

ANALISIS KINERJA IP PBX SERVER PADA SINGLE BOARD CIRCUIT RASPBERRY PI

Aidin Najihi¹⁾, I Wayan Mustika²⁾, Widyawan³⁾, Effan Najwaini⁴⁾

^{1,2,3}Jurusan Teknik Elektro dan Teknologi Informasi

Universitas Gajah Mada

⁴Jurusan Administrasi Bisnis, Program Studi Manajemen Informatika

Politeknik Negeri Banjarmasin

aidin.najihi@gmail.com¹⁾, wmustika@ugm.ac.id²⁾, widyawan@ugm.ac.id³⁾, effan.najwaini@gmail.com⁴⁾

Abstraksi

Raspberry PI merupakan sebuah komputer mini berukuran kartu kredit yang memiliki kemampuan komputasi yang sangat bagus. Komputer dengan ukuran yang sangat kecil memiliki keunggulan dalam fleksibilitas serta penggunaan energi. Raspberry dapat dikembangkan menjadi banyak kegunaan dalam dunia komputer. Konsumsi daya Raspberry sangat rendah yang sangat berguna untuk membuat aplikasi mobile yang hemat energi.

Pada penelitian ini akan dibuat sebuah server IP PBX menggunakan Asterisk yang diinstall pada single board circuit Raspberry PI. Pengujian dilakukan untuk mengetahui kinerja Raspberry PI sebagai server IP PBX yang meliputi jumlah panggilan yang mampu dilayani serta banyaknya panggilan bersamaan yang mampu dilayani dengan menggunakan codec yang berbeda. Sebagai perbandingan, akan dilakukan juga pengujian terhadap IPPBX server Asterisk menggunakan PC.

Berdasarkan hasil pengujian yang dilakukan, Raspberry PI sebagai IP PBX server mampu menangani 60 kedatangan panggilan perdetik tanpa adanya panggilan yang gagal diproses. Raspberry PI juga mampu menangani 110 panggilan bersamaan untuk codec GSM-GSM, 100 panggilan bersamaan untuk codec G711-G711 dan 70 panggilan bersamaan untuk transcoding GSM-G711.

Kata kunci – IP PBX, Raspberry, Asterisk, VoIP, QoS.

1. PENDAHULUAN

Latar Belakang

Telepon menjadi teknologi komunikasi yang sangat penting dan berkembang pesat. Dengan adanya kemajuan teknologi informasi terutama internet, teknologi telepon konvensional ikut mengalami perkembangan menjadi berbasis IP (VoIP). Infrastruktur jaringan internet yang sangat luas menjadikan teknologi VoIP menjadi sangat diminati.

Implementasi VOIP memiliki beberapa permasalahan yang penting yaitu masalah QoS (*Quality of Service*). Telepon analog menghasilkan kualitas suara yang jernih dan waktu *call setup* yang cukup singkat. Pada VoIP kualitas suara dipengaruhi oleh banyak faktor diantaranya yaitu: *bandwidth*, beban jaringan, *delay*, serta kinerja *server* sehingga perlu diperhitungkan jumlah pengguna, kualitas jaringan dan kualitas *server* untuk mendapatkan QoS yang baik (Miller 2005). Menurut Zasca (2009), pemilihan platform

PC atau *server* yang cocok (baik dari segi harga maupun kinerja) merupakan persoalan utama dalam membangun jaringan VoIP. Kinerja *server* yang jelek akan menurunkan kualitas atau bahkan tidak mampu untuk menghubungkan antar user (Zasca et al. 2009). Selain masalah *hardware*, kinerja *server* juga ditentukan oleh *software* yang digunakan. *Software* untuk membangun PBX yang *open source* dan paling efisien serta paling banyak digunakan hingga saat ini yaitu Asterisk (Zasca et al. 2009).

Raspberry PI merupakan sebuah komputer mini berukuran kartu kredit yang memiliki kemampuan komputasi yang sangat bagus. Komputer dengan ukuran yang sangat kecil memiliki keunggulan dalam fleksibilitas serta penggunaan energi. Raspberry dapat dikembangkan menjadi banyak kegunaan dalam dunia komputer. Konsumsi daya Raspberry sangat rendah yang sangat berguna untuk membuat aplikasi mobile yang hemat energi.

Pada umumnya, *software* Asterisk dipasang pada komputer *server*. Komputer ini kemudian dihubungkan ke jaringan IP dimana nantinya klien dapat mengakses baik itu menggunakan media kabel maupun tanpa kabel (*wireless*). Munculnya *single board* komputer seperti Raspberry PI yang lebih murah memungkinkan untuk di-*install* berbagai aplikasi termasuk Asterisk. Raspberry PI dapat difungsikan menjadi berbagai macam *server* termasuk IPPBX *server*, sehingga tidak lagi diperlukan sebuah komputer untuk membangun sebuah jaringan VoIP. Hal ini akan menghemat dari segi biaya pengadaan komputer dan juga biaya operasional. Tetapi, karena keterbatasan perangkat keras pada Raspberry PI dapat membuat kinerjanya menjadi lebih rendah dibandingkan penggunaan komputer untuk *server* IP PBX.

Sampai saat ini belum ada penelitian mengenai analisis quality of service (QoS) dari IPPBX *server* pada *single board circuit* Raspberry PI sehingga dirasakan perlu adanya penelitian untuk mengetahuinya. Dengan mengetahui QoS dari suatu *server* dapat diputuskan apakah pemanfaatan *single board circuit* Raspberry PI sebagai IPPBX *server* layak untuk diimplementasikan sesuai dengan kebutuhan pengguna. Pada penelitian ini akan dibuat sebuah *server* IPPBX menggunakan Asterisk yang di-*install* pada *single board circuit* Raspberry PI. Kemudian dari *server* tersebut akan dilakukan analisis QoS sehingga dapat diketahui kelayakan penggunaannya. Sebagai perbandingan, akan dilakukan juga pengujian terhadap IPPBX *server* Asterisk menggunakan PC.

Rumusan Masalah

Rumusan masalah dalam penelitian ini adalah:

1. Apakah Raspberry PI sebagai IP PBX *server* memiliki kinerja yang layak untuk diimplementasikan?
2. Berapa banyak panggilan yang mampu dilayani oleh IP PBX *server* pada Raspberry PI?

Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk mengetahui kinerja Raspberry PI sebagai *server* IP PBX yang meliputi jumlah panggilan yang mampu dilayani serta banyaknya panggilan bersamaan yang mampu dilayani

sehingga QoS pada *server* tersebut dapat ditentukan.

Manfaat Penelitian

Beberapa manfaat yang diharapkan dari penelitian ini yaitu :

1. Dapat menjadi referensi mengenai prosedur dalam melakukan pengujian kinerja terhadap suatu *server* IP PBX.
2. Sebagai bahan pertimbangan sebelum mengimplementasikan IP PBX *server* menggunakan Raspberry PI agar jumlah user yang menggunakan tidak melebihi kapasitas maksimum yang mampu dilayani oleh *server* tersebut.
3. Menghemat penggunaan sumber daya (*resource*) dan biaya operasional dalam membangun *server* IP PBX.
4. Bagi para pengembang embedded system akan mendapat gambaran mengenai kinerja yang dihasilkan jika membuat suatu perangkat *embedded* untuk *server* IPPBX.

2. LANDASAN TEORI

IP PBX

IP PBX merupakan perangkat *switching* komunikasi telepon dan data berbasis teknologi Internet Protocol (IP) yang mengendalikan ekstensi telepon analog (TDM) maupun ekstensi IP Phone atau softphone (Edvian, 2010). Dengan kata lain IP PBX merupakan PBX yang memiliki konektivitas IP. Penggunaan IP PBX akan menggabungkan komunikasi data dan komunikasi suara, sehingga hanya diperlukan infrastruktur jaringan IP untuk komunikasi suara maupun data. IP PBX mengkombinasikan fungsi dari PBX tradisional dan Voice over Internet Protocol (VoIP) sehingga melalui ekstensi VoIP mampu melakukan panggilan *external local*, *long distance* maupun *international call* melalui PSTN.

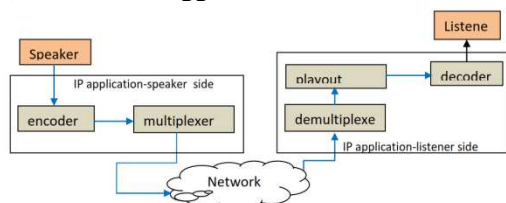
IP PBX lebih mudah di-*install* dibandingkan dengan PBX tradisional dan juga lebih mudah dimanajemen hanya dengan menggunakan interface web-browser yang menyediakan status panggilan, status sistem dan *history* panggilan. Penggunaan IP PBX akan menghemat biaya infrastruktur dan perawatan dibandingkan dengan PBX tradisional karena jaringan data dan suara yang tidak terpisah dan juga tidak memerlukan

biaya yang besar ketika akan menambah *user* atau ekstensi baru. IP PBX juga menawarkan fungsi yang sama seperti PBX tradisional seperti *call queue*, *call recording* maupun *voice mail* (TechKnowPartners, 2012).

VoIP (Voice over IP)

Voice over IP (VoIP) atau biasa juga disebut dengan internet telephony merupakan teknologi yang memungkinkan untuk melakukan percakapan atau komunikasi suara jarak jauh secara real-time dengan memanfaatkan jaringan IP (Schulzrinne dan Rosenberg, 1998). Pada VoIP, suara diubah ke dalam format digital berdasarkan *codec* tertentu dan kemudian dipecah menjadi paket-paket kecil dan dikirimkan melalui jaringan IP. Pada sisi penerima, paket yang diterima kemudian diterjemahkan lagi menjadi gelombang suara (Bacioccola dkk., 2007).

Penggunaan VoIP untuk komunikasi suara memiliki beberapa keuntungan dibandingkan dengan penggunaan PSTN (*Public Switch Telephone Network*) salah satunya yaitu biaya yang lebih murah. Penggunaan jaringan IP yang berbasis *packet switch* membuat biaya untuk percakapan dapat lebih murah dibandingkan dengan menggunakan PSTN yang berbasis *circuit switch* (Purbo, 2007). Pada jaringan *packet*, informasi (suara) dikirimkan dalam bentuk paket, sehingga satu kanal dapat dipakai bersama-sama, sehingga biaya percakapan menjadi lebih murah. Selain itu, penggunaan *codec* tertentu dan juga adanya fitur *silence suppression* dan *voice activity detection* (VAD) pada VoIP dapat menghemat *bandwidth* yang digunakan, sehingga biaya yang dikeluarkan juga akan semakin murah (Anton dan Anggraini, 2008).



Gambar 1 Mekanisme VoIP (Bacioccola dkk., 2007)

Perhitungan Bandwidth VoIP

Untuk menghitung *bandwidth* VoIP dapat menggunakan Persamaan 1, 2 dan 3 (Cisco, 2006; Bintoro, 2011).

Total Packet Size

$$= (L2 Header) + (IP, UDP, RTP Header) + (Voice Payload Size)$$

Persamaan 1

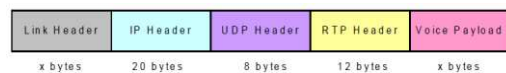
$$Packet\ per\ Second = \frac{codec\ bit\ rate}{(voice\ payload\ size)}$$

Persamaan 2

$$Bandwidth = Total\ Packet\ Size \times Packet\ per\ Second$$

Persamaan 3

Format paket VoIP ditunjukkan pada Gambar 2. Setiap paket VoIP terdiri atas *payload* dan *header*, dimana *payload* berisi suara yang telah disampling berdasarkan *codec* tertentu dan *header* berisi informasi yang dibutuhkan oleh jaringan untuk mengirimkan paket suara tersebut. Pada VoIP terdapat RTP (*Real Time Protocol*) yang berjumlah 12 bytes. Protokol ini berisi informasi tentang *codec* suara yang digunakan, *timestamps* (waktu pengiriman) serta *sequence number* (untuk mengetahui *packet loss* dan pengurutan kembali). *Header* RTP ini dibungkus dengan *transport* UDP yang bersifat *unreliable*. Pada VoIP yang merupakan komunikasi *real-time* lebih mementingkan ketepatan waktu sampainya paket dibandingkan dengan *packet loss*. Komunikasi VoIP dapat mentoleransi beberapa *packet loss* sehingga digunakan protokol UDP sebagai protokol *transport*. Selanjutnya UDP ini dibungkus dengan *header* IP yang berjumlah 20 bytes, yang berguna untuk pengalamatan di jaringan IP. Protokol layer 2 dapat berupa Ethernet jika berada di jaringan LAN. *Header* Ethernet berjumlah 18 bytes dimana sudah termasuk 4 bytes *frame check sequence (FCS)* atau *cyclic redundancy check (CRC)* (Cisco, 2006).



Gambar 2 Format Paket VoIP

Contoh perhitungan *bandwidth* VoIP jika menggunakan *codec* G.711 dengan layer 2 ethernet sebagai berikut:

$$Total\ Packet\ Size = 18 + 20 + 8 + 12 + 160 = 218\ bytes = 1744\ bit$$

$$Packet\ per\ Second = 64\ kbps / 160\ byte = 50\ packet\ per\ second$$

$$Bandwidth = 1744\ bit \times 50\ packet\ per\ second$$

= 87200 bps = 87.2 kbps

3. METODOLOGI

Alat

Alat yang digunakan adalah beberapa perangkat atau piranti sistem komputer dan jaringan yang digunakan berupa:

1. Raspberry PI 2

Raspberry PI 2 adalah komputer papan tunggal (*single board circuit*) yang memiliki ukuran sebesar kartu kredit. Raspberry Pi dapat digunakan untuk berbagai keperluan seperti *game*, *media player* maupun alat kontrol. Raspberry Pi dalam percobaan ini digunakan sebagai IP PBX *server*. Spesifikasi Raspberry Pi dapat dilihat pada Tabel 1.

Tabel 1 Spesifikasi Raspberry PI

CPU	ARM Cortex A7 SOC Broadcom BCM 2836
CPU Speed	900 MHz Quadcore
GPU	Dual Core Video Core IV Multimedia Co-Processor
RAM Size	1 GB LPDDR2 SDRAM
Ethernet Port	1 Port 10/100 Base T Ethernet

2. Notebook

Pada penelitian ini digunakan dua buah notebook. Notebook digunakan sebagai *call generator*. Satu buah sebagai *User Agent Client* (UAC) dan satu buah sebagai *User Agent Server* (UAS). Tabel 2 menunjukkan spesifikasi notebook yang digunakan dalam penelitian ini.

Tabel 2 Spesifikasi Notebook

Merk / Tipe	Acer Aspire 5572NWX Ci	Toshiba Satellite C640
CPU	Intel Core Duo T2300	Intel Pentium Dual Core P6100
CPU Speed	@ 1,66 GHz (2 Cores)	@ 2GHz (2 Core)
Ram Size	1024 MB DDR2	1024 MB DDR3

Hardisk	80 GB Sata 5400 RPM	320 GB Sata 5400 RPM
Operating System	Ubuntu Server 12.04	Ubuntu Server 12.04
Penggunaan	Server UAS untuk <i>CallGenerator</i> SIPp	Server UAC untuk <i>CallGenerator</i> SIPp

3. Personal Computer

Personal Computer digunakan untuk *capture* paket RTP saat terjadi hubungan antara 2 buah *client*. Komputer menggunakan 2 buah ethernet dimana ethernet 1 sebagai client 1 dan ethernet 2 sebagai client 2. Komputer menggunakan sistem operasi windows 7 dan menggunakan dua buah *software* phone sebagai client. Spesifikasi komputer yang digunakan dapat dilihat pada Tabel 3.

Tabel 3 Spesifikasi PC

CPU	Intel Pentium Dual CPU E 5400
CPU Speed	@ 2,7 GHz
Ram Size	2048 Mb DDR 3
Operating System	Windows 7
VGA	Intel GMA X4500 Graphics
Ethernet Port	1 Port 10/100 Base T Ethernet

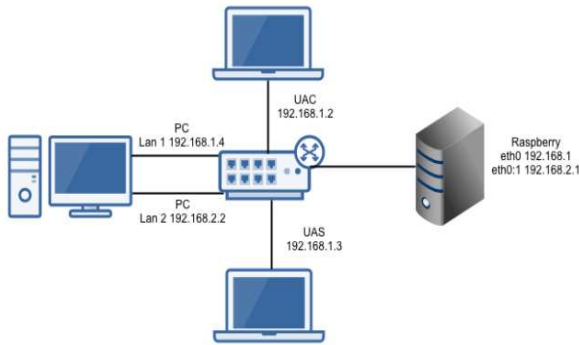
Jalannya Penelitian

Jalannya penelitian dan pengukuran dapat dilihat dalam diagram alir pada Gambar 5.

Studi Pustaka

Langkah pertama penelitian adalah studi pustaka. Studi pustaka meliputi tinjauan literatur, pemilihan kasus, teknik pengumpulan data serta skenario penelitian. Studi pustaka juga bertujuan membangun rumusan masalah yang lebih fokus. Literatur yang digunakan di studi pustaka meliputi *paper*, jurnal, artikel, tesis tentang Raspberry, VoIP dan QoS.

Perancangan Sistem



Gambar 3 Konfigurasi Jaringan

Penelitian ini menggunakan PC sebagai client yang melakukan panggilan antara LAN 1 ke LAN 2. Panggilan dilakukan dalam satu PC untuk memudahkan pengukuran *delay*. Saat terjadi panggilan, maka dalam paket RTP yang dikirimkan terdapat informasi waktu pengiriman. Waktu pengiriman dapat dilihat saat paket yang dikirimkan di-capture dengan menggunakan wireshark. Apabila perhitungan *delay* dengan cara menggunakan dua buah perangkat akan sulit dilakukan perhitungan. Waktu antara perangkat satu dan lainnya tidak sama. Penyamaan waktu dapat dilakukan tetapi karena *delay* yang diukur sangat kecil maka perbedaan waktu yang sangat kecil antar perangkat sangat berpengaruh.



Gambar 4 Diagram Alir Jalannya Penelitian

Penelitian ini menggunakan dua buah *ethernet* dalam satu PC. Masing masing

ethernet akan melakukan panggilan ke *ethernet* yang lain. Agar masing-masing *ethernet* dapat berfungsi dengan benar maka Raspberry juga harus menggunakan dua buah *ethernet*. Untuk mengatasi hal tersebut, linux mempunyai fasilitas untuk membuat IP address Alias.

IP Address Alias adalah suatu kondisi, dimana kita diharuskan menggunakan dua atau lebih Ip Address dalam satu NIC. Seolah-olah computer kita memiliki dua buah NIC, dan terkoneksi dalam dua atau lebih jaringan yang berbeda. *Server* Raspberry perlu dibuat satu buah IP *address* alias untuk melayani *ethernet* kedua di PC.

Pengujian Jumlah Maksimum Panggilan Perdetik

Pengujian jumlah maksimum panggilan perdetik dilakukan dengan menggunakan *call* generator SIPp. Pada pengujian ini satu laptop akan menjalankan *call generator* sebagai UAC (*User Agent Client*) dan laptop lainnya menjalankan *call generator* sebagai UAS (*User Agent Server*). Pemanggilan dilakukan oleh UAC ke UAS dengan *rate* panggilan tertentu yang ditingkatkan sampai *server* tidak mampu lagi melayani panggilan yang datang.

Pengujian dilakukan dengan terlebih dahulu menjalankan SIPp sebagai UAS atau penerima panggilan pada terminal linux SIPp diatur untuk menerima panggilan pada port 6001 sesuai dengan konfigurasi akun pada Asterisk.

Setelah SIPp sebagai UAS telah dijalankan dan siap menerima panggilan. UAS akan terus berjalan sampai dimatikan secara manual. SIPp sebagai UAC dapat segera dijalankan pada *notebook* yang lain.

Pada saat panggilan berlangsung dilakukan pengamatan terhadap penggunaan prosesor pada *server* Asterisk dengan menggunakan perintah “sar”. Data penggunaan prosesor diambil setiap detik selama 10 detik setelah panggilan berjalan.

Pengujian Jumlah Maksimum Panggilan Bersamaan

Pengujian jumlah maksimum panggilan bersamaan dilakukan dengan membangkitkan panggilan menggunakan softphone X-lite dan Eyebeam untuk satu panggilan dan menggunakan SIPp sebagai *call generator* untuk lebih dari satu panggilan. Jumlah panggilan akan dinaikan sampai mencapai

jumlah panggilan bersamaan tertentu dan selanjutnya dilakukan pengamatan terhadap penggunaan prosesor.

Pengujian ini dilakukan terhadap tiga skenario penanganan paket suara oleh *server* melalui *codec* yang digunakan yaitu *codec* GSM ke GSM, *codec* G711 ulaw ke G711 ulaw dan *transcoding* (*codec* GSM ke G711 ulaw).

Pada saat panggilan berlangsung dilakukan pembacaan penggunaan *processor* dan *capture* paket data dari PC menggunakan *wireshark*. *Capture* paket data digunakan untuk pengukuran QOS.

4. HASIL DAN PEMBAHASAN

Maksimum Panggilan Perdetik

Maksimum kedatangan panggilan perdetik diukur dengan membangkitkan panggilan tanpa pengiriman paket RTP. Panggilan dibangkitkan mulai 1 panggilan dan terus ditambah per sepuluh panggilan. Saat pemakaian *prosesor* mencapai 100% dan terjadi kegagalan penanganan panggilan, maka penambahan panggilan diturunkan menjadi per satu panggilan.

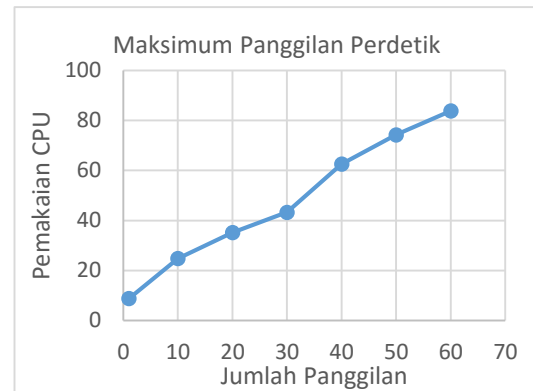
Berdasarkan hasil pengujian, perangkat Raspberry PI mampu melayani 60 kedatangan panggilan perdetik tanpa adanya panggilan yang gagal diproses. Jika jumlah panggilan dinaikkan menjadi 62 panggilan perdetik SIPp menunjukkan adanya retransmisi yang berarti ada sinyal INVITE yang gagal diproses oleh *server*. *Server* tidak mampu melayani semua panggilan yang datang dengan *rate* lebih besar dari 60 kedatangan panggilan perdetik hal ini dapat terlihat dari banyaknya panggilan gagal yang terjadi jika *rate* kedatangan panggilan sebesar 62 panggilan perdetik.

Apabila sudah terjadi retransmisi maka panggilan-panggilan yang datang setelah terjadi proses retransmisi tidak akan dilayani.

Pemakaian *prosesor* pada saat panggilan perdetik dinaikkan ditunjukkan oleh grafik pada Gambar 7. Pemakaian CPU meningkat secara konstan sesuai dengan kenaikan panggilan. Pada saat panggilan 50 panggilan per detik, pemakaian CPU hanya berada di sekitar 70 %, tetapi jika *rate* kedatangan panggilan dinaikkan menjadi 60 panggilan perdetik membuat penggunaan prosesor mencapai 100% sehingga terjadi

banyak panggilan masuk yang tidak mampu dilayani oleh *server*.

Pada pengujian dengan menggunakan PC, *server* mampu menangani panggilan yang masuk hingga 220 panggilan perdetik. Kedatangan panggilan diatas 220 perdetik membuat penggunaan prosesor mencapai 100% dan terjadi banyak kegagalan panggilan. Gambar 8 menunjukkan hasil pengujian kedatangan panggilan tiap detik pada PC.



Gambar 5 Maksimum Kedatangan Panggilan Perdetik Raspberry

Penggunaan Raspberry untuk *server* VoIP tidak dapat disetarakan dengan PC meskipun *prosesor* yang digunakan Raspberry sudah mendekati *prosesor* PC. Hasil pengujian panggilan yang masuk perdetik pada PC hampir empat kali dari kemampuan Raspberry. Meskipun Raspberry menghasilkan hasil yang rendah tetapi Raspberry mempunyai kelebihan dalam bentuk yang ringkas dan power yang rendah.



Gambar 6 Maksimum Kedatangan Panggilan Perdetik PC

Maksimum Panggilan Bersamaan

Pengujian jumlah pembicaraan atau panggilan bersamaan dilakukan terhadap tiga

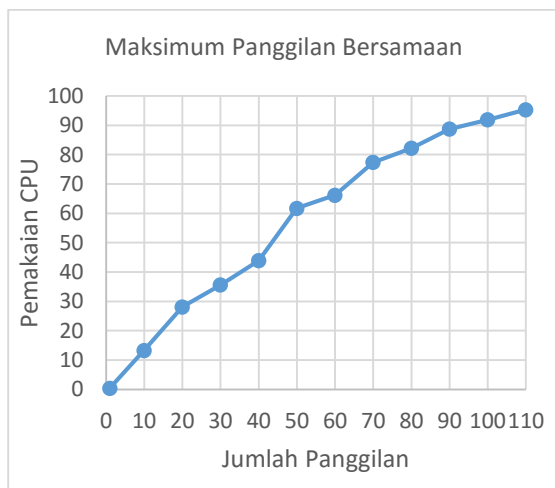
metode pemakaian *codec* yaitu *codec* GSM-GSM, *codec* G711-G711 dan GSM-G711.

Penggunaan Codec GSM-GSM

Hasil pengujian pada penggunaan *codec* GSM tanpa adanya *transcoding* menunjukkan bahwa Raspberry mampu melayani maksimal 110 panggilan. Pemakaian CPU naik secara konstan seiring naiknya jumlah panggilan dalam satu waktu. Hasil pengujian menunjukkan bahwa untuk 110 panggilan masih dapat dilayani oleh *server*. Seluruh panggilan dapat dilayani dengan baik tanpa adanya kegagalan panggilan. Gambar 9 menunjukkan hasil pengukuran dengan menggunakan *codec* GSM-GSM

Penggunaan Codec G711-G711

Hasil pengujian pada penggunaan *codec* G711 tanpa adanya *transcoding* menunjukkan bahwa Raspberry mampu melayani maksimal 100 panggilan. Pemakaian CPU naik secara konstan seiring naiknya jumlah panggilan dalam satu waktu. Hasil pengujian menunjukkan bahwa untuk 100 panggilan masih dapat dilayani oleh *server*. Seluruh panggilan dapat dilayani dengan baik tanpa adanya kegagalan panggilan. Gambar 10 menunjukkan hasil pengukuran dengan menggunakan *codec* G711-G711.



Gambar 7 Maksimum Panggilan Bersamaan GSM-GSM



Gambar 8 Maksimum Panggilan Bersamaan G711-G711

Hasil pengukuran penggunaan *codec* GSM dapat meningkatkan jumlah pembicaraan atau panggilan bersamaan yang dapat dilayani dibandingkan penggunaan *codec* G711 sesuai dengan penelitian Montoro dan Casilari (2009). Hal ini dikarenakan jumlah *payload* perpaket pada *codec* GSM hanya sebesar 33 Byte, sedangkan pada *codec* G711 sebesar 160 Byte. Jumlah *payload* berpengaruh terhadap pemakaian CPU. Jumlah *payload* yang besar juga menghasilkan suara yang lebih baik tetapi membebani jaringan dan *server*.

Dengan *payload* sebesar 160 Byte maka perlu diukur *bandwidth* total yang digunakan. *Bandwidth* total yang digunakan perlu diukur agar tidak lebih besar dari yang mampu lewat di jaringan. Berdasarkan perhitungan *bandwidth* pada persamaan 3 maka untuk 110 pembicaraan VOIP pada *codec* GSM didapat *bandwidth* yang digunakan sebesar 17,88 Mbps. Untuk 100 pembicaraan VOIP pada *codec* G711 didapatkan 17,88 Mbps *bandwidth* yang digunakan.

Pengukuran *throughput* jaringan dilakukan dengan menggunakan *software* tamosoft *throughput test software* di-install di dua komputer yang tersambung di jaringan yang digunakan. *Software* akan mengukur berapa *throughput* yang dapat dilayani jaringan. *Throughput* jaringan harus lebih besar dari *bandwidth* VOIP yang digunakan agar tidak terjadi *bottleneck* yang mempengaruhi jalannya pengujian.

Penggunaan Codec GSM-G711

Hasil pengujian pada penggunaan *codec* GSM-G711 (terjadi *transcoding*) menunjukkan bahwa Raspberry hanya mampu melayani maksimal 70 panggilan. Pemakaian CPU naik secara konstan seiring naiknya jumlah panggilan dalam satu waktu. Hasil pengujian menunjukkan bahwa untuk 70 panggilan masih dapat dilayani oleh *server*. Seluruh panggilan dapat dilayani dengan baik tanpa adanya kegagalan panggilan. Gambar 11 menunjukkan hasil pengujian pada *codec* GSM-G711 (*transcoding*).

Pada pengujian panggilan bersamaan di *PC* dengan menggunakan *Codec* GSM-G711 *server* mampu melayani panggilan hingga 250 panggilan secara bersamaan. Hasil pengujian menunjukkan penggunaan *PC* sebagai *server* VoIP lebih baik hampir empat kali lipat dalam menangani panggilan bersamaan untuk *transcoding*.



Gambar 9 Maksimum Panggilan Bersamaan GSM-G711

5. KESIMPULAN DAN SARAN

Kesimpulan

Raspberry PI dapat diimplementasikan sebagai IPPBX *server* dan mampu melayani komunikasi VOIP dengan baik pada jumlah panggilan tertentu.

Raspberry PI dapat melayani 60 kedatangan panggilan perdetik. Jumlah ini jauh lebih kecil dibandingkan pada *PC* yang mampu menangani 220 panggilan perdetik.

Raspberry PI mampu menangani 110 panggilan bersamaan saat kedua *client* menggunakan *codec* GSM. Saat menggunakan

codec G711 Raspberry hanya mampu melayani 100 panggilan bersamaan. Jumlah ini jauh lebih kecil dibandingkan *PC* yang mampu menangani 500 panggilan bersamaan.

Penggunaan *codec* yang berbeda antara *client* (*transcoding*) menurunkan jumlah panggilan bersamaan hanya sejumlah 70 panggilan bersamaan. Pengujian pada *PC* saat menggunakan *codec* yang berbeda mampu menangani hingga 250 panggilan bersamaan.

Penggunaan *codec* GSM menghasilkan jumlah panggilan yang lebih besar dibandingkan dengan *codec* G711. Hal ini dikarenakan jumlah *payload* yang berbeda sehingga membedakan besar paket RTP yang dikirimkan. Semakin besar paket, semakin memerlukan sumber daya CPU yang besar di *server*.

Saran

Perlu dilakukan penelitian lebih lanjut mengenai penggunaan beberapa Raspberry PI sebagai IPPBX *server* dalam sistem terdistribusi.

Perlu dilakukan penelitian lebih lanjut menggunakan penggunaan perangkat *single board* lain sebagai IPPBX *server*.

6. DAFTAR PUSTAKA

- Ardilestian, B. D., 2010, Implementasi Server VoIP pada Access Point Linksys WRT54GL Berbasis Sistem Operasi GNU/Linux Distro OpenWRT, *Tesis*, Program Studi S2 Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Gadjah Mada, Yogyakarta.
- Bacioccola, A., Cicconetti, C. dan Stea, G., 2007, User-level Performance Evaluation of VoIP Using NS-2, *ValueTools '07 Proceeding of the 2nd International Conference on Performance Evaluation Methodologies and Tools*, Brussels, Belgium.
- Bintoro, A., 2011, Analisis QoS Voice over Internet Protocol (Studi Kasus di Pemkab Sleman), *Tesis*, Program Studi Magister Teknologi Informasi, Jurusan Teknik Elektro, Fakultas Teknik, Sekolah Pasca Sarjana, Universitas Gadjah Mada, Yogyakarta.
- Cisco, 2006, Voice over IP – Per Call Bandwidth Consumption, *cisco.com*, diakses 12 April 2015.

- ITU, 1996, Method for Subjective Determination of Transmission Quality, *ITU-T Recommendation P.800*.
- ITU, 2000, Transmission planning for private/public network interconnection of voice traffic, *ITU-T Recommendation G.175*.
- ITU, 2003, One-way transmission time, *ITU-T Recommendation G.114*.
- ITU, 2008, Methods for Subjective Determination of Transmission Quality, *ITU-T Recommendation E.800*.
- ITU, 2014, The E-Model: A Computational model for Use in Transmission Planning, *ITU-T Recommendation G.107*.
- Ma, A., 2001, *Voice over IP (VoIP)*, Spirent Communications, Inc., Crawley.
- Najwaini, E., 2014, Analisis Kinerja VoIP Server pada Wireless Access Point, Program Studi S2 Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Gadjah Mada, Yogyakarta.
- Nugroho, S., 2011, Analisis Perbandingan VoIP Server Antara Brikker IPPBX dengan Asterisk, *Tesis*, Program Studi S2 Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Gadjah Mada, Yogyakarta.
- Purbo, O. W., 2007, *VoIP : Cikal Bakal "Telkom Rakyat" (Panduan Lengkap Setting VoIP)*, PT Prima Infosarana Media, Jakarta.
- Rughinis, R. & Iconaru, C., 2008. A Practical Analysis of Asterisk SIP Server Performance. In *The 7th RoEduNet International Conference*. Cluj-Napoca, pp. 61–64.
- Surimi, L., 2014, Analisis Kualitas VoIP pada SCTP yang menggunakan Explicit Congestion Notification (ECN) dan Active Queue Management (AQM), *Tesis*, Program Studi S2 Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Gadjah Mada, Yogyakarta.
- TechKnowPartners, 2012, An Introduction to Network Telephony Why IP-PBX System are Good for Business, <http://www.techknowpartners.com/collateral/whyippbx.pdf>, diakses 12 April 2015.
- Zasępa, M. et al., 2009. Implementation of Cost-effective VoIP Network. In *16th International Conference "Mixed Design of Integrated Circuits and Systems"*. Lodz: Department of Microelectronics & Computer Science, Technical University of Lodz, pp. 4–7.