Analisa dan Perancangan Keamanan Mail Server Zimbra pada Sistem Operasi Ubuntu 8.04

I. E. S. W. Mangunkusumo [1], A. Lumenta [2], H. Wowor [3], A. Sinsuw [4]
(1)Mahasiswa (2)Pembimbing 1 (3)Pembimbing 2 (4)Pembimbing 3

imax_w@yahoo.com[1] arie.lumenta@unsrat.ac.id [2] hanswowor@unsrat.ac.id [3]
alicia.sinsuw@unsrat.ac.id[4]

Jurusan Teknik Elektro-FT, UNSRAT, Manado-95115

Abstract

Zimbra is an email application server freely licensed which has full features, easy of installation and management of mail server, despite mail server security issues being a major factor. Designing Security for mail server is very important which can prevent attacks from happening in cracker in computer networks such as port scanning, brute force and viruses. Well security can optimize the performance of the mail server itself.

This final project designing network security of zimbra mail server and it local network. Then zimbra mail server will be analyzed in terms of safety as well as the advantages and disadvantages of the networks.

Key Word: Mail Server, Network Security, Ubuntu, Zimbra

Abstrak

Zimbra merupakan aplikasi email server berlisensi bebas dimana memiliki fitur-fitur yang lengkap dan juga kemudahan untuk installasi maupun management mail server, meskipun masalah keamanan mail server menjadi faktor yang utama. Perancangan keamanan untuk mail server sangatlah penting dimana dapat mencegah serangan-serangan dari cracker yang terjadi di jaringan komputer seperti *port scanning, brute force* dan *virus*. Keamanan yang baik dapat mengoptimalkan kinerja dari mail server itu sendiri.

Pada tugas akhir ini akan dilakukan perancangan keamanan jaringan *mail server* zimbra. Kemudian *mail server* zimbra akan di analisa segi keamanannya serta kelebihan dan kekurangan pada jaringan tersebut.

Kata kunci : Keamanan Jaringan, Mail Server, Ubuntu, Zimbra

I. PENDAHULUAN

Dengan kemajuan teknologi, email menjadi salah satu media komunikasi yang penting. Penyedia layanan *email* atau yang sering disebut dengan *mail server* menjadi suatu aplikasi penting pada sebuah perusahaan atau instansi lainnya.

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.

Membangun jaringan *mail server* sekarang ini tidaklah cukup dengan hanya menginstall dan menjalankan saja, tapi ada beberapa proses juga yang dikerjakan agar supaya *mail server* yang dibuat dapat aman dan berjalan dengan lancar. Tuntutan teknologi juga yang menyebabkan setiap perusahaan harus mempunyai *mail server* sendiri sehingga menjadi lebih efisien dan juga disamping itu mengutamakan keamanan data terlebih mengantisipasi serangan-serangan *cracker* yang terjadi melalui jaringan komputer.

Contoh kasus yang saat ini berkembang adalah *brute force* dan *port scanning*, salah satu langkah yang dilakukan *cracker* sebelum masuk ke *server* yang ditargetkan adalah melakukan pengintaian. Cara yang dilakukan adalah dengan melakukan *port scanning* atau *brute force* untuk melihat *service-service* apa saja yang tersedia di server target. Sebagai contoh, hasil scanning dapat menunjukkan bahwa server target menjalankan program web *server* apache, mail *server sendmail*, dan seterusnya. Analogi hal ini dengan dunia nyata adalah dengan melihat-lihat apakah pintu rumah terkunci, merek kunci yang digunakan, jendela mana yang terbuka, apakah pagar terkunci (menggunakan *firewall* atau tidak) dan seterusnya. Dimana yang bersangkutan memang belum melakukan kegiatan pencurian atau penyerangan, akan tetapi kegiatan yang dilakukan sudah mencurigakan. (Budi Rahardjo mengatakan dalam forum IDCERT – Indonesia Computer Emergency Response Team).

Dewasa ini zimbra merupakan *software open source* mail *server* yang mulai banyak digunakan dengan kemudahan instalasi dan *management*. Di masa yang akan datang zimbra dapat menjadi suatu aplikasi *mail server* yang paling banyak digunakan seperti *postfix*, *sendmail* dan *qmail*.

Menurut penelitian yang dilakukan vavai kelebihan zimbra dibandingkan dengan *mail server* lain misalnya *exchange*, *qmail* dan *postfix* terletak pada *performance mail server* tersebut seperti jauh lebih ringan, *feature* jauh lebih lengkap, mendukung berbagai distro linux, sudah terintegrasi dengan anti spam, anti virus dan webmail.

Mail Server zimbra yang akan dibuat dalam rangka penelitian ini hanya bisa di installkan pada sistem operasi linux. Ubuntu Server adalah salah satu pilihan dari beberapa distro linux yang menunjang penginstallan

zimbra. Selanjutnya *mail server* zimbra ini akan diuji keamanannya untuk mengetahui apakah implementasi keamanan *mail server* yang dimaksud berjalan dengan baik dan aman.

II. LANDASAN TEORI

A. *Mail Server*

Mail server (juga dikenal sebagai sebuah *mail transfer agent* atau MTA, *mail router* atau *mailer* Internet) adalah sebuah aplikasi yang akan menerima email masuk dari pengguna lokal (orang-orang dalam satu domain) dan jarak jauh pengirim dan meneruskan *e-mail* keluar untuk pengiriman. Sebuah komputer yang didedikasikan untuk menjalankan aplikasi tersebut juga disebut sebagai *mail server*. (Danphi, 2008, 4)

Microsoft Exchange, *qmail*, *Exim* dan *sendmail* adalah lebih umum di antara program-program *server mail*. *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*. (Danphi, 2008,5)

B. *Zimbra*

Zimbra adalah sebuah produk *groupware* yang dibuat oleh Zimbra, Inc yang berlokasi di Palo Alto, California, Amerika Serikat. Pada masa awal- awalnya perusahaan ini di beli oleh Yahoo! tepatnya pada bulan september 2007.

Zimbra pada dasarnya sekelas dengan aplikasi Microsoft Exchange Server. Bedanya, Zimbra tersedia dalam 2 edisi, yaitu Open Source Edition dan Network Edition. Zimbra Open Source Edition menggunakan lisensi Mozilla Public License yang salah satu butir lisensinya menyatakan bahwa perubahan atau modifikasi yang dilakukan pada kode sumber Zimbra harus dikembalikan pada komunitas. (Danphi, 2008, 1)

Dalam proses transisi menuju era open source, bahwa peralihan *back office* harus dilakukan paling awal. Dimana kebutuhan para enduser menjadi perhatian. Untuk memenuhi tuntutan itu salah satu produk open source yang berkaitan dengan *email* yang memenuhi syarat tersebut adalah Zimbra Collaboration Suite. (Zaida, 2010, 1)

Zimbra Collaboration Suite adalah kolaborasi dari beberapa aplikasi *open source software*, diantaranya Apache Jetty, Postfix, OpenLDAP, dan MySQL. Kolaborasi ini menghasilkan *email server* yang power full dengan fitur-fitur yang lengkap. (Zaida, 2010, 1 & 2) (gambar 1).

Gambar 1. Logo Zimbra di akuisisi Yahoo

C. *Komponen Zimbra*

Berikut aplikasi *open source* yang digunakan Zimbra Collaboration Suite yang sudah merupakan aplikasi standar yang dipakai di dunia industry: (Zaida, 2010, 1 & 2)

- Jetty, aplikasi *server web* yang menjalankan aplikasi Zimbra.
- Postfix, aplikasi *open source* MTA (*Mail Transfer Agent*) yang menjalankan email server Zimbra.
- OpenLDAP, aplikasi *open source* sebagai *Lightweight Directory Accses Protocol* (LDAP) yang berguna untuk autentikasi user.
- MySQL, aplikasi database
- Lucane, aplikasi open-source power full text index dan search engine.
- Anti-Virus and anti-spam, aplikasi open source yang terdiri dari : Clamav anti virus scanner yang melindungi file dari serangan virus, SpamAssassin mail filter yang mengidentifikasi adanya Spam dan Amavisd-new sebagai interface antara MTA dengan yang lain.
- James/Sieve filtering, membuat filter untuk email.

D. *Aspek keamanan Sistem*

Dalam keamanan komputer dan sistem terdapat 5 aspek keamanan yaitu: (Stallings, 2011, 14)

1. Integrity
   Aspek ini menekankan bahwa keamanan sistem tidak boleh diubah tanpa seijin pemilik. Adanya virus seperti trojan horse, atau pemakai lain yang mengubah informasi tanpa ijin merupakan contoh masalah yang harus dihadapi. Sebuah *email* dapat saja disadap di tengah jalan, diubah isinya kemudian diteruskan ke alamat yang dituju. Dengan kata lain, integritas dari sistem sudah tidak terjaga. Penggunaan enkripsi atau *digital signature*, dapat mengatasi masalah ini. Contoh serangan lain adalah yang disebut "*man in the middle attack*" dimana seseorang menempatkan diri di tengah pembicaraan dan menyamar sebagai orang lain.
2. Authentication
   Aspek ini berhubungan dengan metoda untuk menyatakan bahwa informasi betul-betul asli, orang yang mengakses atau memberikan data adalah betul-betul orang yang dimaksud, atau *server* yang kita hubungi adalah betul-betul *server* yang asli. Masalah pertama, membuktikan keaslian dokumen,. Masalah kedua biasanya berhubungan dengan akses kontrol, yaitu berkaitan dengan pembatasan orang yang dapat mengakses data. Dalam hal ini pengguna harus menunjukkan bukti bahwa memang dia adalah pengguna yang sah, misalnya dengan menggunakan *password*, *biometric* (ciri-ciri khas orang), dan sejenisnya. *Authentication* biasanya diarahkan kepada orang (pengguna), namun tidak pernah ditujukan kepada *server* atau mesin.
3. Availability
   Aspek availability atau ketersediaan berhubungan dengan ketersediaan informasi ketika dibutuhkan. Sistem yang diserang atau dijebol dapat

menghambat atau meniadakan akses ke data. Contoh hambatan adalah serangan yang sering disebut dengan "*Denial of Service attack*" (DoS), dimana *server* dikirimi permintaan (biasanya palsu) yang bertubitubi atau permintaan yang diluar perkiraan sehingga tidak dapat melayani permintaan lain atau bahkan sampai *down, hang, crash*. Contoh lain adalah adanya *mailbomb*, dimana seorang pemakai dikirimi *e-mail* bertubi-tubi dengan ukuran yang besar sehingga sang pemakai tidak dapat membuka *e-mail*nya atau kesulitan mengakses *e-mail*nya .

4. Akses Kontrol
   Aspek ini berhubungan dengan cara pengaturan akses kepada data informasi. Hal ini biasanya berhubungan dengan klasifikasi data (*public, private, confidential, top secret*) & user (*quest, admin, top manager,* dsb.), mekanisme autentikasi dan juga privasi. Akses Kontrol seringkali dilakukan dengan menggunakan kombinasi *username* / *password* atau dengan menggunakan mekanisme lain (seperti kartu pintar atau *smartcard*).
5. Non-repudiation
   Aspek ini menjaga agar seseorang tidak dapat menyangkal telah melakukan sebuah transaksi. Sebagai contoh, seseorang yang mengirimkan email untuk memesan barang tidak dapat menyangkal bahwa dia telah mengirimkan email tersebut. Aspek ini sangat penting dalam hal transaksi. Penggunaan *digital signature* dan teknologi kriptografi secara umum dapat menjaga aspek ini.

E. Tahapan-tahapan hacking

Dalam tahapan melakukan Hacking, terdapat 5 tahapan yaitu: (Graves, 2010, 8)

1. *Reconnaissance* (Pengintaian)
   Reconnaissance atau pengintaian adalah tahap mengumpulkan data. Peretas akan mengumpulkan semua data sebanyak-banyaknya mengenai target. Proses pengintaian terbagi menjadi dua yaitu pengintaian secara aktif dan pasif.

- Pengintaian secara pasif (*Passive Reconnai*ssance) adalah proses pengumpulan data tanpa berhubungan langsung dengan target. Dalam Passive Reconnaissance ada beberapa teknik yang dipakai untuk mendapatkan data target misalnya *sniffing*. *Sniffing* adalah teknik dimana *hacker* mendengar paket data pada suatu jaringan yang melewati komputer *hacker*. Ada 2 tipe *Sniffing* yaitu *Passive Sniffing* dimana hacker hanya berdiam diri menunggu datangnya data dan *Active Sniffing* dimana hacker memaksa data untuk melewati komputernya.
- Pengintaian secara aktif (*active Reconnai*ssance) adalah proses pengumpulan data dengan berhubungan dengan target. Peretas melakukan aktifitas dalam lingkungan target untuk mendapatkan informasi sebanyak-banyaknya.

2. *Scanning*
   Scanning merupakan tanda dari dimulainya sebuah serangan oleh peretas (*pre-attack*). Pada tahap ini, peretas akan mencari berbagai kemungkinan yang dapat digunakan untuk mengambil alih komputer atau sistem dari target. Tahapan ini dapat dilakukan jika informasi yang didapat pada tahap *reconnaissance* mencukupi sehingga peretas bisa mencari "jalan masuk" untuk menguasai sistem. Berbagai peralatan (*tools*) dapat membantu seorang peretas untuk melalui tahapan ini misalnya *port scanner, network mapper,* dll.
3. *Gaining Access* (Pengambilan alih)
   Pada tahap ini *hacker* (peretas) akan memulai proses penyerangan terhadap komputer atau sistem korban setelah peretas mengetahui kelemahan dari komputer atau sistem korban. Dalam tahap ini *hacker* mengambil hak akses sistem korban. Jika sudah mendapatkan hak akses maka *hacker* bisa melakukan apapun terhadap sistem.
4. *Maintaining Access* (Memelihara akses)
   Setelah mendapatkan kekuasaan terhadap suatu sistem, terdapat kemungkinan ulah peretas diketahui oleh korban sehingga akan timbul tindakan dari korban untuk memperbaiki kelemahan dari sistemnya. Agar dapat mempertahankan kekuasaannya, peretas bahkan bisa memperbaiki beberapa kelemahan yang ada pada komputer atau sistem korban agar peretas lain tidak bisa memanfaatkannya untuk mengambil alih komputer atau sistem yang sama. Seorang peretas akan mempertahankan kekuasaannya terhadap sistem tersebut.
5. *Covering Tracks* (Menutupi jejak)
   Agar kegiatan dari seorang peretas tidak diketahui oleh korban, maka ada tahapan saat peretas menghapus *log file* serta menutupi semua jejak yang mungkin ditinggalkan. Maka itu seringkali korban tidak menyadari akan aktivitas peretas karena mereka membuatnya dalam modus tersembunyi (*hidden*).

F. Ubuntu

Ubuntu merupakan distro linux, yang dibuat berdasarkan pengembangan dari debian. Ubuntu menyediakan sistem yang berdasarkan pada Debian dengan frekuensi rilis yang teratur, ketersediaan dukungan untuk pengguna perusahaan, dan tampilan *desktop* yang lebih dipertimbangkan. Ubuntu melakukan penyebaran dengan cara yang hampir sama dengan selalu rilis disertai aplikasi terbaru berbasis *open source*. Proyek Ubuntu disponsori oleh Canonical Inc (perusahaan milik Mark Shuttleworth). (Handaya, Suteja & Ashari, 2010, 61)

III. METODOLOGI PENELITIAN

A. Tempat dan Waktu Penelitian

Dalam pelaksanaan tugas akhir ini penulis mengambil tempat penelitian pada Ruang Laboratorium Sistem Komputer (LSK), Jurusan Teknik Elektro, Fakultas Teknik Universitas Sam Ratulangi (UNSRAT), dan rumah penulis. Waktu penelitian yang dilakukan dalam penyelesaian tugas akhir ini adalah 8 bulan.

B. Bahan dan Peralatan

Dalam mengerjakan tugas akhir ini mulai dari mendesain sampai tahap pemrograman penulis menggunakan perlengkapan komputer sebagai media untuk menjalankan program. Secara lebih spesifik perlengkapan komputer beserta pendukung yang digunakan yaitu:

1. Spesifikasi Komputer Client
   a. Sistem Operasi: Windows 7
   b. Processor Intel Core i5 - 2310M 2.4Ghz
   c. Memory RAM 4GB DDR3
   d. Harddisk 500Gb HDD
2. Spesifikasi Komputer Audit
   a. Sistem Operasi: Windows 7
   b. Processsor AMD
   c. Memory RAM 2GB
   d. Harddisk 320Gb HDD
3. Spesifikasi Komputer Server
   a. Sistem Operasi Ubuntu 8.04 32 bit
   b. Processor Intel Core 2 Duo T5300 1.73 Ghz
   c. Memory RAM 3GB DDR2
   d. Harddisk 320Gb HDD

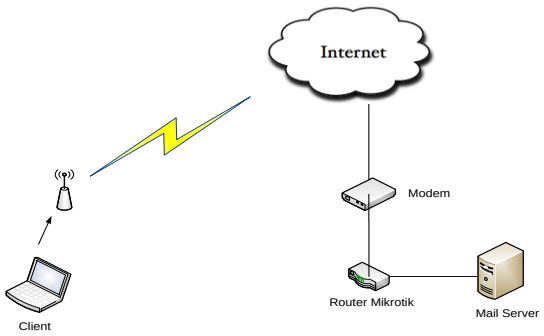
Adapun Aplikasi - aplikasi yang di pakai penulis dalam penyelesaian Tugas Akhir adalah:

1. Win SCP
2. Putty
3. Road'kill Port Scan
4. Zimbra Collaboration Suite 5.0.10 32 bit
5. Mikrotik Winbox Loader v2.2.16
6. Brutus

C. Prosedur Penelitian

Prosedur yang dilakukan dalam pembuatan analisa keamanan Mail Server Zimbra pada sistem operasi Ubuntu adalah sebagai berikut:

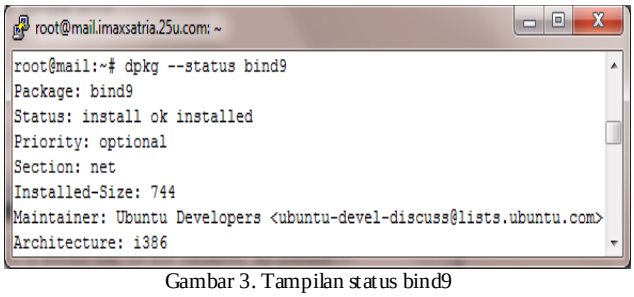
1. Sebelum melakukan penelitian, penulis terlebih dahulu melakukan studi literatur. Penulis mencari materi-materi yang berhubungan dengan keamanan mail server, pembuatan mail server khusunya zimbra.
2. Setelah mendapatkan informasi yang dibutuhkan, maka penulis mencari program-program pendukung dalam pembuatan tugas akhir.
3. Penulis menggunakan program brutus untuk mengecek keamanan *mail server*, Zimbra Collaboration Suite 5.0.10 untuk membangun *mail server*.
4. Selanjutnya penulis melakukan instalasi aplikasi yang akan digunakan.
5. Setelah itu penulis melakukan konfigurasi *mail server* yang dikombinasikan dengan Mikrotik.
6. Kemudian penulis juga melakukan penginstalan Brutus pada komputer audit untuk menganalisis keamanan *mail server*.
7. Penulis menguji keamanan mail server yang berjalan pada jaringan internet dengan Brutus.



Gambar 2. Skema topologi Mail Server

TABEL I. PORT-PORT YANG DIGUNAKAN ZIMBRA

Nama Service	Port
Remote Queue	22
Postfix	25
HTTP	80
POP3	110
IMAP	143
LDAP	389
HTTPS	443
Mailboxd IMAP SSL	993
Mailboxd POP SSL	995
Mailboxd LMTP	7025



Gambar 3. Tampilan status bind9

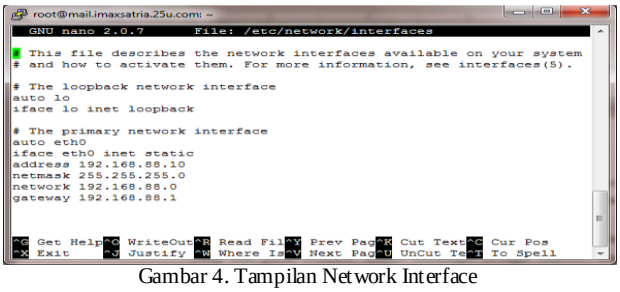
D. Perancangan Mail Server

Adapun perancangan jaringan Mail Server di buat penulis menggunakan router mikrotik adalah sebagai berikut (gambar 2).

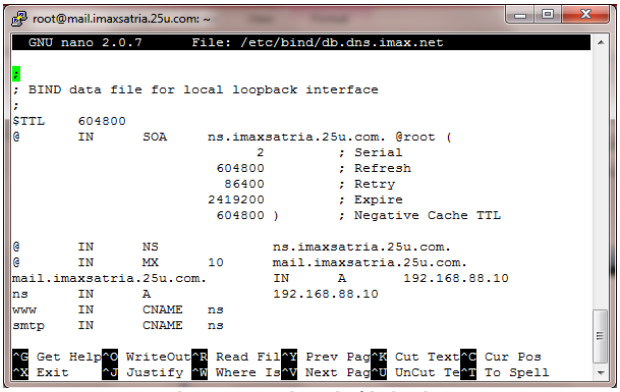
Awalnya pada *Server* Ubuntu di installkan Bind untuk membuat sebuah *Domain Name Server* (DNS), setelah DNS berhasil dibuat maka Zimbra akan siap diinstalkan ke dalam *server*. Jika terdapat kesalahan pada DNS maka penginstallan Zimbra tidak akan berjalan dengan lancar. Perlu diperhatikan juga *port-port* yang digunakan oleh Zimbra agar tidak berjalan, karena saat port-port tersebut berjalan maka Zimbra tidak akan berjalan dengan baik. Port-Port yang digunakan oleh Zimbra adalah (tabel I).

E. Proses Konfigurasi network dan DNS

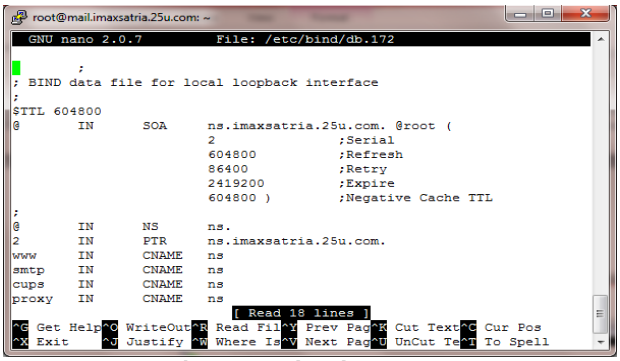
1. Masuk ke root, kemudian masukan password setelah masuk ke dalam *root* install bind9 dengan perintah # apt-get install bind9.
2. Setelah selesai melakukan penginstallan cek status *bind* dengan perintah #dpkg --status bind9 (gambar 3).



Gambar 4. Tampilan Network Interface

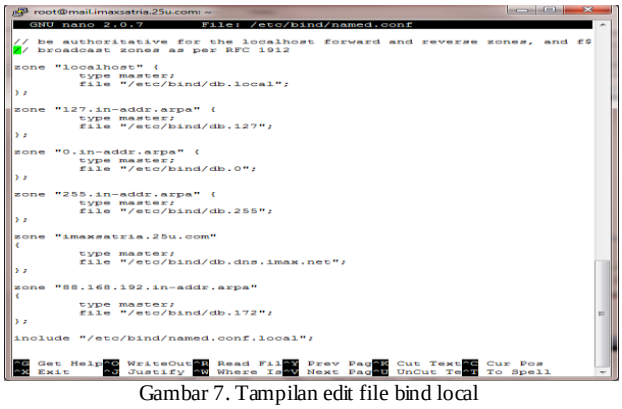


Gambar 5. Tampilan edit file bind DNS

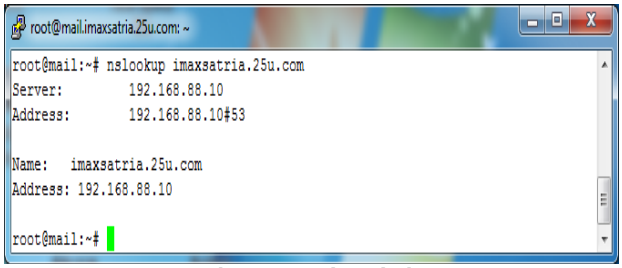


Gambar 6. Tampilan edit zone reverse

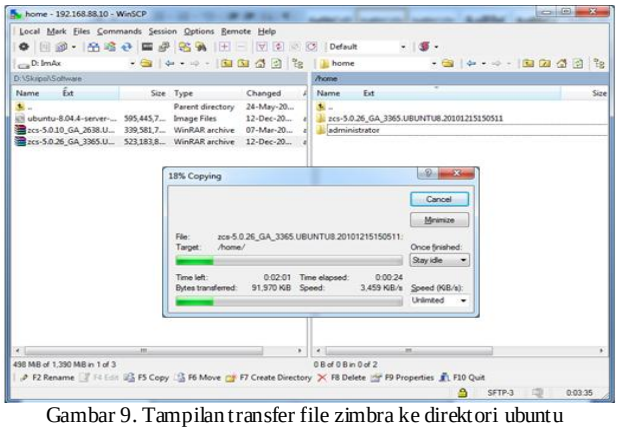
3. Mengkonfigurasi *Network interface* jaringan dengan perintah # nano /etc/network/interfaces (gambar 4)
4. Membuat file untuk zone imaxsatria.25u.com dengan mengetikkan perintah #nano/etc/bind/db.dns.imax.net (gambar 5)
5. Membuat *zone reverse* untuk imaxsatria.25u.com dengan mengetikan perintah # nano /etc/bind/db.172 (gambar 6).
6. Mengkonfigurasi bind9 yang sudah diinstall pada *server* dengan mengetik perintah # `nano /etc/bind/named.conf` (gambar 7).
7. Melakukan pengujian DNS dengan nslookup : # nslookup imaxsatria.25u.com (gambar 8)
8. Download file Zimbra Collaboration Suite 5.0.26 32 bit untuk Sistem Operasi Ubuntu di www.zimbra.com. Setelah Zimbra didownload pindahkan file tar Zimbra ke home Sistem operasi Ubuntu 8.04 memakai aplikasi FTP WinSCP dengan mengakses alamat IP dari Ubuntu.
9. Pindahkan file tar Zimbra dari direktori tempat penyimpanan ke direktori home Ubuntu 8.04 dengan cara copy file (gambar 9).



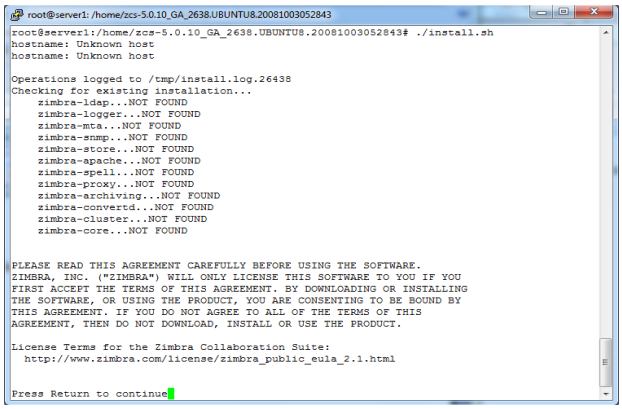
Gambar 7. Tampilan edit file bind local



Gambar 8. Tampilan nslookup



Gambar 9. Tampilan transfer file zimbra ke direktori ubuntu

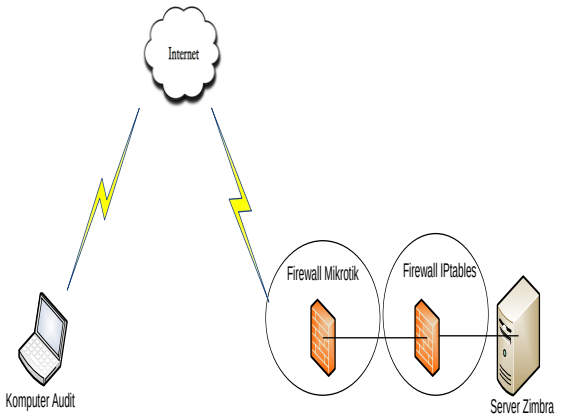


Gambar 10. Tampilan Install file Zimbra

F. Installasi Zimbra

Setelah file yang di download dipindahkan ke direktori Ubuntu lalu memulai installasi dengan mengeksekusi file dengan perintah install.sh (gambar 10).

Gambar 11. Tampilan pembuatan account pada change ip



Gambar 12. Skema Keamanan

G. Konfigurasi pada router mikrotik

1. Pertama buat dulu modem dalam keadaan *bridge* masuk ke modem lalu masuk ke setup lalu setting Bridge. Isi VCI dan VPI sesuai setting modem adsl. Biasanya VPI=8, VCI=81.
2. Pindah ke router mikrotik masuk ke new terminal, ketikan settingan mikrotik ini:

```
/ interface pppoe-client
add name="Speddy" max-mtu=1480 max-
mru=1480 interface=ether1
 user="173*********@telkom.net"
password="*******" profile=default
 service-name="internet" ac-name=""
add-default-route=yes dial-on-
demand=no
 use-peer-dns=yes
allow=pap,chap,mschap1,mschap2
disabled=no
```

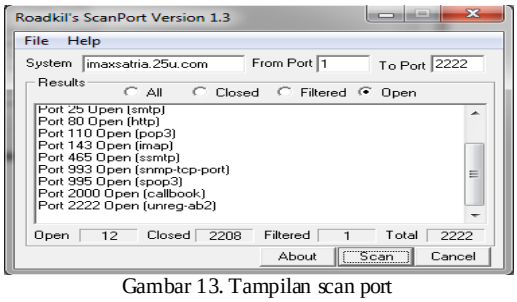
username dan password di isi sesuai username dan password speedy.

H. Pembuatan DNS pada change ip

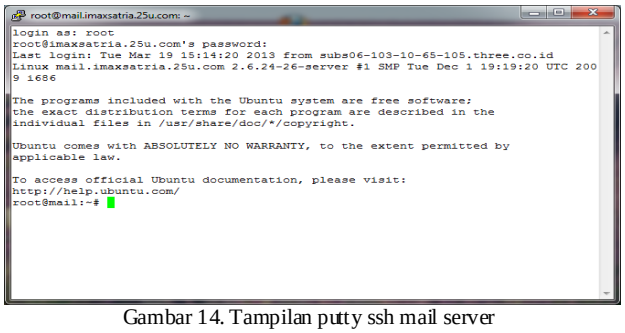
Pembuatan DNS di buat secara gratis pada website www.changeip.com dan untuk membuatnya harus mendaftar pada websitenya (gambar 11).

IV. PENGUJIAN DAN PEMBAHASAN

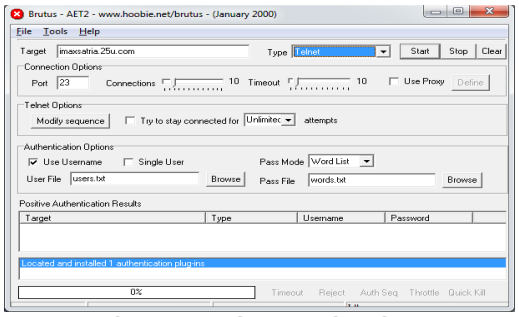
Setelah melakukan proses perancangan dan konfigurasi pada mail server, pada bab ini penulis akan melakukan implementasi dan pengujian keamanan mail server zimbra dan jaringan lokalnya terhadap jaringan publik (gambar 12).



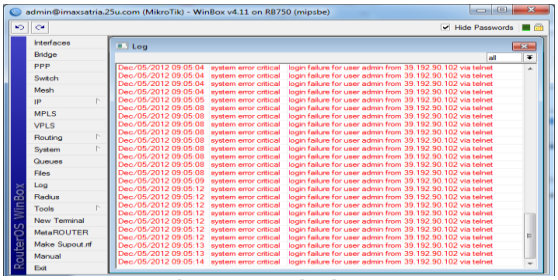
Gambar 13. Tampilan scan port



Gambar 14. Tampilan putty ssh mail server



Gambar 15. Tampilan putty ssh mail server

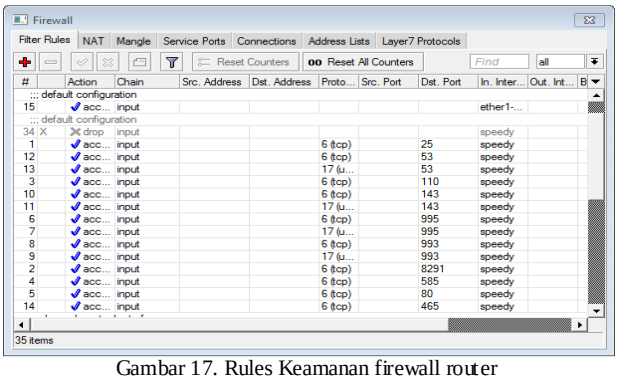


Gambar 16. Tampilan log router

Melakukan pengujian dimana implementasi keamanan belum diterapkan. Pengujian yang dilakukan adalah : 1. Pengintaian.

2. Mencoba masuk melalui *ssh mail server*.
3. *Brute force attack* pada *mail server*.

1. Pengintaian
   Melakukan pengintaian menggunakan *software road kill's scan port*. *Scan open port* 1-2222 (gambar 13).
2. Mencoba masuk melalui *ssh mail server*.
   Masuk menggunakan putty untuk *ssh* ke *mail server* dari jaringan luar ke jaringan lokal *mail server* (gambar 14).
3. *Brute force attack* pada *mail server*.
   Melakukan *brute force login* pada *gateway* jaringan lokal *mail server* dengan menggunakan *software brutus* (gambar 15). Tampilan *log router* setelah program brutus dijalankan (gambar 16).



Gambar 17. Rules Keamanan firewall router

Setelah dilakukan pengujian pada mail server dimana implementasi keamanan belum diterapkan maka Implementasi keamanan pada *mail server* zimbra untuk mengatasi masalah keamanan adalah sebagai berikut:

1. Membuka *port* yang digunakan dan menutup *port* yang tidak digunakan untuk akses dari jaringan luar ke jaringan lokal.
2. Membuat *rules firewall* untuk *port forwarding*.
3. Membuat *rules* iptables pada *server* zimbra.
4. Membuat *rules brute force login prevention*.

A. Membuka port yang digunakan dan menutup port yang tidak digunakan untuk akses dari jaringan luar ke jaringan lokal.

Port-Port yang diperlukan pada Mail Server (gambar 17).

- Port 21 (ftp) : Port protokol Internet yang berjalan di dalam lapisan aplikasi yang merupakan standar untuk pentransferan berkas (*file*).
- Port 22 (ssh) : Port ini ini adalah port standar untuk SSH, biasanya diubah oleh pengelola server untuk alasan keamanan.
- Port 23 (telnet) : Port yang digunakan client telnet untuk hubungan dengan server telnet.
- Port 25 (SMTP) : Port ini digunakan untuk mengirim email, baik dari server email ke internet, maupun dari internet ke server zimbra.
- Port 53 (domain) : DNS, atau *Domain Name Server* port. *Name Server* menggunakan *port* ini, untuk penerjamahan nama domain ke IP *Address*.
- Port 80 (http) : WWW atau HTTP port *server web*. Port yang paling umum digunakan di Internet.
- Port 110 (pop3) : alias *Post Office Protocol*, port server pop mail. Apabila ingin mengambil email yang tersimpan di server dapat menggunakan teknologi POP3 yang berjalan di port ini.
- Port 143 (imap) : *Interim Mail Access Protocol*. Merupakan aplikasi yang memungkinkan kita membaca email yang berada di *server* dari komputer, protokol ini sedikit berbeda dengan POP.
- Port 585 : Untuk *Blackberry* juga dapat mengakses Zimbra *server*.
- Port 465 (ssmtp) : SMTP atas SSL, protokol *server* email.
- Port 993 (snmp-tcp-*port*) : IMAP melalui SSL.
- Port 995 (spop3) : POP melalui SSL.
- Port 2000 (callbook) : *OpenWindows*; menampilkan data dan *keystroke*; *block*.
- Port 2222 (unreg-ab2) : *port* modifikasi untuk ssh pada zimbra *Server*.
- Port 8291 : *port* untuk mengakses winbox

B. Membuat rules firewall untuk port forwarding.

Port- port yang diteruskan ke IP server email :

- Port 25 (SMTP) – TCP: *Port* ini digunakan untuk mengirim email, baik dari *server* email ke internet, maupun dari internet ke *server* zimbra.
- Port 110 (POP3) - TCP: port ini digunakan untuk membuat *server* email Zimbra dapat diakses dengan menggunakan email *client* yang sudah mendukung POP3.
- Port 143 (IMAP) – TCP dan UDP: port in digunakan untuk membuat *server* email Zimbra dapat diakses dengan menggunakan aplikasi yang mendukung pembaca email dengan protokol IMAP, seperti program email *client* di perangkat bergerak seperti *mobile phone*, komputer tablet, dan Blackberry.
- Port 995 (POP3S) – TCP dan UDP: port ini digunakan untuk mengakses *server email* dengan aplikasi email *client* melalui *port* POP yang sudah diamankan. Menggunakan *port* ini lebih aman dari pada port 110 sebelumnya, namun efeknya menjadi kurang nyaman karena banyaknya permintaan konfirmasi sertifikat keamanan.
- Port 993 (IMAPS) – TCP dan UDP: port ini digunakan untuk mengakses *server email* dengan aplikasi email *client* melalui *port* IMAP yang sudah diamankan. Menggunakan port ini lebih aman dibandingkan *port* 143 sebelumnya, efeknya akan sama seperti *port* 995.
- Untuk Blackberry juga dapat mengakses Zimbra *server*, *port* 585 – TCP yang harus diarahkan ke IP *server* Zimbra.
- Mengakses *Zimbra Web Client* via https, *port* 443 – TCP yang harus di arahkan pada IP *server* Zimbra (gambar 15).

C. Membuat rules iptables pada server zimbra

Agar mail server zimbra tidak dapat diakses dari jaringan luar maka digunakan *iptables* untuk membatasi akses sebagai pengamanan Zimbra dengan memblok semua *port* yang tidak digunakan pada ip 192.168.88.10/24. Rules iptables ini juga berfungsi agar port ssh dari zimbra tidak dapat di scan oleh cracker. Buat perintah *iptables* dan simpan pada *file iptables.rules*

```
# Mendefinisikan INPUT
1. -A INPUT -i lo -j ACCEPT
2. -A INPUT -p icmp -m icmp --icmp-type
   any -j ACCEPT
3. -A INPUT -m state --state
   RELATED,ESTABLISHED -j ACCEPT

# Mengizinkan SSH dan SNMP
1. -A INPUT -s 192.168.88.0/255.255.255.0
   -p tcp -m state --state NEW -m tcp --
   dport 2222 -j ACCEPT
2. -A INPUT -s 192.168.88.0/255.255.255.0
   -p udp -m state --state NEW -m udp --
   dport 161 -j ACCEPT
```

```
# Mengizinkan port-port Zimbra
-A INPUT -p tcp -m state --state NEW -m
tcp --dport 25 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m
tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m
tcp --dport 110 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m
tcp --dport 143 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m
tcp --dport 389 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m
tcp --dport 443 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m
tcp --dport 465 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m
tcp --dport 993 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m
tcp --dport 995 -j ACCEPT
-A INPUT -s 192.168.88.0/255.255.255.0 -p
tcp -m state --state NEW -m tcp --dport
7071 -j ACCEPT

# Menolak Port-Port Lain selain port yang
diatas
-A INPUT -j REJECT --reject-with icmp-
host-prohibited
COMMIT
```

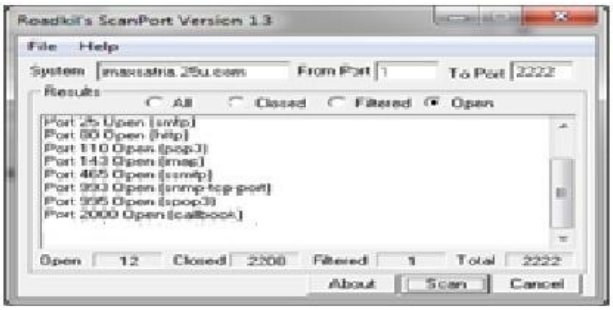
D. Membuat rules brute force login prevention

Rules Brute Force Login Prevention pada port 23

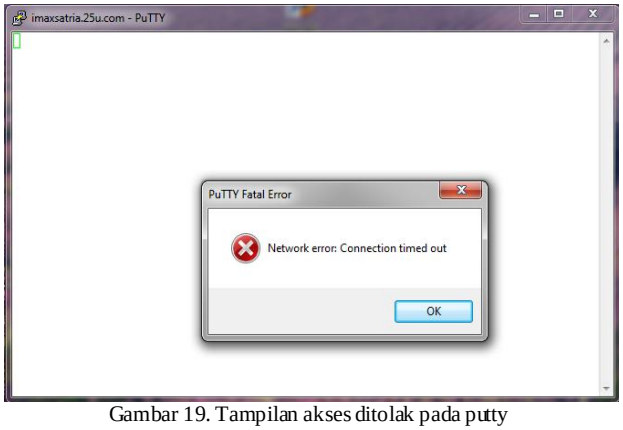
- `add chain=input protocol=tcp dst-port=23 src-address-list=telnet_blacklist action=drop \ comment="drop telnet brute forcers" disabled=no`
- `add chain=input protocol=tcp dst-port=23 connection-state=new \ src-address-list=telnet_stage3 action=add-src-to-address-list address-list=telnet_blacklist \ address-list-timeout=10d comment="" disabled=no`
- `add chain=input protocol=tcp dst-port=23 connection-state=new \ src-address-list=telnet_stage2 action=add-src-to-address-list address-list=telnet_stage3 \ address-list-timeout=1m comment="" disabled=no`
- `add chain=input protocol=tcp dst-port=23 connection-state=new src-address-list=telnet_stage1 \ action=add-src-to-address-list address-list=telnet_stage2 address-list-timeout=1m comment="" disabled=no`
- `add chain=input protocol=tcp dst-port=23 connection-state=new action=add-src-to-address-list \ address-list=telnet_stage1 address-list-timeout=1m comment="" disabled=no`

Pembahasan

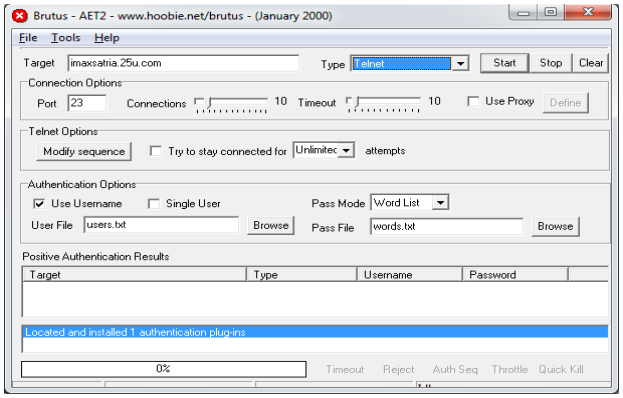
Melakukan pengujian kembali setelah implementasi keamanan *mail server* zimbra yang telah diterapkan adalah sebagai berikut :



Gambar 18. Tampilan port 2222 tidak terdeteksi

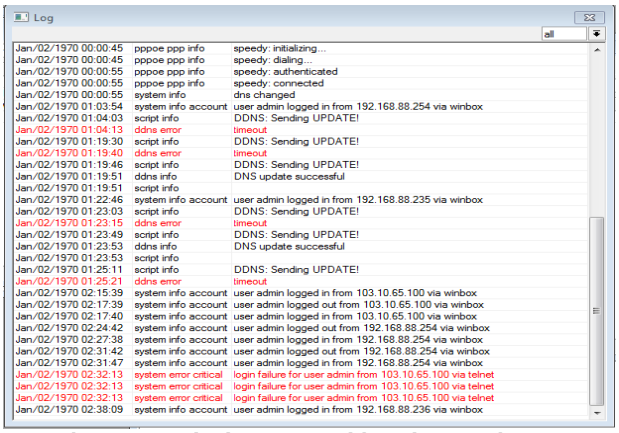


Gambar 19. Tampilan akses ditolak pada putty



Gambar 20. Tampilan aplikasi brute force

1. Pengujian *scanning port*.
   Port yang tidak ditampilkan pada aplikasi *scan port* merupakan *port ssh* dari *mail server* yaitu *port* 2222. Implementasi keamanan dari *iptables* yang membuat *port ssh mail server* tidak terdeteksi dimana pada pengujian sebelumnya implementasi keamanan yang diterapkan, port 2222 dapat terdekteksi (gambar 18).
2. Pengujian masuk melalui *putty* ke *ssh mail server*.
   Setelah *rules iptables* diterapkan maka jaringan luar tidak dapat melakukan *ssh* ke jaringan lokal *mail server*. *Rules iptables* yang dibuat berfungsi untuk pembatasan akses ke *mail server* dimana hanya jaringan lokal yang dapat mengakses *ssh mail server* (gambar 17).
3. Pengujian *brute force attack* pada *mail server*.
   Pengujian setelah menggunakan rules *bruteforce login prevention*, jalankan kembali program Brutus untuk melakukan *bruteforce* ke port telnet dari *server* (gambar 20).



Gambar 21. Tampilan log router setelah implementasi keamanan

Tampilan log dari *router* menunjukkan adanya *login* dari ip address 103.10.65.100 yang masuk dalam *rules brute force login prevention* (gambar 21).

Cara kerja *rules bruteforce login prevention* adalah untuk setiap percobaan *login* selama 2 kali (2 stage) dalam 1 menit menggunakan IP yang sama maka untuk percobaan login yang ketiga (stage 3) akan masuk pada *blacklist* IP dimana IP tersebut tidak akan bisa masuk lagi untuk mencoba login selama 10 hari kedepan.

V. KESIMPULAN

Dari pengujian yang di lakukan pada perancangan keamanan *mail server*, penulis dapat menarik beberapa kesimpulan yaitu:

1. Dengan menutup *port-port* yang tidak digunakan akan membatasi akses ke *port* selain dari port yang digunakan oleh *service mail server* tersebut. Hal ini menghindarkan mail server dari exploitasi *port* yang tidak digunakan.
2. Mengubah *port* SSH *mail server* sangat membantu untuk mencegah akses *hacker* untuk masuk kedalam *mail server*.
3. *Rules brute force login prevention* berfungsi untuk mencegah serangan ke *mail server* dengan metode *bruteforce.*

DAFTAR PUSTAKA

[1] T. Athailah, "*Panduan Membuat Email Server dengan Zimbra"*, Jasakom, Jakarta, 2012.

[2] D. Dhamdhere, "*Operating Systems A Concept-Based Approach*", McGraw-Hill Companies, 2009.

[3] K. Graves, "*Certified Ethical Hacker Study Guide"*, Wiley Publishing, Inc, Indianapolis, Indiana, 2010.

[4] F. Danphi, "*Zimbra Mail Server with Ubuntu 8.04*", Informatika, Jakarta, 2010.

[5] W. Handaya, B. Suteja, A. Ashari, "*Linux System Administrator",* Informatika, 2010.

[6] H. Linto, C. Azis, "*Panduan Lengkap Menguasai Router Masa Depan Menggunkan Mikrotik RouterOS*™", Penerbit Andi, Yogjakarta, 2008.

[7] J.Stanger and T.Lane, "*Proffing Linux: A Guide to Open Source Security",* Syngress Publishing Inc, 2001.

[8] L. Long and N. Long, "*Computers Information Technology in Perspective 10th edition*", Pearson Education Inc,Upper saddle River, New Jersey, 2002.

[9] A. Silbersachatz, P. Galvin, G. Gagne, "*Operating System Concepts Essentials",* John Wiley & Sons Inc,2011.

[10] M. Sobell, "*A Practical Guide to Linux Commands, Editors and Shell Programming",* Prentice Hall, 2005.

[11] S. Shah and W. Soyinka, "*Linux Administration A Beginner's Guide –Fourth Edition",* McGraw-Hill, 2005.

[12] W. Stallings, *"Network Security Essential Applications and Standards 4th edition",* Pearson Education Inc, New Jersey, 2011.

[13] B. Sosinsky, "*Network Bible*", Wiley Publishing, Inc, Indianapolis, Indiana, 2009.

[14] Sutanto, "*Certified Ethical Hacker 200% Illegal",* Jasakom Publisher, 2009.

[15] J.R. Vacca, "*Computer and Information Security Handbook*", Morgan Kaufmann Publishers,ElsevierInc, Burlington, 2009.

[16] E. Zaida, "*Panduan praktis Membangun Server Email Enterprise Dengan Zimbra",* Info Linux Dian Rakyat, Jakarta, 2010.