

IMPLEMENTASI MINI SERVER BERBASIS SECURITY PROXY DENGAN MENGGUNAKAN RASPBERRY PI SECARA PORTABLE

Mohammad Rizki

Duddy Soegiarto

Anang Sularsa

Teknik Komputer Universitas Telkom

mohammad.rizki5@gmail.com

Duddy@tass.telkomuniversity.ac.id

ananks@outlook.com

Abstrak

Perkembangan teknologi informasi yang semakin berkembang mempermudah masyarakat untuk mengakses *internet*. *Internet* menjadi hal yang dibutuhkan masyarakat, sehingga banyak transportasi memberikan fasilitas *internet* gratis untuk menarik perhatian masyarakat. Penambahan fasilitas *internet* pada transportasi menghabiskan biaya yang cukup besar untuk pemakaiannya, maka dibutuhkan sebuah server yang menjaga keamanan jaringan dan mengatur pemakaian *internet*. Bagaimanapun hal ini menjadikan kendala bila harus menambahkan server pada transportasi, dan juga biaya yang harus dikeluarkan untuk mendapatkan sebuah server pun tidak sedikit.

Untuk memenuhi kebutuhan tersebut, maka pada Proyek Akhir ini dibangun mini server yang berbasis *security proxy* dengan menggunakan Raspberry Pi secara *portable*. Raspberry Pi memiliki dimensi yang lebih kecil dari laptop maka hal ini menjadikan Raspberry Pi sebagai server yang *portable* dan harganya lebih murah. Raspberry Pi difungsikan sebagai server untuk menjaga keamanan informasi data pada jaringan dan mengatur *management bandwidth*, *cache*, *filtering web* untuk pengaksesan *internet* oleh pengguna, dengan cara ini biaya yang dikeluarkan dapat di perkecil dan pengguna layanan *internet* dapat terorganisir.

Kata kunci: *Proxy Server, system monitor, Modem (GSM), Raspberry Pi*

Abstract

The development of information technology is growing easier for people to access the internet. Internet becomes a community needs, so a lot of transportation provide free internet facility to attract public attention. Adding internet facilities in transportation spending considerable cost to use, it takes a server that maintain network security and regulate the use of the internet. However this makes the constraint having to add servers to the transport, and also the costs to obtain a server was not small.

To meet these needs, then the final project is built mini proxy server-based security using the Raspberry Pi is portable. Raspberry Pi has smaller dimensions than a laptop then it makes Raspberry Pi as a server that is portable and cheaper. Raspberry Pi function as a server for security information and organize data on the network bandwidth management, caching, web filtering for Internet access by the user, with the way the costs incurred by the company can be minimized and users of Internet services can be organized.

Keywords: *M Proxy Server, system monitor, Modem (GSM), Raspberry Pi*

1. Pendahuluan

1.1 Latar Belakang

Penggunaan teknologi jaringan komputer telah banyak berkembang dengan cepat seiring perkembangan teknologi informasi. Teknologi jaringan komputer juga menyediakan internet sebagai penghubung antar pengguna komputer untuk saling berhubungan, khususnya jarak yang jauh, bertukar informasi dan lainnya. Penggunaan internet sudah tidak asing di masyarakat umum sekarang untuk melakukan transaksi ataupun mencari informasi. Banyak jasa transportasi yang menggunakan modem GSM untuk memberi layanan internet kepada para penumpang. Pada modem GSM cukup sulit

untuk menambahkan fitur-fitur keamanan dan layanan untuk *client*, karena modem GSM telah dikonfigurasi oleh *provider* secara *default*.

Server adalah sebuah sistem komputer yang menyediakan jenis layanan tertentu dalam sebuah jaringan komputer dan salah satu pendukung untuk membuat sebuah jaringan. Server dengan ukuran yang mini akan mempermudah server untuk dipasang pada transportasi umum. Agar kinerja server baik diperlukan keamanan pada *proxy*, pengaturan *bandwidth*, dan *cache* pada jaringan.

Untuk memenuhi kondisi tersebut, maka pada Proyek Akhir ini dibangun server *portable* menggunakan Raspberry Pi . Raspberry Pi adalah mini komputer mainboard yang bersifat SoC(*System on chip*) yang menggabungkan komponen dan fungsi-fungsi komputer serta elektronika ke dalam satu chip atau bisa juga disebut dengan *embedded system*. Biaya

yang diperlukan juga tidak terlalu besar. Sistem operasi yang digunakan pada Raspberry Pi adalah *Linux Debian Squeeze* sebagai sistem operasi yang direkomendasikan. Sedangkan *proxy* adalah sebuah program komputer yang berperan sebagai penghubung antara suatu komputer dengan jaringan internet maupun intranet, dan juga berfungsi untuk menyaring/membatasi permintaan, meningkatkan kinerja koneksi.

Pada proyek akhir ini akan dibuat server yang portable pada transportasi yang memberikan layanan *internet* kepada *client*. Informasi dan data pada server dapat di monitor oleh *admin* dengan cara meremote server di dalam satu jaringan.

1.2 Rumusan Masalah

Adapun rumusan masalah dalam proyek akhir ini adalah sebagai berikut :

1. Bagaimana membangun infrastruktur dan sistem keamanan jaringan pada transportasi ?
2. Bagaimana melakukan *Management Bandwith, cache, filtering web* pada aplikasi *proxy* di perangkat portable?

1.3 Tujuan

Adapun tujuan dari proyek akhir ini adalah sebagai berikut :

1. Membangun sistem keamanan *proxy* pada jaringan transportasi.
2. Menggunakan Squid untuk melakukan *Management Bandwith, cache, filtering web* pada aplikasi keamanan jaringan yang ditanamkan pada Raspberry Pi.

1.4 Batasan Masalah

Adapun batasan masalah pada proyek akhir ini adalah :

1. Membangun server *portable* sebagai pusat layanan.
2. *Client login* menggunakan *username* dan *password* yang diberikan *admin*.
3. Menggunakan Sistem Operasi Raspbian Wheezy yang digunakan pada Raspberry Pi.
4. Sistem keamanan menggunakan *captive portal* dengan *protocol HTTPS*.

5. *Admin* mengakses *server* menggunakan *remote ssh* dalam jaringan local.
6. Menggunakan satu modem GSM.
7. Pengujian pada kendaraan pribadi roda empat.
8. Membahas kewanalaan yang digunakan dan pembagian *bandwith, cache, dan filtering web*.

1.5 Definisi Operasional

1. Server Proxy adalah mediator antara client dan server di luar jaringan untuk menyaring/membatasi permintaan, meningkatkan kinerja koneksi dan *sharing*.
2. Sistem monitor adalah kegiatan memantau suatu kinerja, yang dijalankan pengguna untuk mengumpulkan informasi dari sistem yang dibuat. Dalam sistem monitor dapat diketahui data seperti pemakaian *bandwidth, user* dan lainnya.
3. Modem (GSM) adalah suatu alat penghubung antara jaringan lokal dengan internet, dengan menggunakan sistem telepon selular yang dilengkapi dengan pemegang kartu SIM yang terintegrasi.
4. Raspberry Pi adalah mini komputer mainboard yang bersifat SoC (*System on chip*) yang menggabungkan komponen dan fungsi-fungsi komputer serta elektronika ke dalam satu chip atau bisa juga disebut dengan *embedded systems*.

1.6 Metode Pengerjaan

Tahapan pengerjaan Proyek Akhir ini adalah :

1. Pengumpulan data
Tahap pertama adalah pengumpulan data. Pada tahap ini didefinisikan kebutuhan – kebutuhan proyek akhir yang akan dikerjakan, penyusunan proposal, dan gambaran sistem yang akan dibuat.
2. Perancangan
Tahap kedua adalah perancangan. Pada tahap ini dilakukan

perancangan sistem yang akan dibuat seperti desain sistem server, perangkat keras dan perangkat lunak yang diperlukan.

3. Implementasi
Mengimplementasikan perancangan sistem dari model yang telah dibuat. Simulasi pengujian sistem monitor server secara *remote* agar berjalan dengan baik.
4. Pengujian
Tahap keempat adalah pengujian. Tahap pengujian akan dilakukan

2.1 Proxy Server

Proxy server adalah server yang diletakkan antara suatu aplikasi *client* dan aplikasi server yang dihubungi. Aplikasi *client* dapat berupa browser web, *client FTP*, dan sebagainya. Sedangkan aplikasi server dapat berupa server *web*, server *FTP*, dan sebagainya. Berdasarkan kegunaan dari *proxy server*, berikut fungsinya :

1. *Connection sharing*
Bertindak sebagai *gateway* yang menjadi batas antara jaringan lokal dan jaringan luar.
2. *Filtering*
Bekerja pada layar aplikasi sehingga berfungsi sebagai *firewall* paket *filtering* yang digunakan untuk melindungi jaringan lokal terhadap gangguan atau serangan dari jaringan luar.
3. *Caching*
Proxy server memiliki mekanisme penyimpanan obyek-obyek yang sudah diminta dari server di *internet*. Mekanisme *caching* akan menyimpan obyek-obyek yang merupakan permintaan dari para pengguna yang di dapat dari *internet*.

2.2 Captive Portal

Captive Portal sebenarnya merupakan mesin *router* atau *gateway* yang memproteksi atau tidak mengizinkan adanya *trafik* sehingga *user* melakukan *registrasi/authentication*. Secara umum *Captive Portal* memiliki fungsi untuk mencegah atau memblokir koneksi yang tidak diinginkan dan mengarahkan *client* ke protokol tertentu. *Captive Portal* sebenarnya sama dengan *router* atau *gateway* yang memiliki fungsi untuk menyaring semua koneksi yang masuk dan menolak koneksi yang tidak diinginkan (*client* tidak terdaftar). Biasanya *Captive Portal*

untuk memeriksa kesesuaian hasil implementasi yang dibuat dengan model solusi dan kebutuhan.

5. Dokumentasi
Pada tahap terakhir ini adalah dokumentasi. Tahap dokumentasi dilakukan saat pengujian hasil implementasi melalui foto dan *screenshot*

2. Tinjauan Pustaka

digunakan pada infrastruktur *wireless* seperti hotspot area, tapi tidak menutup kemungkinan diterapkan pada jaringan kabel.

2.3 HTTPS

HTTPS adalah penggabungan antara *Hypertext Transfer Protocol (HTTP)* dengan *SSL / TLS* protokol. Semua komunikasi yang dilakukan melalui *HTTPS* akan dienkripsi dengan tujuan untuk keamanan saat terjadi transaksi data di *internet*. Biasanya para *hacker* atau peretas *internet* yang biasa menggunakan *tools wireshark* sangat mudah untuk mencuri data dari klien yang terhubung ke *internet* dengan menggunakan *HTTP*. *HTTPS* keamanannya lebih terjamin dibandingkan *HTTP*.

2.4 MySQL

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (*RDBMS*) yang didistribusikan secara gratis. Setiap pengguna dapat secara bebas menggunakan *MySQL*, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. *MySQL* sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; *SQL (Structured Query Language)*. *SQL* adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Kehandalan suatu sistem basisdata (*DBMS*) dapat diketahui dari cara kerja pengoptimasianya dalam melakukan proses perintah-perintah *SQL* yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. *MySQL* mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional. Pada modus operasi non-transaksional, *MySQL* dapat dikatakan unggul dalam hal unjuk kerja di bandingkan perangkat lunak basisdata kompetitor lainnya. Namun demikian pada modus non-transaksional tidak ada jaminan atas

reliabilitas terhadap data yang tersimpan, karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (*wordpress*), *CMS*, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basisdata transaksional, hanya saja sebagai konsekuensinya unjuk kerja *MySQL* pada modus transaksional tidak secepat unjuk kerja pada modus non-transaksional.

2.5 FreeRadius

Freeradius merupakan radius server yang cukup populer di ranah open source. Dikembangkan oleh Alan Dekok dan Miquel Van Smoorenburg pada tahun 1999. Sebelumnya Miquel mengembangkan *cistron radius*, namun kemudian menjadi tidak aktif dikembangkan lagi. Freeradius kemudian dapat diterima secara luas dan mendapat dukungan dari komunitas opensource. Seiring berkembangnya waktu, *freeradius* selain *support text*, *file* kemudian *support ldap*, *sql(mysql, oracle, postgresql, msql, etc)* dan *eap*. Freeradius sendiri diklaim cukup cepat, kaya fitur, dapat diandalkan dan *skalabel*. Freeradius sendiri berjalan di banyak *platform operating system* seperti *windows*, keluarga *linux/unix*, *mac os*, *sun solaris*.

Fitur-fitur dari free radius secara garis besar :

- Memiliki performance yang tinggi, mendukung *ha*, *fail-over*.

3. Kebutuhan dan analisis

3.1 Analisis Sistem Saat Ini (atau Produk)

3.1.1 Proses Penginputan Data

- 1) *Client* mengisi data di loket transportasi yang berisi dengan nama, nomer bangku penumpang, banyak penumpang, no telepon dan lain-lain
- 2) Petugas loket (*admin*) mengambil data dari *client* yang akan menjadi rujukan sebagai *username* dan *password* untuk data *login*. *Username* terdiri dari data nama dan nomer bangku *client*, *password* terdiri dari nomer seri tiket yang telah tercantum pada tiket dan nama *client*.

- Kaya fitur, dukung banyak *operating system*, *eap*, *database*, *fungsiionalitas AAA*, *virtual server*, *proxy*, *etc*.
- Modular, dukungan *addon/plugin* tambahan, dapat diimplementasikan pada *embedded system*.

Skalabilitas, untuk beban yang tinggi *freeradius support* berdasarkan maksimum *request* dan maksimum server(*cpu*).

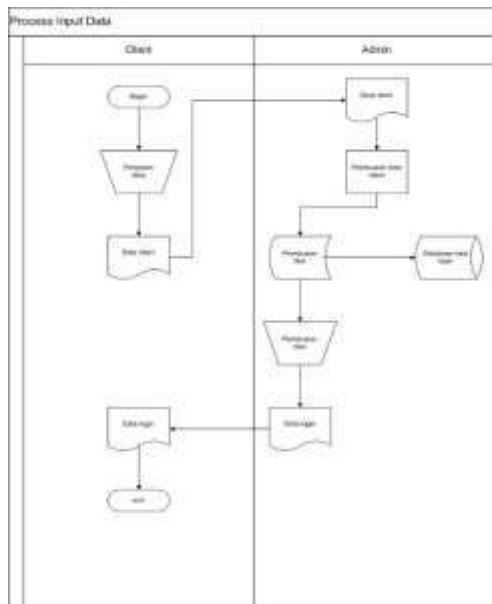
2.6 Coova Chilli

CoovaChilli adalah sebuah *software access controller open source*, berbasis pada *Chillispot project* yang sekarang sudah tidak aktif. CoovaChilli aktif di kembangkan oleh kontributor *Chillispot*. CoovaChilli adalah sebuah *software access control* yang kaya akan fitur, yang dapat memberikan *captive portal/walled-garden environment* dan menggunakan *RADIUS* untuk mengontrol akses dan akunting. CoovaChilli adalah bagian integral dari *firmware CoovaAP OpenWRT*.

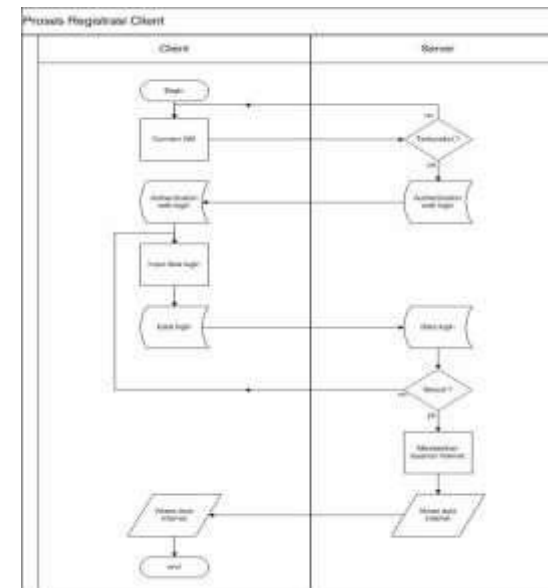
2.7 Putty

Putty adalah Software *SSH client* gratis untuk *Windows*, protokol *SSH* adalah cara aman untuk membuat sambungan aman ke sistem *remote*. Putty dapat meremote server yang menggunakan sistem operasi *linux*, sedangkan sistem operasi komputer menggunakan *windows*. Dengan menggunakan *putty* dengan kata lain dapat melakukan komunikasi antara dua sistem (komputer dan *remote server vps*).

- 3) *Admin input data login* yang akan disimpan pada *database* dalam penginputan *admin* memberikan (*user expiration*) pada *data login client* dengan batasan waktu selama 3 hari
- 4) Hasil dari data tersebut berupa tiket yang akan diberikan kepada *client*



Gambar 3- 1 Flowmap Proses Penginputan Data



Gambar 3- 2 Flowmap Proses Registrasi Client

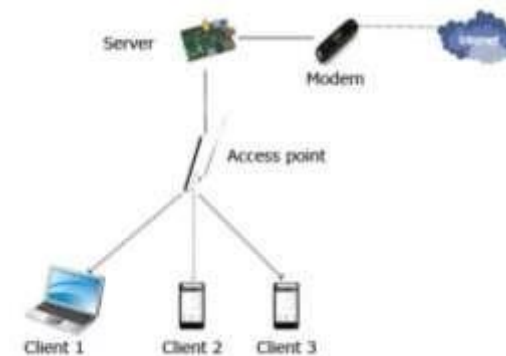
3.2 Analisis Sistem Keamanan Saat Ini

3.2.1 Proses Registrasi Client

- 1) *Client* masuk ke jaringan dengan *Wifi* transportasi dengan *SSID* "TRANSPORTAL"
- 2) *Client* akan di paksa untuk melalui port 443 <https://10.10.10.1/uam/> untuk melakukan *login*, *client* akan diminta memasukkan *username* dan *password* yang telah diberikan *admin*, server akan memeriksa kecocokan pada *database*
- 3) Apabila tidak cocok maka *client* akan kembali ke port 443 <https://10.10.10.1/uam/>
- 4) Apabila benar server akan memberikan *client* layanan untuk *access internet*.

3.3 Topologi Jaringan

Perancangan jaringan yang digunakan pada Proyek akhir adalah membuat jaringan dengan sebuah mini komputer *raspberry* yang di dalamnya terdapat sistem operasi *Raspbian wheezy* sebagai server



Gambar 3- 3 Topologi Jaringan

3.4 Kebutuhan Perangkat Keras dan Perangkat Lunak

3.4.1 Kebutuhan Perangkat Keras

Tabel 3- 1 Spesifikasi Komponen Jaringan

No	Nama Komponen	Spesifikasi	Deskripsi
1.	TP-LINK TL-WN722N	Tampilan USB 2.0 Gain Antena 4dBi Frekuensi 2.400-2.4835GHz	TP-LINK TL-WN722N berfungsi untuk menghubungkan komputer desktop atau notebook ke jaringan nirkabel.

Tabel 3- 2 Spesifikasi Komponen Raspberry Pi

No	Nama Komponen	Spesifikasi	Deskripsi
1.	CPU / Central Processing Unit	700 MHz ARM1176JZF-S core	Perangkat yang berfungsi untuk menerima dan melaksanakan perintah atau data dari perangkat keras lain / perangkat lunak
2.	GPU / Graphics Processing Unit	Broadcom VideoCore IV, OpenGL ES 2.0, 1080p30 h.264/MPEG-4 AVC high-profile decoder	Perangkat keras yang mendukung penggunaan grafik
3.	Memory (SDRAM)	512 MB	Perangkat keras yang berfungsi sebagai tempat penyimpanan sementara
4.	USB 2.0 ports: 2 (via integrated USB hub)		Perangkat keras yang berfungsi sebagai <i>port</i> untuk penambahan perangkat keras yang lain
5.	Video Outputs	Composite RCA (PAL & NTSC), HDMI (rev 1.3 & 1.4), raw LCD Panels via DSI 14 HDMI resolutions 7 from 6840×350 to 1920×1200 plus various PAL and NTSC standards.	Sebagai media untuk mendukung tampilan / <i>display</i> yang akan ditampilkan pada monitor
6.	Audio Outputs	3.5 mm jack, HDMI	Perangkat keras yang membantu dalam <i>output</i> suara

7.	Onboard Storage	SD / MMC / SDIO card slot 4 Gb	Perangkat keras yang digunakan untuk menyimpan perangkat tambahan seperti memori SD Card
8.	Daya Listrik	700 mA (3.5 W)	Perangkat yang berfungsi sebagai <i>Power Supply</i> , untuk menjadi tenaga Raspberry Pi
9.	Sistem Operasi	Debian GNU/Linux, Fedora, Arch Linux ARM, RISC OS, Raspbian Wheezy	Sistem operasi yang digunakan pada Raspberry Pi

Tabel 3- 3 Spesifikasi Laptop Administrator

Perangkat keras	Spesifikasi
	HP 431
Prosesor	Intel Core i3
Memory	DDR 2 GB
Harddisk	700 GB

Tabel 3- 4 Spesifikasi Client

Perangkat Keras	Client 1	Client 2	Client 3
Prosesor	Intel Core i5	1.2 GHz quad-core	1 GHz Cortex-A8
Memory	2 GB	1.5 GB	512 MB
Harddisk	500 GB	8 GB	16 GB

3.4.2 Kebutuhan Perangkat Lunak

Tabel 3- 5 Kebutuhan Perangkat Lunak

No	Perangkat	Perangkat Lunak
1.	Raspberry Pi	Raspbian wheezy
2.		Squid
3.		Apache
4.		Coova Chilli
5.		Freeradius
6.		Mysql
7.		Usb-modeswitch
8.		Wvdial
9.		Hostapd
10.		Openssl
11.		Daloradius
13.		Windows 7

15.	Laptop Admin	Putty
16.	Laptop Client	Windows 7
17.		Wireshark
18.		Cain & Able
19.		Idm
19.	Smartphone Client	Android
20.		iOs

3.5 Skenario Pengujian

Skenario pengujian dilakukan dengan 5 pengujian, yaitu :

1. Pengujian *login*
Pengujian ini dilakukan di dalam ruangan dengan beberapa *client* dalam pengujian ini akan menunjukkan fungsi *input* data *client* oleh *admin*.
2. Pengujian *bandwith*
Pengujian ini dilakukan di dalam ruangan dengan 1 *client*, dalam pengujian ini akan memperlihatkan kecepatan dari *client* yang di berikan oleh server.
3. Pengujian *filtering web*
Pengujian ini dilakukan di dalam ruangan dengan beberapa *client*, dalam pengujian ini akan memperlihatkan bahwa kewanaman *client browsing* berjalan.
4. Pengujian keamanan *login*
Pengujian ini dilakukan di dalam ruangan dengan 1 *client*. dalam pengujian ini akan memperlihatkan bahwa kewanaman data antar *client* dalam jaringan berjalan.
5. Pengujian *login* dan *filtering web* secara *mobile*
Hal yang dilakukan dalam pengujian ini memperlihatkan bahwa fungsi dari server berjalan dengan baik terhadap *client* yang melakukan *login* dan membuka *web* yang telah di blokir oleh server.

4. Implementasi dan Pengujian

4.1 Implementasi

Implementasi Keamanan jaringan dilakukan dengan menggunakan perangkat Raspberry Pi dengan sistem operasi *Raspbian Wheezy*. Raspberry Pi digunakan sebagai server penyedia layanan jaringan internet kepada *client* dengan keamanan *proxy*. Adapun langkah-langkah implementasi yang dilakukan adalah sebagai berikut :

1. Instalasi Raspbian Wheezy dan penyimpanan pada microSD
2. Konfigurasi network interfaces
3. Konfigurasi Modem dan TP-Link Wireless
4. Konfigurasi Captive Portal
 - a. Instalasi debhelper libssl-dev libcurl4-gnutls-dev
 - b. Instalasi mysql-server freeradius freeradius-mysql -openssl
 - c. download [coova-chilli-1.3.0.tar.gz](#), haserl-0.9.30.tar.gz, daloradius
 - d. Konfigurasi Freeradius dan Mysql
 - e. Konfigurasi Coovachilli, Apache ,dan Openssl
5. Konfigurasi Coovachilli
6. Konfigurasi Squid3

4.2 Pengujian

Pada pengujian di dalam ruangan ini *client* akan mencoba untuk melakukan *hacking* pada jaringan *TRANSPORTAL* yang diberikan oleh server untuk menguji keamanan pada jaringan, *client* akan melakukan *browsing* situs dan *download file* untuk melihat kinerja *squid*.

Tabel 4- 1 Pengujian di ruangan

No	Tujuan	Parameter yang diuji	Teknik pengujian	Hasil yang diharapkan	Hasil uji / status
----	--------	----------------------	------------------	-----------------------	--------------------

1	Dapatkah <i>squid</i> berjalan dengan baik, dan apakah keamanan <i>login</i> dapat di <i>sniffing</i> dan <i>brute force attack</i> .	<i>Bandwith, caching, filtering.</i> Keamanan pada saat <i>login</i> dan keamanan <i>portal</i> .	<i>Client</i> menghitung kecepatan <i>download</i> dengan <i>idm</i> . <i>Client</i> membuka web yang telah tersimpan di <i>caching</i> dan yang belum. <i>Client</i> membuka situs yang di <i>block</i> oleh <i>squid</i> . <i>Client</i> akan menggunakan <i>wireshrak</i> untuk <i>sniffing</i> dan <i>cain & able</i> untuk melakukan <i>brute force attack</i>	Pembagian <i>bandwith</i> , <i>caching</i> , dan <i>filtering</i> berjalan pada <i>client</i> , jaringan tidak dapat di <i>sniffing</i> dan <i>brute force attack</i> .	Hasil <i>squid</i> berjalan dengan baik terhadap <i>client</i> . Jaringan tidak dapat di <i>sniffing</i> dan <i>brute force attack</i> .
---	---	--	---	---	--

4.2.1 Pengujian Login

Pengujian ini di lakukan untuk melihat kemungkinan yang terjadi pada saat *client login*, dalam pengujian ini menggunakan 2 *client* yang telah terhubung pada jaringan dan dalam pengujian ini akan diperlihatkan tampak halaman *webmin daltoradius* untuk *input data login client* oleh *admin*.



Gambar 4- 1 halaman login daltoradius



Gambar 4- 2 Halaman login pada client 2

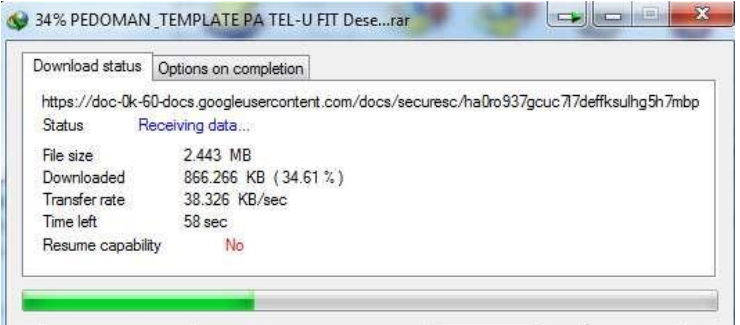
Tabel 4- 2 Pengujian Login					
Nama Field	Username	Password	Kelebihan yang diharapkan	Hasil Keluaran	Kesimpulan
Client 1	Kosong	Kosong	Kembali ke halaman login	Enginx hotspot login failed	Berhasil
	Huruf (A-Z, a-z+Angka)	Kosong	Kembali ke halaman login	Enginx hotspot login failed	Berhasil
	Kosong	Huruf (A-Z, a-z+Angka)	Kembali ke halaman login	Enginx hotspot login failed	Berhasil
	Huruf (A-Z, a-z+Angka)	Huruf (A-Z, a-z+Angka)	Halaman logout	Halaman logout	Berhasil
Client 2	Kosong	Kosong	Kembali ke halaman login	Enginx hotspot login failed	Berhasil
	Huruf (A-Z, a-z+Angka)	Kosong	Kembali ke halaman login	Enginx hotspot login failed	Berhasil
	Kosong	Huruf (A-Z, a-z+Angka)	Kembali ke halaman login	Enginx hotspot login failed	Berhasil
	Huruf (A-Z, a-z+Angka)	Huruf (A-Z, a-z+Angka)	Halaman logout	Halaman logout	Berhasil

4.2.2 Pengujian Bandwith

Pada pengujian ini *bandwith* yang diberikan server kepada *client* 1KB jika *client* mengakses *file* dengan *extention* yang ada pada *file download.txt* pada pukul 08.00 sampai 23.00 dan jika *client* melakukan *download* pada pukul 00.00 sampai 08.00 maka *client* mendapatkan *bandwith* tidak lebih dari 40KB. *Client* akan mendapatkan *bandwith* 65KB pukul 08.00 sampai 23.00 jika tidak melakukan *download* dengan *extention file* yang ada pada *file download.txt*, *client* akan mendapatkan *bandwith* 100KB pukul 00.00 sampai 08.00 jika tidak melakukan *download* dengan *extention file* yang ada pada *file download.txt*.



Gambar 4- 3 Kecepatan download file *.mp3 pada pukul 08.00-23.00 dengan menggunakan IDM



Gambar 4- 4 Kecepatan download file *.rar pada pukul 08.00-23.00 dengan menggunakan IDM

Tabel 4- 3 Pengujian Bandwith

Applikasi	type file download	Kecepatan rata	Bandwith default	Hasil
IDM	*.mp3	Ya	-	900 Bytes
	*.mp3	-	Ya	39KB
	*.rar	Ya	-	40 KB
	*.rar	-	Ya	94 KB

4.2.3 Pengujian Cache

Pada pengujian ini dilakukan dengan 1 *client* untuk membandingkan kecepatan situs yang sudah di *caching* dan yang belum di *caching* dengan *stopwatch*, dan

cara membuktikan bahwa *cache* sudah berfungsi bisa di monitor dengan meremote server dan membuka *file cache.log*.



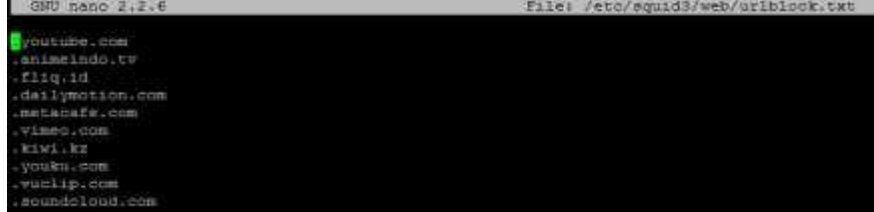
Gambar 4- 5 File cache.log pada server

Tabel 4- 4 Pengujian Cache

Situs yang belum di cache	Google.com	Lebih kurang 4.30 detik	Membuka situs yang sudah tersimpan pada <i>cache sever</i> lebih cepat di banding yang belum
	Facebook	Lebih kurang 4.49 detik	
Situs yang sudah di cache	Google.com	Lebih kurang 2.38 detik	
	Facebook.com	Lebih kurang 2.65 detik	

4.2.4 Pengujian Filtering Web

Pada pengujian ini dilakukan dengan 1 *client*, *client* akan membuka situs yang telah di blok oleh *squid* dengan menggunakan *acl domain web* yang dituju pada folder *urlblock.txt*.



Gambar 4- 6 File urlblock.txt



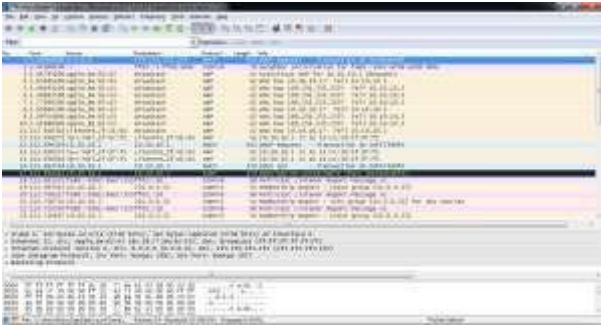
Gambar 4- 7 Halaman ERROR Soundcloud.com

Tabel 4- 5 Pengujian Filtering Situs

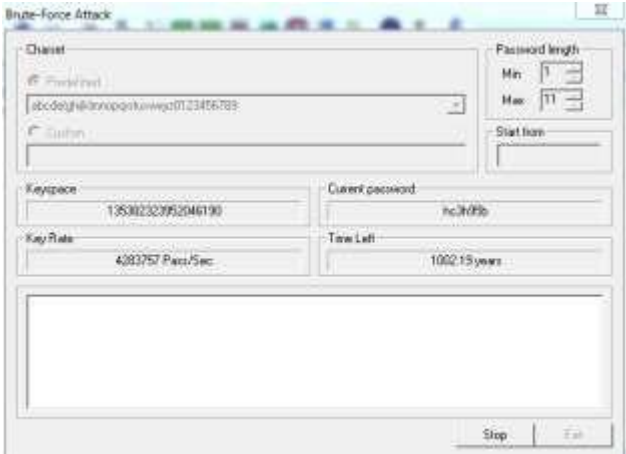
Nama	Nama situs	Hasil Keluaran	Kesimpulan
Situs yang di blok	Youtube.com	ERROR	Berhasil
	Animeindo.tv	ERROR	Berhasil
	Soundcloud.com	ERROR	Berhasil
Situs yang tidak di blok	Facebook.com	Situs web	Berhasil
	Mangacablog.com	Situs web	Berhasil

4.2.5 Pengujian Keamanan Login

Pengujian ini dilakukan dengan 2 teknik *sniffing* dan *brute force attack* pada jaringan. Tujuan dari pengujian ini untuk melihat keamanan dari sebuah *protokol* yang digunakan oleh *portal* dan keamanan *type password* yang digunakan untuk *client*.



Gambar 4- 8 Sniffing pada jaringan dengan menggunakan wireshrak



Gambar 4- 9 Brute force attack dengan password “002629rizky” yang telah di enkripsi oleh MD5

Tabel 4- 6 Pengujian Keamanan Login dengan teknik sniffing

Aplikasi	Protokol <i>portal</i>	Type password <i>client</i>	Hasil
	HTTPS	Clear text	Password tidak terlihat
Wireshrak	HTTPS	MD5	Password tidak terlihat

Tabel 4- 7 Pengujian Keamanan pada password dengan type password MD5 menggunakan teknik bruteforce attack

	Parameter	Password	Enkripsi Password MD5	Hasil	Waktu
Cain & able	Angka	002629	245aecda3b898afc0ada58968cb825f7	Password terlihat	< 1 menit
	Huruf	rizky	49d8712dd6ac9c3745d53cd4be48284c	Password terlihat	< 1 menit
	Angka dan huruf	002629rizky	e7abf5e7451d01b23703ce11ca4abc62	Password terlihat	> 1 menit

4.2.6 Pengujian Login dan Filtering Web secara Mobile

Pada pengujian ini hanya menunjukkan bahwa jaringan *Captive portal*, dan *filtering web* berjalan dengan baik pada transportasi dengan menggunakan 2 client.

Tabel 4- 8 Pengujian pada kendaraan roda 4

Kondisi	Keamanan login	Browsing	Hasil	Kesimpulan
Client 1	Ya	Google.com	berhasil	Webserver dan squid filtering web berjalan
	Ya	Youtube.com	ERROR	
Client 2	Ya	Detik.com	berhasil	
	Ya	Animeindo.tv	ERROR	
Client 3	Ya	Facebook.com	Berhasil	
	Ya	Souncloud.com	ERROR	

5. Kesimpulan dan Saran

5.1 Kesimpulan.

1. Dari hasil kerja pada proyek akhir ini dapat disimpulkan bahwa pembuatan server berbasis *security proxy* pada *raspberry pi* dengan membangun keamanan *proxy* pada transportasi membuat informasi data pengguna terhindar dari *sniffing* dan butuh waktu yang lama untuk melakukan *brute force attack* pada *password login* yang menggunakan huruf dan angka.

2. *Management bandwidth, cache, filtering web* pada aplikasi *squid proxy* berjalan dengan baik. Pengguna layanan mendapat *bandwidth* yang sama, akses *internet* lebih cepat untuk situs yang telah di *caching*, dan keamanan untuk mengakses internet di dapatkan oleh pengguna layanan. Kinerja *raspberry pi* dalam ruangan dan pada transportasi tidak memiliki masalah signifikan.

5.2 Saran

Adapun hal yang baru dan masih belum terpenuhi pada proyek akhir ini adalah bagaimana cara *admin* menambah data *login user* pada server *raspberry* yang berada pada transportasi dalam keadaan berjalan, dalam hal ini dibutuhkan sebuah *ISP* sebagai pusat jaringan server *raspberry* yang memiliki sebuah *IP Public* yang akan mempermudah *admin* melakukan *Remote SSH* secara jarak jauh.

6. Ucapan Terimakasih

Alhamdulillahirabbil'alamin, puji syukur kehadiran Allah SWT karena atas kehendak dan karunia-Nya penelitian berjudul "Implementasi Mini Server Yang Berbasis Security Proxy Dengan Menggunakan Raspberry PI Secara Portable" ini dapat terselesaikan.

Proyek akhir ini dibuat sebagai solusi dari permasalahan yang tengah dihadapi dengan menyediakan layanan *internet* dan keamanan jaringan pada transportasi umum menggunakan media *Raspberrry Pi* sebagai server yang berjalan. Proyek akhir ini dikerjakan dengan semangat yang tidak ada habisnya yang datang dari berbagai pihak.

Pada kesempatan ini Penulis mengucapkan terima kasih yang sebesar-besarnya kepada seluruh pihak yang telah bekerja sama dan membantu Penulis dalam menyelesaikan Proyek Akhir ini, secara khusus Penulis ingin menyampaikan terima kasih kepada :

1. Bapak Duddy Soegiarto dan Bapak Anang Sularsa yang telah sabar membimbing dalam menyelesaikan proyek akhir ini.
2. Teman-teman PCE1001 yang tidak pernah lelah dalam memberikan semangat.

Semoga proyek akhir ini dapat bermanfaat untuk seluruh pihak dan dapat digunakan dengan maksimal.

7. Daftar Pustaka

- [1] Devenport, Malcolm. "Proxy Server". 3 Januari 2011.
<https://www.scribd.com/doc/47413772/Proxy-Server>
- [2] Lilik Suheri, 2009, "Analisis Manajemen Hotspot dengan Captive Portal", Semarang, Skripsi.
- [3] Fernando, Hary. 2010. Studi dan Implementasi Sistem Keamanan Berbasis Web dengan Protokol SSL di Server Students Informatika ITB. Departemen Teknik Elektro, Program Studi Teknik Elektronika, Institut Teknologi Bandung. Via <http://budi.insan.co.id/>
- [4] (2013, Januari). Diambil Kembali dari
Mysql: (<http://www.maniacms.web.id/pengertian/mysql.html>)
- [5] Hassel, J. 2002. RADIUS. Sebastopol. O'Reilly.
- [6] Redra, Fredy, 2010, "Install Easy hotspot dan Konfigurasi Freeradius seta CoovaChilli di Ubuntu 10.04 LTS Server Edition". Diambil tanggal 22 Januari 2011 dari <http://ndra.gmib26.net/2010/06/install-easyhotspotdan-konfigurasi-freeradius-serta-coova-chilli-di-ubuntu-10-4-lts-server-edition/>
- [7] Urban, T. (2002). *Learn Cacti and design a robust Network Operation Center*. Packt Publishing Ltd.

*Allah swt telah memberi sepasang malaikat
nyata di dunia kepada saya yaitu Bapak
dan Ibu*