Sedangkan pada skenario kedua menggunakan besar data, jumlah pengguna dan integritas data yang dikirimkan sebagai parameter pengujiannya.

#### 1. Skenario Pengujian Pertama

*Mail server* diakses melalui sebuah komputer personal dengan menggunakan jaringan LAN. Personal komputer tersebut telah diinstal aplikasi *ServerTool* 10.4.0 sebagai alat bantu uji dari *mail server embedded* yang diterapkan. Aplikasi ini memiliki beberapa *field* atau kolom yang harus diisi oleh pengguna sebelum melakukan proses pengujian. Kolom *Host* diisi dengan alamat *domain* dari *server* yang akan diuji, kolom *From* mengenai alamat dari pengirim *email* dan kolom *To* dapat diisi dengan alamat *email* yang akan menerima *email* tersebut. Pengguna dapat memilih untuk mengirimkan pesan yang disediakan oleh aplikasi atau pengguna juga dapat memilih *file* yang akan dikirimkan



Gambar 5.2 Screenshot Pengujian

Tabel 5.1 Hasil Pengujian

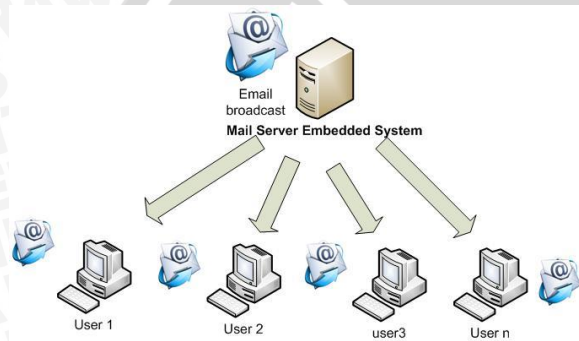
Besar file (MB)	Transfer rate Rata – rata (KBps)	Waktu pengiriman rata – rata (s)
5	275.83	18.14
10	389.1	25.7
15	450.03	33.4
20	505.7	39.6
25	491.4	51
30	482.6	62.3

Pengujian ini dilakukan dalam lingkungan jaringan yang sama untuk setiap proses pengujian. Pengujian dilakukan sebanyak tiga kali percobaan untuk setiap parameter uji yang digunakan. Selanjutnya diambil nilai rata - rata dari tiga nilai yang didapat untuk setiap

parameter uji (Lampiran 2). Hal ini dilakukan untuk meminimalkna varian sehingga *error* atau kesalahan yang dilakukan dapat diminimalisir [RAA-11].

## 2. Skenario Pengujian Kedua

Pada skenario kedua memiliki perbedaan dengan pengujian pada skenario pertama yakni mengenai parameter yang digunakan dan tahapan yang dilakukan pada saat pengujian. Pengujian yang dilakukan pada skenario kedua melalui sepuluh tahap. Tahapan tersebut dibagi berdasarkan besar *file* yakni 1Mb, 2Mb, 3Mb, 4Mb dan 5Mb. Jumlah pengguna yang terdaftar pada saat pengujian sebanyak 50 pengguna. Berikut adalah hasil pengujian pada skenario kedua :



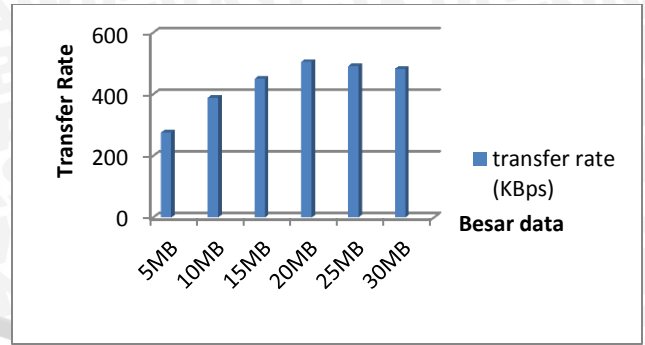
Gambar 5.3 Skenario Pengujian kedua

Tabel 5.2 pengujian skenario kedua

Jumlah User	Besar File	Integritas Data
50	1 x 1 MB	100%
50	2 x 1 MB	100%
50	3 x 1MB	100%
50	4 x 1 MB	100%
50	5 x 1 MB	100%

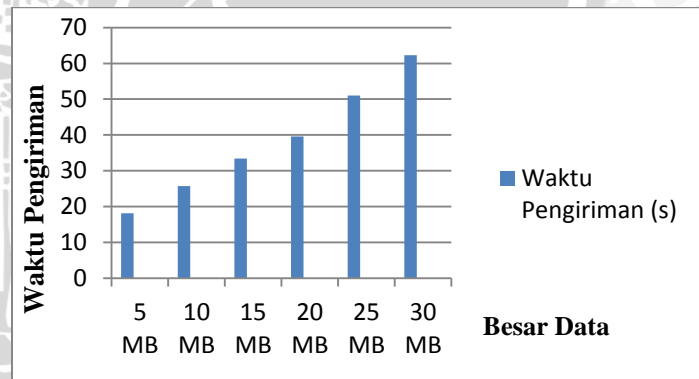
### B. Analisis

Analisis pada pengujian skenario pertama dengan menggunakan aplikasi ServerTool 10.4.0 dapat dilihat pada gambar 5.4 dibawah ini. Pengujian ini bertujuan untuk mengetahui beberapa hal yakni *transfer rate* pada saat mengirim *file* atau data. Gambar dibawah ini didapatkan berdasarkan hasil yang ada pada Tabel 5.1.



Gambar 5.4 Grafik Hasil Pengujian Skenario 1

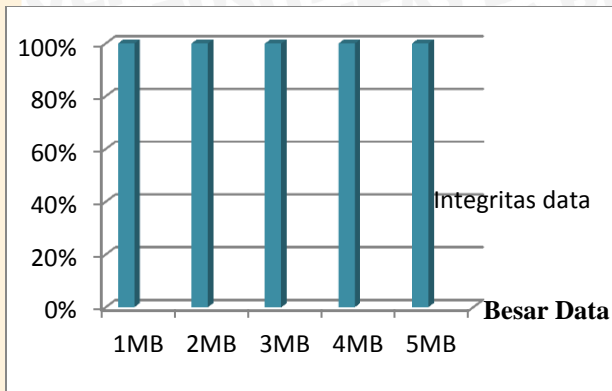
Hasil pengujian pada skenario pertama khususnya pada parameter pengujian ke – 1 sampai ke – 4 menghasilkan nilai *transfer rate* yang selalu meningkat atau sebanding dengan besar *file* yang dikirimkan. Namun nilai tersebut menurun pada parameter pengujian ke – 5, dimana besar *file* yang dikirimkan sebesar 25MB. *Mail server embedded system* mengalami penurunan performa ketika dibebani pekerjaan untuk mengirimkan *file* sebanyak 50 dengan besar data 500 KB kepada dua pengguna. Hal ini dipengaruhi oleh terbatasnya kapasitas *buffer* atau tempat penyimpanan data sementara yang dimiliki oleh perangkat mini komputer Raspberry Pi.



Gambar 5.5 Grafik Waktu Pengiriman

Analisis selanjutnya dilakukan pada skenario pengujian yang kedua dimana pada pengujian ini tidak memerlukan aplikasi sebagai alat bantu pada saat pengujian. Hasil pengujian pada skenario ini dapat dilihat pada gambar 5.6 dibawah ini.





**Gambar 5.6** Grafik hasil pengujian skenario 2

Pengujian pada skenario kedua bertujuan untuk mendapatkan nilai integritas data yang dikirimkan melalui *mail server embedded system*. Hasil yang didapat pada saat pengujian adalah nilai integritas data yang dikirimkan mencapai 100% untuk setiap parameter pengujian yang ada. Hal ini didapatkan berdasarkan pengecekan pada akun email yang menjadi tujuan pada waktu pengujian. nilai presentase untuk setiap parameter pengujian didapatkan dengan menggunakan persamaan sebagai berikut :

$$\text{Nilai integritas data} = \frac{\text{jumlah data yang terkirim}}{\text{jumlah data yang dikirimkan}} \times 100 \%$$

## 6. KESIMPULAN

Berdasarkan hasil perancangan, implementasi, pengujian dan analisis yang telah dilakukan, maka diambil kesimpulan sebagai berikut :

1. Perangkat mini komputer Raspberry Pi dapat digunakan sebagai *mail server embedded system*.
2. Perangkat lunak Citadel dapat digunakan sebagai *mail server* pada spesifikasi default perangkat mini komputer Raspberry Pi ( CPU 700MHz + RAM 512MB).
3. Pengujian mengenai integritas data pada *mail server embedded system* menggunakan lima parameter uji. Nilai presentase data yang didapat untuk setiap parameter pengujian yang ada adalah 100%. Dengan kata lain data yang terkirim pada alamat email yang dituju sesuai dengan data yang dikirimkan oleh email pengirim.
4. *Mail server embedded system* menggunakan perangkat mini komputer Raspberry Pi telah diuji menggunakan

aplikasi ServerTool 10.4.0. Hasil yang didapat adalah *mail server embedded system* secara optimal dapat melakukan pengiriman *file* dengan *transfer rate* rata – rata 505.7 KBps pada besar *file* 20MB. Sedangkan ketika besar *file* yang dikirimkan > 20MB nilai *transfer rate* yang didapatkan lebih kecil dibandingkan dengan pengiriman *file* sebesar 20MB.

## 7. Saran

1. Infrastruktur ini dapat dimodifikasi dan dikembangkan dengan merubah perangkat lunak sesuai dengan kebutuhan pengguna guna pengembangan lebih lanjut.
2. Perlu dilakukan uji coba lapangan untuk mengetahui bagaimana infrastruktur ini berinteraksi dengan pengguna secara langsung.
3. Perlu dilakukan penelitian lebih lanjut terkait jumlah pengguna yang dapat menggunakan *mail server embedded system*, sehingga dapat diketahui banyaknya jumlah pengguna secara pasti untuk skala pengguna besar.

## 7. DAFTAR PUSTAKA

- [CAS-12] Computing at school. 2012, “The Raspberry Pi Education Manual”.
- [EXS] Exile, Spam. “Citadel by SpamExile The E-mail System for Small to Medium Enterprise”, SpamExile Ltd.
- [ESY-13] Embedded System, 2010. Diakses melalui [Http://adiyasan.wordpress.com](http://adiyasan.wordpress.com). Tanggal akses: 23-09-2013.
- [HSE-03] Heath, S. 2003. “Embedded-Systems Design, Second Edition”, Newnes.
- [KAR-08] Kamal, Raj. 2008, “Embedded System”, 2<sup>nd</sup> edition, McGraw-Hill Education.
- [LMJ-11] Li, Y., Mingzhe Z., dan Jiaqi Gu. 2011, “An Email Server Optimized for Storage Issues”, Fudan University.

- [MIE-13] Mangunkusumo, I.E.S.W. 2013, ” Analisa dan Perancangan Keamanan Mail Server Zimbra pada Sistem Operasi Ubuntu 8.04”. Hal 1.
- [NAE-13] Narendro Arifia.”E-mail”, Universitas Gunadarma. Hal 2 & 13.
- [PNP-13] Padhye, N., Preet Jain. 2013. “Implementation of ARM Embedded Web Server for DAS Using Raspberry Pi”. Madhya Pradesh. India
- [RAA-11] Raupong., Anisa. 2011. “Bahan Ajar Mata Kuliah Perancangan Percobaan”. Universitas Hasanuddin. Makassar
- [RPF-13] Raspberry pi Foundation, 2013. Diakses melalui <http://www.raspberrypi.org/faqs>. Tanggal akses : 07-10-2013
- [RPI-13] Raspberry Pi. 2013. [http:// downloads.element14.com](http://downloads.element14.com). Tanggal akses: 16-10-2013
- [RPT-13] Raspberry Pi: The Perfect Home Server, 2013. Diakses melalui <http://www.linuxjournal.com/>. Tanggal akses: 16-10-2013
- [RRA-06] Rafiudin, rahmat. 2006, “Sistem Komunikasi Data Mutakhir”, CV Andi offset. Yogyakarta.
- [RFF-11] Rajab, Fahmi Febrian. 2011. “ Building a Mail Server Using Mail Citadel Transfer Agent on Linux Ubuntu 10:10 and accessing WITH Mozilla Thunderbird”. Universitas Gunadarma. Depok.
- [TUY-10] The University of York. 2010, “Email standards, protocol and guidance”.
- [TOM-05] Transfer rate Outgoing mail, 2005. Diakses melalui <http://www.mailarchive.com/mdaemonl@duaint.com/msg07528>. Tanggal akses: 21-10-2013.
- [WJY-04] Wan, Jun., Yiming Hu. 2004. “ A Performance Study on Internet Server Provider Mail Servers”.