

MEMBANGUN MAIL SERVER BERBASIS LINUX DENGAN MENGGUNAKAN POSTFIX



Diajukan Oleh :

DWI WAHYUDI
NIM. M3304060

TUGAS AKHIR

ditulis dan diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Ahli Madya Ilmu Komputer

**PROGRAM DIPLOMA III ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SEBELAS MARET
SURAKARTA
2007**

**TUGAS AKHIR
MEMBANGUN MAIL SERVER BERBASIS LINUX DENGAN
MENGUNAKAN POSTFIX**

yang disusun oleh
DWI WAHYUDI
M3304060

dibimbing oleh

Pembimbing I,

Pembimbing II,

Hartono, S.Si
NIP. 500 127 069

Mohtar Yuniarto, M.Si
NIP. 132 310 079

telah dipertahankan didepan Dewan Penguji
pada hari Senin, tanggal 9 Juli 2007
dan dinyatakan telah memenuhi syarat.

Anggota Tim Penguji

Tanda tangan

1. Hartono, S.Si
2. Mohtar Yuniarto, M.Si
3. Irwan Susanto, S.Si, DEA

- 1.
- 2.
- 3.

Surakarta,

Disahkan oleh
Fakultas MIPA
Dekan,

Ketua Program Studi DIII
Ilmu Komputer,

Prof. Drs. Sutarno, M.Sc. Ph.D
NIP. 131 649 948

Irwan Susanto, S.Si, DEA
NIP. 132 134 694

ABSTRAKSI

Dwi Wahyudi, 2007, **MEMBANGUN MAIL SERVER BERBASIS LINUX DENGAN MENGGUNAKAN POSTFIX**. Program D3 Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sebelas Maret.

Tujuan dari penelitian ini adalah untuk membangun sebuah sistem mail server yang bertugas untuk menangani lalu-lintas email dalam suatu sistem jaringan komputer yang di implementasikan pada platform Linux yang terkenal *free*, handal dan stabil dalam menangani sebagai fungsi server dengan menggunakan aplikasi MTA Postfix.

Dalam membangun mail server berbasis Linux menggunakan Postfix dilakukan pengkajian dan penelitian menggunakan metode studi literatur pada berbagai sumber baik melalui buku maupun *browsing* di internet. Selain itu juga di gunakan metode wawancara serta metode studi pustaka dengan OS Slackware Linux dan *software* Postfix, Apache, MySQL, BIND, IMAP dan Uebimiau.

Dari hasil Tugas Akhir dapat disimpulkan bahwa Postfix merupakan salah satu alternatif MTA berbasis Linux yang dirancang lebih cepat, lebih mudah dirawat dan lebih aman dalam menangani lalu-lintas penerimaan dan pengiriman email.

MOTTO

- *Read, think, write and do it*
- *Di dalam kesusahan pasti ada kemudahan*
- *Menangislah di masa muda maka kau akan tertawa di masa tua, tapi jika kau tertawa di masa muda maka kau akan menangis di masa tua*
- *Segalanya akan terasa berbeda, tergantung bagaimana kita menyikapinya.*

PERSEMBAHAN

Kupersembahkan untuk:

Ayahanda tercinta Samsi

Ibunda tercinta Mudrikah

Mbak tersayang Eni Rahmawati

KATA PENGANTAR

Bismillahirrohmanirrohim

Alhamdulillahirobbil'alamin. Segala puji syukur hanya untuk Allah SWT, *Rabb* seluruh alam semesta atas limpahan rahmat, nikmat dan karunia-Nya sehingga Penulis dapat menyelesaikan laporan Tugas Akhir ini sebagai prasyarat untuk mendapatkan gelar Ahli Madya pada Program Studi D3 Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sebelas Maret.

Pada laporan Tugas Akhir ini penulis mengambil tema tentang bagaimana membangun sebuah *mail server* sebagai sebuah *tool* yang bertugas untuk menangani berbagai proses yang berkaitan dengan email (*Electronic Mail*) dengan menggunakan aplikasi MTA (*Mail Transfer Agent*) Postfix berbasis Linux OS, dalam hal ini penulis mengimplmentasikannya pada distribusi Slackware 11 yang merupakan rilis terakhir dari distro ini.

Pada kesempatan kali ini Penulis tidak menempatkan secara spesifik objek analisa pembangunan mail server ini, yang artinya dalam proyek Tugas Akhir ini tidak ada objek khusus, dengan demikian implementasi dari sistem ini lebih fleksibel tanpa tergantung pada objek tertentu, dengan harapan siapa saja yang membaca laporan ini dapat mencermati dan memahami bagaimana membangun mail server menggunakan Postfix pada platform Linux terutama distribusi Slackware untuk kemudian dapat menerapkannya di mana saja.

Keberhasilan Penulis dalam menyelesaikan laporan ini tidak lepas dari bantuan dan dukungan dari semua pihak yang telah membantu Penulis dalam menyelesaikan laporan ini. Untuk itu pada kesempatan ini Penulis ingin menyampaikan rasa terima kasih dan penghargaan kepada:

1. Bapak Hartono, S.Si, selaku pembimbing I. Terima kasih untuk seluruh bimbingan, masukan, arahan dan dukungan yang di berikan kepada Penulis.
2. Bapak Mohtar Yunianto, M.Si, selaku pembimbing II. Terima kasih untuk bimbingan, arahan, masukan dan dukungan serta nasehat-nasehatnya yang telah di berikan kepada Penulis.

3. Bapak, Ibu dan Mbak tercinta, *my great hero in my live forever*. Terima kasih untuk cinta, kasih sayang, perhatian dan dukungan yang telah di berikan kepada Penulis.
4. Mas Bambang, terima kasih buat doa dan dukungannya.
5. Dek Uci dan Ibu nun jauh disana, terima kasih atas semua doa, dukungan, semangat dan kasih sayang yang telah di berikan kepada Penulis untuk lebih semangat menuju hidup yang lebih baik.
6. Seluruh keluargaku tercinta. Terima kasih atas dukungan dan kasih sayang telah di berikan kepada Penulis.
7. Seluruh teman satu kos, Sarwo, Gigih, Arya, Danang, dan Dedi. Terima kasih untuk persahabatan dan kasih sayang yang tulus. *You are my best friends*.
8. Terima kasih buat Mahendra dan Mas Andi, sahabat terbaikku, terima kasih untuk semua bantuan dan dukungannya.
9. Semua teman-teman Penulis di jurusan Teknik Komputer 2004. *Bravo Tkom'04*.
10. Seluruh teman-teman Asisten, mantan Asisten dan pengelola Laboratorium Komputasi FMIPA UNS, terima kasih untuk kasih sayang, persahabatan, masukan dan dukungan serta fasilitas yang diberikan kepada Penulis untuk menunjang penyelesaian laporan Tugas Akhir ini.
11. Semua pihak yang secara langsung maupun tidak langsung telah membantu Penulis dalam menyelesaikan laporan ini.

Penulis menyadari bahwa dalam pembuatan maupun penyampaian laporan ini, masih banyak kekurangan dan kesalahan, untuk itu saran dan masukan senantiasanya Penulis harapkan untuk perbaikan selanjutnya.

Surakarta, Juli 2007

Hormat kami

Penulis

DAFTAR ISI

	Halaman
JUDUL	i
HALAMAN PENGESAHAN.....	ii
ABSTRAKSI	iii
MOTTO	iv
PERSEMBAHAN.....	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	x
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN.....	xii
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	2
BAB II LANDASAN TEORI	
2.1 Pengertian dan Sejarah Linux	4
2.2 Linux Slackware	8
2.3 Pengertian Email.....	9
2.4 Email Client	10
2.5 Web Based Email.....	11
2.6 Mail Server.....	11
2.7 POP3 dan IMAP.....	12
2.8 Apache Web Server.....	12
2.9 Domain Name Server (DNS)	13
2.10 SMTP (<i>Simple Mail Transfer Protocol</i>)	16

2.11 MTA (<i>Mail Transfer Agent</i>) Postfix	17
2.12 MUA (<i>Mail User Agent</i>) UebiMiau.....	20
BAB III METODE PENELITIAN	
3.1 Prosedur Pelaksanaan Penelitian.....	22
3.1.1 Pengkajian Masalah	23
3.1.2 Implementasi dan Uji Coba Sistem.....	25
3.1.3 Dokumentasi	26
3.2 Alat dan Bahan	
3.2.1 Kebutuhan Alat	26
3.2.2 Kebutuhan Bahan	27
3.2.3 Kebutuhan Kualifikasi Administrator	27
BAB IV HASIL DAN PEMBAHASAN	
4.1 Perancangan Sistem	28
4.2 Hasil Penelitian	29
4.3 Pembahasan.....	29
4.3.1 Instalasi Linux Slackware	29
4.3.2 Konfigurasi Web Server (Apache Server)	41
4.3.3 Konfigurasi DNS Server (BIND 9)	49
4.3.4 Instalasi dan Konfigurasi Postfix	56
4.3.5 Instalasi dan Konfigurasi IMAP server (UW IMAP)	61
4.3.6 Instalasi dan Konfigurasi UebiMiau	62
4.3.7 Uji Coba	65
4.4 Pengujian Keseluruhan Sistem.....	66
BAB V PENUTUP	
5.1 Kesimpulan	67
5.2 Saran.....	68
DAFTAR PUSTAKA	69

DAFTAR TABEL

Tabel 4.1 Tabel partisi harddisk.....	31
---------------------------------------	----

DAFTAR GAMBAR

Gambar 2.1 Logo Linux (Tux).....	5
Gambar 2.2 Logo Linux Slackware	8
Gambar 2.3 Logo Apache	12
Gambar 2.4 Logo Postfix	17
Gambar 2.5 Logo UebiMiau	20
Gambar 3.1 Diagram pembangunan <i>mail server</i>	22
Gambar 3.2 Diagram alur analisis dan perancangan sistem <i>mail server</i>	23
Gambar 4.1 Tampilan awal instalasi Slackware 11	30
Gambar 4.2 Tampilan Login instalasi	30
Gambar 4.3 Program partisi <i>cfdisk</i>	31
Gambar 4.4 Program Instalasi Wizard Slackware 11	31
Gambar 4.5 Konfigurasi Partisi SWAP	32
Gambar 4.6 Penambahan Partisi root (/).....	32
Gambar 4.7 Pemilihan media sumber instalasi	33
Gambar 4.8 Pemilihan <i>Series Packages</i> Instalasi	34
Gambar 4.9 Pemilihan Mode Instalasi	34
Gambar 4.10 Pemilihan Paket Instalasi	35
Gambar 4.11 Pemilihan media sumber instalasi Kernel	35
Gambar 4.12 Pemilihan Kernel Linux	36
Gambar 4.13 Konfigurasi mode instalasi LILO.....	36
Gambar 4.14 Konfigurasi <i>Frame Buffer Console</i>	37
Gambar 4.15 Konfigurasi LILO <i>Destination</i>	37
Gambar 4.16 Konfigurasi <i>Mouse</i>	38
Gambar 4.17 Konfigurasi <i>Network Setup</i>	39
Gambar 4.18 Konfigurasi <i>Startup Service To Run</i>	39
Gambar 4.19 Konfigurasi <i>Password root</i>	40
Gambar 4.20 Tampilan LILO	40
Gambar 4.21 Tampilan halaman login UebiMiau.....	65
Gambar 4.22 Tampilan halaman utama UebiMiau	65

BAB I

PENDAHULUAN

1.1 Latar Belakang

Electronic mail atau yang akrab disebut email, merupakan istilah populer untuk pesan atau surat elektronik, biasanya berbentuk pesan teks yang ditulis oleh seseorang (user) melalui sebuah sistem komputer dan ditransmisikan ke komputer lain yang dituju dengan melintasi jaringan komputer.

Sistem email dapat dibentuk oleh dua atau lebih sistem komputer yang dihubungkan via jaringan. Originalnya sistem email ini hanya dapat mengirim dan menerima pesan email secara lokal diantara orang-orang yang dilengkapi software *email* pada sebuah kantor atau jaringan. Akan tetapi dengan adanya ekspansi Internet, vendor mulai mengaplikasikan sistem email secara luas melalui kapabilitas koneksi Internet. Dengan demikian orang-orang tidak hanya dapat berkirim email secara lokal, melainkan dapat secara global, kapan pun dan dari mana pun mereka berada. Penghantaran *email* melintasi jaringan global Internet akan melibatkan berbagai software yang bertugas mengkonversi pesan ke bentuk standar yang bisa ditransmisikan via email.

Beberapa keuntungan sistem *mail electronic* atau email bagi penggunanya, diantaranya lebih ekonomis, lebih simpel, sangat cepat, mudah dikelola, dan mampu mentransmisi berbagai format dokumen. Berbagai peran yang dapat dilakukan oleh sistem email tersebut tentunya tidak lepas dari peran *software* yang mampu menangani proses pengiriman maupun penerimaan sebuah pesan dari atau ke seseorang (user) yaitu mail server.

Dalam lingkungan UNIX dan turunannya, banyak *software mailer* yang dapat difungsikan sebagai basis server email, salah satunya adalah Postfix yang merupakan *software free* atau gratis. Postfix merupakan *software mailer* yang dirancang lebih cepat, lebih mudah dirawat dan lebih aman. Hal ini yang mendasari pembuatan sistem mail server berbasis LINUX sebagai salah satu solusi untuk menuhi kebutuhan manusia dalam hal komunikasi yang mudah, cepat, tepat, akurat, dan ekonomis.

1.2 Rumusan Masalah

Bagaimana membangun sistem *mail server* berbasis LINUX yang murah, mudah dan handal dengan menggunakan software *mailer* Postfix?

1.3 Batasan Masalah

Untuk mempermudah dalam melakukan penelitian ini, maka beberapa pembatasan pokok bahasan penelitian perlu dilakukan, diantaranya:

1. Penelitian hanya terbatas pada software MTA (*Mail Transfer Agent*) Postfix.
2. Sistem ini hanya diimplementasikan pada platform LINUX khususnya distro Slackware.
3. Penelitian hanya terbatas pada pembangunan *mail server stand alone* dalam jaringan lokal area.
4. Penelitian hanya membahas pada *mail client* berbasis web.
5. Dalam sistem ini hanya mengimplementasikan Uebimiau sebagai aplikasi *webmail* tanpa menjelaskan secara mendetail tentang sistem kerja webmail tersebut.

1.4 Tujuan Penelitian

Tujuan dari pembuatan sistem ini adalah untuk menghasilkan sebuah sistem email yang *free* berbasis LINUX dengan kemampuan yang handal dalam usaha pemenuhan kebutuhan manusia terutama dalam hal komunikasi yang mudah, cepat, tepat, akurat, dan ekonomis.

1.5 Manfaat Penelitian

1.5.1 Umum

Manfaat yang didapat terkait dengan penelitian yang dilakukan adalah memberikan salah satu alternatif dalam pembangunan sebuah *mail server* dengan biaya murah atau gratis dan pengelolaan yang mudah tanpa mengurangi kemampuan atau kehandalan dari sistem email tersebut.

1.5.2 Khusus

Mahasiswa

1. Sebagai salah satu sarana untuk menerapkan dan menguji kemampuan mahasiswa sebagai hasil studi selama tiga tahun di jenjang D3 Ilmu Komputer FMIPA UNS.
2. Sebagai salah satu referensi belajar bagi mahasiswa untuk meningkatkan pemahamannya khususnya dalam bidang server Linux.
3. Sebagai salah satu acuan bagi mahasiswa yang lain dalam pelaksanaan penelitian tugas akhir dengan tema yang sama.

Kampus

1. Sebagai sarana untuk menguji kemampuan mahasiswa dalam bidang IT sebagai langkah awal untuk menuju dunia kerja, sehingga setiap lulusan D3 Ilmu Komputer FMIPA UNS memiliki kompetensi untuk dapat bersaing di dunia kerja.
2. Sebagai bahan evaluasi bagi pihak Fakultas terutama Prodi D3 Ilmu Komputer FMIPA UNS dalam menyelenggarakan sistem pendidikan di UNS untuk lebih meningkatkan mutu sistem pendidikan yang ada, sehingga dapat dihasilkan lulusan yang kompeten di bidangnya dalam menghadapi tantangan dunia kerja.

BAB II

LANDASAN TEORI

2.1 Pengertian dan Sejarah Linux

Linux adalah sebuah program *open source* yang gratis di bawah lisensi GNU, yang merupakan turunan dari Unix dan dapat dijalankan pada berbagai macam *platform* perangkat keras mulai dari Intel (x86), hingga prosesor RISC. Linux sebagai program *open source* yang gratis, adalah salah satu yang membuat Linux bisa berkembang dengan pesat dan banyak pengguna komputer yang beralih ke sistem operasi berlogo penguin ini.

Linux adalah sistem operasi yang disebarluaskan secara luas dengan bebas dibawah lisensi GNU (*GNU Not Unix*), yang berarti *source code* Linux juga tersedia. Itulah yang membuat Linux sangat spesial dan mampu berkembang dengan pesat dan banyak pengguna komputer yang beralih ke sistem operasi berlogo *Tux* (penguin) ini. Dengan demikian, dapat dikatakan bahwa Anda dapat memperoleh Linux tanpa harus membayar sama sekali. Jika Anda harus membayar tiap kali instal perangkat lunak di lain komputer, maka dengan Linux Anda dapat menginstalnya dimana saja tanpa harus membayar lisensi.

Kebebasan yang paling penting dari Linux, terutama bagi *programmer* dan administrator jaringan, adalah kebebasan memperoleh kode sumber (*source code*) dan kebebasan untuk mengubahnya. Ini berimplikasi pada beberapa hal penting yaitu keamanan dan dinamika.

Keterbukaan kode sumber memungkinkan sistem operasi berkembang dengan pesat. Jika sebuah program dengan sistem tertutup dan hanya dikembangkan oleh vendor tertentu, paling banyak sekitar seribu hingga lima ribu orang. Sedangkan Linux, dengan keterbukaan kode sumbernya, dikembangkan oleh sukarelawan seluruh dunia. *Bug* lebih cepat diketahui dan program penambalnya (*patch*) lebih cepat tersedia. Pendekatan pengembangan sistem operasi ini disebut *Bazaar*. Kebalikannya sistem *Chatedraal* sangat tertutup dan hanya berpusat pada satu atau dua pengembang saja.

Linux dulunya adalah proyek hobi yang dikerjakan oleh Linus Torvalds yang memperoleh inspirasi dari Minix. Minix adalah sistem Unix kecil yang dikembangkan oleh Andrew S. Tanenbaum, seorang profesor yang menggeluti penelitian masalah OS dari Vrije Universiteit, Belanda.

Di Suonen Tasavalta, Republik Finlandia, seorang mahasiswa bernama Linus Torvalds mengikuti mata kuliah unix dan bahasa pemrograman C. Saat itu Linus menggunakan sistem operasi mini berbasis unix bernama Minix. Dia merasa Minix mempunyai banyak kelemahan, dan ia berkeyakinan mampu membuat lebih baik dari itu. Di usia 23, Linus mulai mengotak-atik kernel Minix dan menjalankannya di mesin Intel x86.

Linux versi 0.01 dikerjakan sekitar bulan Agustus 1991. Kemudian pada tanggal 5 Oktober 1991, Linus mengumumkan versi resmi Linux 0.02 yang hanya bisa menjalankan *bash (GNU Bourne Again Shell)* dan *gcc (GNU C Compiler)*. Sekarang Linux adalah sistem UNIX yang lengkap, bisa digunakan untuk jaringan (networking), pengembangan software, dan bahkan untuk keperluan sehari-hari. Linux sekarang merupakan alternatif OS yang jauh lebih murah jika dibandingkan dengan OS komersial, dengan kemampuan yang setara atau bahkan lebih.

Linux menggunakan logo bergambar penguin. Awalnya logo Linux di kompetisikan kepada umum lewat diskusi pada mailing list Linux Kernel. Pencetusnya adalah Alan Cox, dan logo terpilih diberi nama Tux, dibuat oleh Larry Ewing.



Gambar 2.1 Logo Linux (Tux)

Sebagai tambahan, Linux menyediakan bahasa pemrograman gratis, lengkap dengan kompilernya, maupun program pembantunya. Beberapa diantaranya adalah :

- a) ADA
- b) BASIC
- c) C
- d) C++
- e) Expect
- f) FORTRAN
- g) GTK, untuk membuat aplikasi GUI di Linux
- h) PASCAL
- i) Python
- j) Skrip Shell
- k) TCL
- l) Perl (*The Practical Extraction and Report Language*), sering dipakai untuk membuat skrip CGI di web.

Beberapa fitur Linux yang patut dicatat diantaranya :

- a) *Multi tasking* dan dukungan 32 bit; mampu menjalankan beberapa perintah secara bersamaan, dan dengan memanfaatkan model terlindung (*protected mode*) dari Intel 80836 keatas, Linux merupakan sistem operasi 32 bit.
- b) *Multi user dan Multi session*; Linux dapat melayani beberapa user yang login secara bersamaan. Sistem filenya sendiri mempunyai keamanan yang ketat, dan dapat dimodifikasi secara optimal untuk akses file kepada user atau group tertentu saja. - Sebagian besar Linux ditulis dalam bahasa C
- c) Dukungan Java; jika dikompilasi pada level kernel, Linux dapat menjalankan Java Applet sebagai aplikasi.
- d) *Virtual Memory*. Linux menggunakan sebagian dari hardisk Anda dan memperlakukannya sebagai memory, sehingga meningkatkan memory Anda yang sebenarnya.
- e) Linux menawarkan sistem file yang hierarkis, dengan beberapa folder utama yang sudah dibakukan (*File System Standard/FSSTND*)

- f) Grafis antar muka pemakai (*Graphical User Interface/GUI*) yang dipergunakan Linux adalah sistem X Window atau X dari MIT.

Sebagai *standalone* PC, atau komputer pribadi yang berdiri sendiri, ditemukan bahwa Linux adalah sistem operasi yang dapat diandalkan. Tidak ada lagi gangguan virus, macet ditengah jalan, atau *reboot* puluhan kali. Apalagi jika komputer Anda terhubung ke Internet, Linux menjanjikan keamanan yang cukup memadai. Beberapa program untuk aplikasi sehari-hari yang sudah berjalan di Linux diantaranya adalah :

- a) Program Database Case, CodeBase, DBMS, Informix, LEAP, Postgresql, MySql.
- b) Program Bisnis Star Office, Applixware, Koffice, Xspreadsheet, Abiword, WordPerfect 8, TEX.
- c) Aplikasi Grafis Blender, Gimp, ME10, Megahedron, OCRShop, PostShop, ScanShip, Sketch, VariCAD
- d) Aplikasi Jaringan SAINT, Nmap, MRTG, Tripwire, SSH (*Secure Shell*), VisualRoute, SQUID

Di Linux, semua program (yang berada dibawah lisensi GNU) dapat diperoleh secara cuma-cuma dan legal dengan men-*download*-nya dari internet. Hampir setiap hari bermunculan program-program baru yang dikembangkan oleh para sukarelawan maupun organisasi non profit diseluruh dunia.

Sebagian besar program di Linux berukuran relatif kecil, terutama yang berbasis *console* (non GUI). Yang perlu Anda lakukan hanyalah mengambilnya dari sebuah situs (*download*), dan instal di komputer Anda.

Selain tangguh sebagai komputer pribadi, Linux menawarkan kinerja optimal untuk dijadikan sebagai server. Beberapa aplikasi server yang selalu disertakan hampir di setiap distribusi diantaranya :

- a) Web Server (httpd)
- b) FTP Server (ftpd)
- c) Mail Server (smtp, pop 3, LDAP, IMAP)

- d) Name Server
- e) Daemon standar (telnetd, fingerd, identd, syslogd dsb)
- f) DHCP Server

Selain itu, protokol-protokol standar sebagai platform dalam komunikasi jaringan, telah terintegrasi pada level kernel, di antaranya :

- a) File Transfer Protocol (FTP)
- b) Gopher Protocol
- c) Hypertext Transfer Protocol (HTTP)
- d) Post Office Protocol (POP)
- e) Point to Point Protocol (PPP)
- f) Serial Line Internet Protocol (PLIP)
- g) Simple Mail Transfer Protocol (SMTP)
- h) Telnet Protocol
- i) Transmission Control Protocol/Internet Protocol (TCP/IP)

2.2 Linux Slackware

Menurut Slackware.com (2005), Slackware, Inc. adalah perusahaan veteran yang merupakan perusahaan pertama pertama yang berhasil memperkenalkan distro Linux kepada masyarakat. Slackware, Inc. dibangun oleh Patrick J. Volkerding pada akhir 1992. sebelumnya, Volkerding mulai mengenal Linux ketika Ia membutuhkan penafsir LISP yang murah untuk sebuah proyek. Pada saat itu masih sedikit sekali distro, sehingga Ia menggunakan distro *Soft Landing System* (SLS linux).



Gambar 2.2 Logo Linux Slackware

Ketika SLS mengalami beberapa permasalahan, Volkerding mulai memperbaharui permasalahan-permasalahan yang Ia temukan. Segera setelah hal itu, ia memutuskan untuk menggabungkan pembaharuan-pembaharuan tersebut kedalam sebuah distro untuk dirinya dan teman-teman dekatnya. Distro ini sangat cepat sekali perkembangannya, sehingga Volkerding menjadikannya tersedia untuk umum dengan nama Slackware.

Sepanjang perjalanannya, Volkerding menambahkan beberapa program baru seperti program instalasi yang akrab pengguna berbasis sistem menu dan kosep manajemen paket-paketnya. Hal ini mempermudah untuk menambahkan, menghapus, atau menatar paket perangkat lunak dari sistem mereka masing-masing.

2.3 Pengertian Email

Email atau *Electronic Mail* adalah layanan yang memudahkan user untuk saling bertukar pesan. Tiap user email memiliki kotak surat (*mailbox*) yang digunakan untuk menyimpan email yang ditujukan ke user tersebut. Salah satu keuntungan email adalah kemampuannya menghantarkan pesan kepada user lain dalam waktu yang singkat bahkan hanya dalam hitungan detik meskipun berada pada lokasi yang jauh.

Sejak kali pertama diperkenalkan oleh seorang ilmuwan BBN Technologies, **Ray Tomlinson**, lebih dari tiga puluhan tahun yang lalu, teknologi *electronic mail* atau e-mail telah memprakarsai sebuah revolusi besar dalam sejarah komunikasi manusia. Kemudahan dalam mengaksesnya, kecepatan yang bisa diandalkan, serta hemat biaya, membuat surat-surat elektronik ini mampu mempengaruhi perilaku manusia dalam berkomunikasi. Tidak mengherankan jika dalam perkembangannya, aktivitas penggunaan email menjadi sangat populer di kalangan pengguna teknologi jaringan komputer mulai dari *Local Area Network* (LAN), *Wide Area Network* (WAN) sampai dengan Internet. Mulai sekadar digunakan untuk berkomunikasi antarpribadi sampai urusan pekerjaan, bisnis, ataupun lainnya.

Kemampuan utama sistem email adalah untuk mengirimkan pesan elektronik ke alamat email seseorang. File Non-ASCII (*file binary*) dapat dikirim sebagai lampiran (*attachment*) sebuah pesan email. File ini di sebut dengan file *attachment* MIME (*Multimedia Internet Mail Extention*). MIME di kembangkan untuk membantu software email mengatasi berbagai format file. Sebagai contoh, file yang dibuat dengan Microsoft Word dapat di *attach* ke dalam pesan email dan dapat diambil oleh user penerima dengan program email yang sesuai.

Beberapa keuntungan penggunaan pesan elektronik atau email, diantaranya:

- a) Pesan dapat dikirim setiap saat, melintasi wilayah yang tak terbatas.
- b) Sebuah pesan dapat dikirmkan ke seseorang penerima atau multi penerima sekaligus tanpa harus mengeluarkan biaya ekstra.
- c) Dalam kondisi normal, pesan dapat disampaikan dengan sangat singkat (dalam hitungan detik).
- d) Transmisi pesan dapat di-log, meyakinkan berbagai informasi tentang pesan dapat dicatat, misalnya dari mana pesan datang, kapan pesan tiba, melintasi rute mana pesan dihantarkan, dan sebagainya.
- e) Penerima dapat membaca dan mengambil pesan kapanpun menginginkan dari manapun lokasi sepanjang terkoneksi dengan jaringan (Internet).
- f) User mobile juga dapat melakukan hal serupa, mengakses arsip mail mereka dan berbagai lokasi dan kapan pun mengijinkan.
- g) Tidak membutuhkan mailing konvensional, seperti kertas, amplop, jasa pos, atau lainnya, selain spasi disk dan akses ke jaringan.

2.4 Email Client

Email Client adalah software aplikasi yang digunakan user untuk membaca, menulis dan mengirim email. Singkatnya, email client adalah software yang memberikan antarmuka langsung antara user dengan sistem email.

Program email client biasanya memberikan beragam fasilitas, seperti *text editor*, *attactment address book*, *filing cabinet*, dan modul komunikasi.

Ada banyak *software email client* tersedia yang bisa kita install untuk mengelola email. Beberapa yang populer diantaranya MS Outlook, Outlook Express, Netscap Mail, Eudora, Aol. Masing-masing menawarkan fasilitas yang beragam. Meski begitu fasilitas yang telah disebutkan diatas merupakan fitur standar dan dimiliki oleh hampir semua *email client*.

2.5 Web Based Email

Bentuk lain dari *email client* adalah *Web Based Email*, atau email berbasis Web. Jika *software email client* normal diinstal langsung pada komputer lokal yang terkoneksi jaringan, Web Based Email menggunakan browser Web sebagai jalan bagi user untuk mengelola email.

Fitur atau element Web Based Email tidak jauh berbeda dengan *software email client* normal, diantaranya kita bisa menulis, menghapus, atau mengirim pesan. Kita juga bisa mengirim file dalam *attachment*, atau mengoleksi *address user* lain dalam ruang yang tersedia.

Beberapa contoh situs yang mengimplementasikan Web Based Email adalah *mail.yahoo.com*, *hotmail.com*, *mail.com*, *mail.google.com*, *plasa.com*, dan lain sebagainya. Fungsi – fungsi sistem email ditangani secara sentral dalam server mereka, sedang komputer yang digunakan user hanya sebagai sarana untuk mengakses. Dibutuhkan sebuah user ID dan password untuk dapat melakukan akses ke sistem email.

2.6 Mail Server

Mail server, atau email server adalah aplikasi yang menangani penghantaran pesan email. Mesin ini senantiasa menerima pesan dari *email client* yang digunakan user, atau mungkin dari server email lainnya. Sesuai dengan namanya, server email adalah pusat kendali sistem email.

Sebuah mail server biasanya terdiri dari area penyimpanan, set konfigurasi user, daftar user dan seri modul komunikasi.

Mesin server email biasanya dirawat oleh seorang yang biasa dipanggil Postmaster. Salah satu tugas postmaster adalah mengelola *account user* yang

berhak berkirim email, memonitor operasi server, dan berbagai tugas administratif lainnya. Meski begitu, kebanyakan server email dirancang untuk beroperasi tanpa banyak intervensi manual. Mereka menunggu pesan dari mail client untuk diteruskan ke tujuan lain, memprosesnya sesuai dengan yang digariskan, atau menerima pesan dari server email pada interval tertentu.

2.7 POP3 dan IMAP

Pada umumnya ketika pertama kali mensetup layanan email, user diberi opsi apakah ingin menggunakan POP3 atau IMAP4. Kedua istilah ini menjelaskan metode bagaimana pesan email ingin ditangani.

Sederhananya dengan memilih POP3, maka program komputer secara aktual akan mendownload pesan email dari server email ke komputer lokal. Sebaliknya, kita bisa mengirim pesan email ke server email dari komputer lokal untuk disampikan ke tujuan tertentu.

Adapun dengan IMAP4, maka kita akan menyimpan pesan email sepenuhnya dalam server email dan menggunakan komputer lokal untuk mengirim dan mengambilnya kapanpun diinginkan, disini user cukup men-*scan* subjek pesan dan memutuskan apakah perlu membacanya, mendownload file, atau menghapusnya.

2.8 Apache Web Server



Gambar 2.3 Logo Apache

Apache [<http://www.apache.org>] merupakan web server yang terpopuler saat ini. Menurut survey *netcraft* lebih dari 50% situs di Internet menggunakan Apache sebagai Web server. Server Apache sangat luwes, memenuhi standar HTTP/1.1, mengimplementasikan protokol terbaru dan sangat luwes dikonfigurasi serta dapat ditambahkan modul lainnya melalui modul Apache. Tersedia untuk berbagai sistem operasi.

Apache menyediakan fasilitas yang kaya, yang sangat dibutuhkan suatu server serius, seperti otentikasi, pengaturan akses direktori, *virtual host*, kemampuan URL *rewriting*, dan juga alias. Kemampuan melakukan *content negotiation* membuat Apache mampu melayani beragam client secara otomatis, baik untuk berbagai *browser* yang memiliki kemampuan berbeda ataupun untuk *device* akses yang berbeda. Fungsi *log* yang dimiliki oleh Apache dapat dikirim melalui proses *piping*, sehingga dapat dilakukan rotasi log, filter log, serta melakukan pemisahan log secara langsung.

Awalnya Apache dikembangkan berdasarkan keinginan untuk memperbaiki Web server yang saat itu populer (NCSA web server). Tetapi akhirnya mengalami perombakan dan penulisan ulang dan menjadi Web server yang berdiri sendiri dan berbeda dengan NCSA. Kini malah mengalahkan kepopuleran NCSA Web server. Pada tahun 1999 dibentuk Apache Software Foundation untuk mengurus perkembangan Apache ini. Apache telah membuktikan sebagai web server yang cepat, stabil dengan fitur yang paling kaya di antara web server lainnya. Saat ini proyek Apache telah berkembang dan tidak hanya sekedar Web server.

2.9 Domain Name Server (DNS)

Bagi komputer, dalam berkomunikasi antar komputer dengan komputer sudah cukup menggunakan alamat IP, namun bagi manusia, komunikasi antara manusia dengan komputer tidak cukup hanya dengan IP saja karena manusia kesulitan dalam mengingat IP sehingga diperlukan sebuah nama untuk memudahkan mengingat IP atau yang biasa disebut dengan DNS (*Domain Name Server*). Contohnya : 167.205.64.155 dengan www.divkom.ee.itb.ac.id.

DNS adalah kependekan dari Domain Name Server. DNS merupakan service yang memetakan nama host ke IP address atau sebaliknya sehingga orang tidak perlu mengingat IP tetapi tinggal menggunakan nama saja. Sejarahnya pengaturan nomor IP dan nama host diatur secara tersentral oleh IANA 1 <http://www.iana.org> yang dimotori oleh Jon Postel (<http://www.postel.org>).

Pada awalnya, Daftar tabel sentral di download secara berkala. Sistem label sentral ini hanya bisa digunakan untuk jumlah mesin yang tidak terlalu

banyak. Perkembangan Internet yang semakin cepat mengakibatkan jumlah host semakin bertambah banyak sehingga membuat tabel tersentral menjadi sangat besar dan susah untuk dikelola. Oleh karena itu, dikembangkanlah tabel yang bersifat terdistribusi. Dengan pendistribusian ini, masing - masing organisasi bertanggung jawab atas database yang berisi informasi mengenai jaringannya sendiri. Misalnya DNS server UNS hanya bertanggung jawab atas domain *uns.ac.id*. Informasi yang disimpan dalam DNS server berupa alamat IP dan *namahost*. Oleh karena itu, format penamaan host harus konsisten untuk semua host. Format penamaan host di internet dibuat membentuk hierarki. Skema hierarki tersebut membentuk tree. Satu titik atau node membentuk tree memiliki beberapa subnode, dan setiap subnode membentuk beberapa tree yang memiliki beberapa subnode. Setiap node memiliki label yang disebut domain. Domain ini bisa berupa *namahost*, subdomain, atau top level domain.

Pada saat suatu host di dalam suatu jaringan terhubung ke jaringan lain melalui nama host, disebut juga *fully qualified domain name* (SQDN), DNS digunakan untuk mengetahui IP address dari host tersebut.

Berikut adalah namadomain di dunia yang terbagi dalam beberapa domain:

1. gTLD (*generic Top Level Domain*)

a) Domain yang populer:

.com : Organisasi Komersial.

.org : Organisasi no profit

.edu : Institusi Pendidikan atau Universitas.

.net : Network

.gov : Organisasi pemerintah

.mil : Organisasi militer

.int

b) Tambahan domain baru

.aero

.biz

.coop

.info

.museum

.name

.pro

2. ccTLD (*country code Top Level Domain*)

.ID, .JP, .SG, .US, dll.

Masing-masing dikelola oleh seorang administrator yang ditunjuk oleh IANA Untuk Indonesia dikelola oleh IDNIC 2 (<http://www.idnic.net.id>) susunannya adalah sebagai berikut :

Domain harus di bawah second level domain. Strukturnya adalah sebagai berikut :

.ac.id : academic

.co.id : company

.or.id : others

.net.id : net

.go.id : government

.mil.id : military

.sch.id : school

.web.id : web

Khusus:

.war.net.id : warnet

Domain teratas ialah root domain. Domain ini ditulis dalam bentuk titik. Semua node yang berada tepat dibawah root domain adalah top level domain. Node yang berada tepat dibawah *top level domain* disebut *seconde level domain* dan semua node yang berada di bawah second level domain disebut *third level domain*.

Sebagai contoh:

web.master.design.com

Tanda . menunjuk root domain

com menunjukkan top level domain

design menunjukkan second level domain

master menunjukkan third level domain

web menunjukkan nama host datau komputer

Setiap jaringan yang terhubung dengan Internet di pastikan memiliki nama domain (*domain name*) yang di asosiasikan dengannya. Ini di gunakan untuk meyakinkan email atau trafik lainnya – dapat tersampaikan ke alamat yang tepat. Sebagai contoh, jika sebuah ISP bernama ”Firdaus” dan domain yang mereka miliki bernama “Firdaus.com”, maka semua email untuk seorang user akan diarahkan ke ”user@firdaus.com”, yang aktualnya terdaftar dalam mail server ISP tersebut.

Dalam konsteks diagram jaringan lokal, domain dapat di presentasikan sebagai “Local.Office.A”, ”Local.Office.B” atau “Solo.Office”. Hal ini untuk memudahkan *addressing user* dalam sebuah departemen, atau diantara departemen. Domain-domain lokal ini tentunya tidak bisa digunakan di Internet. Mereka tidak bisa mencapai atau di capai user internet. Domain lokal dapat diperluas dengan mudah untuk bisa menjangkau Internet. Misalnya “Solo.Office” diregistrasikan menjadi “solo.office.perusahaan.com”. Dengan demikian domain ini dapat digunakan untuk *addressing email* Internet user karena sudah berupa *address internet legal* yang sudah teregistrasi.

2.9 SMTP (*Simple Mail Transfer Protocol*)

Simple Mail Transfer Protocol adalah protokol standar untuk pengiriman *electronic mail* (email) di internet. SMTP adalah protokol yang cukup sederhana, berbasis teks dimana protokol ini menyebutkan satu atau lebih penerima email untuk kemudian diverifikasi. Jika penerima email valid, maka email akan segera dikirim. SMTP menggunakan port 25 dan dapat dihubungi melalui program telnet. Agar dapat menggunakan SMTP server lewat nama domain, maka record DNS (*Domain Name Server*) pada bagian MX (*Mail Exchange*) digunakan.

Karena protokol SMTP berawal dari protokol yang benar-benar berbasis teks ASCII, maka SMTP tidak bekerja terlalu baik dalam mengirimkan file-file

binary. Standar untuk meng-*encode* file-file biner agar dapat dikirimkan lewat SMTP dikembangkan dan menelurkan standar-standar seperti MIME (*Multipurposes Internet Mail Extensions*). Saat ini, hampir semua SMTP server mendukung 8BITMIME, yang dapat mengirimkan file-file biner semudah mengirimkan file teks.

SMTP hanya protokol yang melakukan “push”, artinya dia hanya bisa mengambil email dari client tetapi tidak bisa melakukan “pull”, yaitu melayani pengambilan email di server oleh client. Pengambilan pesan atau email tersebut dilakukan dengan menggunakan protokol tersendiri yaitu protokol POP3 (*Post Office Protokol*) atau IMAP (*Internet Message Access Protocol*).

2.9 MTA (*Mail Transfer Agent*) Postfix

Postfix ditulis oleh Wietse Venema (<http://www.porcupino.org/wietse/>), dan termasuk salah satu proyek *freeware*. Penyandang gelar Ph.D. dalam bidang fisika ini juga membuat program dan tools-tools yang lain, misalnya SATAN dan TCP Wrapper. Proyek mulai dikerjakan Wietse saat berkunjung ke IBM T.J. Watson Research. Wietse diberi kesempatan oleh IBM untuk menulis software ini. Originalnya software tersebut diberi nama Vmailer, namun karena alasan kemiripan dengan merek dagang yang telah ada, diganti menjadi Postfix atas saran IBM.

Postfix berusaha memberikan alternatif untuk program Sendmail yang telah luas memasyarakat. Postfix dirancang lebih cepat, lebih mudah dirawat, dan lebih aman. Isu keamanan yang sempat mengguncang Sendmail pada rilis-rilis terdahulu memberi inspirasi bagi Wietse (dan programmer-programer *mailer* lainnya) untuk menulis software mail yang menekankan sekuritas.



Gambar 2.4 Logo Postfix

Dalam sisi keamanan dan modularitas postfix benar-benar memosisikan diri sebagai rival mail server buatan D. J. Bernstein. Keduanya sangat mengutamakan keamanan dan modularitas. Meskipun dari berbagai benchmark banyak orang berpendapat Postfix lebih cepat.

Sebagaimana dituturkan dalam *official site*-nya, sasaran utama proyek Postfix adalah untuk mengimplementasikan alternatif yang reliabel atas program Sendmail yang telah luas digunakan dalam situs-situs UNIX.

Adapun secara lebih spesifik, Postfix berusaha mencapai sasaran berikut:

- a) Penyebaran yang luas. Postfix harus bisa diadopsi oleh masyarakat luas dalam rangka melayani impact signifikan atas performansi dan security mail Internet. Oleh karena itu software Postfix disediakan secara cuma-cuma.
- b) Performansi. Postfix diakui tiga kali lebih cepat dibanding kompetitor utamanya. Sebuah PC desktop yang menjalankan Postfix dapat menerima dan menghantarkan jutaan pesan berbeda per hari. Postfix memanfaatkan trik-trik web server guna mereduksi overhead pembuatan proses dan menggunakan trik-trik lainnya untuk mereduksi overhead sistem, tanpa harus mengganggu realibilitasnya.
- c) Kompatibilitas. Postfix dirancang sebagai “sendmail-compatible” guna memudahkan orang-orang melakukan migrasi. Postfix diantaranya mensupport file-file `/var/[spool]/mail`, `/etc/aliases`, NIS, dan `~/forward`. Meski demikian Postfix juga berusaha untuk mudah dirawat dan diadministrasi. Oleh karenanya Postfix tidak menggunakan `sendmail.cf`.
- d) Ketahanan. Postfix juga dirancang untuk bertindak rasional dibawah tekanan atau beban berat. Saat sistem lokal mengkonsumsi habis spasi disk atau memori, software Postfix memilih menahan diri daripada memperburuk masalah. Postfix berjalan dibawah kendali yang bisa dikontrol secara manual.
- e) Fleksibilitas. Postfix dibangun dari lusinan program kecil yang masing-masing hanya membentuk satu tugas spesifik: menerima pesan via SMTP, menghantarkan pesan via SMTP, menghantarkan pesan secara lokal,

menulis ulang (*rewrite*) sebuah address, dan lain-lain. Site-site dengan tuntutan yang spesifik dapat menganti satu atau lebih program kecil ini dengan versi alternatif lainnya. Postfix juga memberikan kemudahan dalam menutup sebuah fungsionalitas tertentu, seperti firewall dan atau mengontrol *workstation* klien yang tidak membutuhkan penghantaran lokal sama sekali.

- f) *Security*. Postfix menggunakan pertahanan *multilayer* untuk melindungi sistem lokal dari gangguan *intruder*. Kebanyakan daemon Postfix dapat berjalan dalam area yang disebut *chroot jail*, dengan pemberian priveledge yang rendah. Tidak aada path langsung dari jaringan ke program penghantaran lokal yang memiliki *security* sensitif. Seorang intruder harus mendobrak beberapa program lain terlebih dahulu. Postfix sendiri tidak “mempercapai” *content file-file* antrian itu, atau content pesan-pesan IPC-nya. Postfix terlebih dahulu akan memfilter informasi yang dikirim pengirim (*sender*) sebelum mengeksponnya via variabel-variabel *environment*. Dan yang terakhir, program Postfix juga bukan tergolong *set-uid*. Jadi cukup aman untuk dioperasikan.

Beberapa fitur unggulan yang dimiliki oleh Postfix sebagai sebuah sistem email yang handal:

a) *Multitransport*

Postfix dirancang cukup fleksibel dimana ia dapat beroperasi dalam beragam *environment*, seperti internet, DECnet, X 400, dan UUCP, tanpa membutuhkan *domain virtual*. Meski begitu rilis awal Postfix diakui memang hanya dapat berkomunikasi dengan SMTP dan terbatas untuk UUCP.

b) *Domain Virtual*

Menambahkan *domain virtual* pada Postfix cukup mudah dimana kita hanya perlu mengubah tabel lookup tunggal, sedangkan mailer lainnya pada umumnya memerlukan multilevel aliasing atau redireksi untuk memperoleh hasil yang sama.

c) *Retriksi Relay*

Postfix memberi jalan bagi kita untuk merestriksi *host*, nama yang dapat merelay mail melalui sistem Postfix, dan mail mana yang diijinkan masuk. Untuk kebutuhan ini Postfix mengimplementasikan operasi *blacklist*, *RBL lookups*, *HELO/sender DNS lookups*.

d) *Table Lookups*

Postfix tidak mengimplementasikan bahasa address rewriting, melainkan mempekerjakan apa yang disebut tabel *lookups*. Tabel – tabel dapat berupa dbm lokal atau file –file db, atau peta-peta NIS atau NetInfo jaringan. Penambahan dukungan untuk mekanisme *lookup* lainnya juga cukup mudah.

2.10 MUA (*Mail User Agent*) UebiMiau



Gambar 2.5 Logo UebiMiau

Uebimiau merupakan salah satu aplikasi yang menghubungkan antara *user* dengan sistem *mail server* berbasis web (*web based*) yang ditulis dan dikembangkan oleh Aldoir Ventura menggunakan bahasa pemrograman PHP. UebiMiau termasuk salah satu proyek *free software* di bawah lisensi GNU yang *multiplatform*, artinya dapat diimplementasikan pada berbagai macam sistem operasi, baik Windows maupun Linux, serta dapat digunakan untuk semua *mail server*. Beberapa fitur yang dimiliki UebiMiau diantaranya:

- a. SMTP Compatible
- b. POP3 Compatible
- c. IMAP Compatible
- d. MIME Compatible
- e. Receive Attachments
- f. Send Attachments
- g. Folders/E-mail management support (Trash/Inbox/Sent/[Personal])

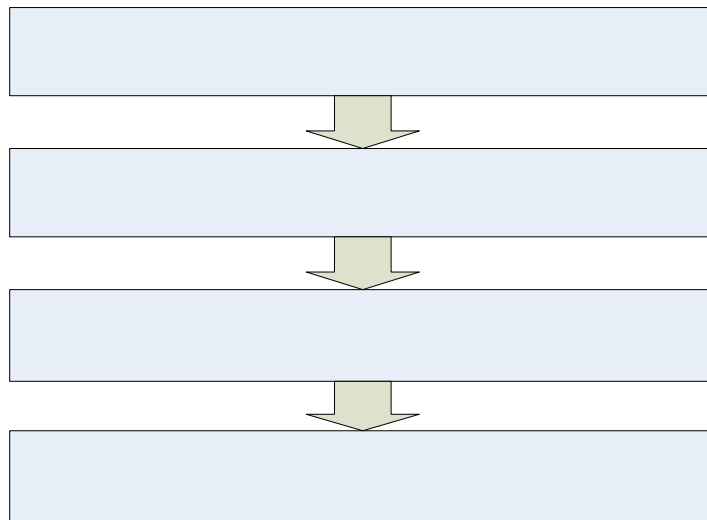
- h. Address book
- i. Language support
- j. Themes support
- k. Search in messages
- l. Personalized order messages
- m. Personal preferences
- n. Send HTML e-mails
- o. Quota Limit

3.1.1 Pengkajian Masalah

Dalam tahap pengkajian masalah terdiri dari tiga sub tahapan yaitu tahap persiapan awal dan tahap analisis dan perancangan sistem mail server yang akan dibangun. Dalam tahap persiapan awal, pemahaman konsep dasar sistem operasi Linux yang menjadi salah satu pondasi dasar dalam membangun *mail* server berbasis Linux, dimana semua perintah yang digunakan menggunakan perintah standar dalam sistem Linux. Selain itu, perlu dilakukan pengayaan akan konsep sistem mail server sebagai referensi untuk menghasilkan sistem yang handal dengan performansi tinggi menggunakan metode interview dengan narasumber serta melakukan kajian literatur tentang topik terkait.

Studi literatur yang memuat paparan konsep dasar pengembangan sistem email server berbasis Linux untuk menghasilkan sebuah sistem server yang handal dan *secure* dalam menangani pengiriman dan penerimaan email dalam suatu jaringan. Pada kasus ini penulis mengimplementasikan MTA (Mail Transfer Agent) Postfix pada *platform* Linux berbasis Slackware 11.

Analisis awal konsep dan pengembangan sistem server email berbasis Linux sebagai langkah awal untuk membangun konsep rancangan sistem yang akan di bangun, untuk menghasilkan output yang optimal. Ada beberapa tahap yang dilakukan dalam proses ini:



Gambar 3.2 Diagram alur analisis dan perancangan sistem *mail* server

Desain sistem email sebagai konsep dasar rancangan sistem email yang akan dibangun, yaitu adalah sebuah sistem *mail server* dengan *cost* yang murah namun memiliki performa dan stabilitas yang baik untuk menangani trafik pengiriman dan penerimaan email dalam suatu jaringan komputer dengan mengimplementasikan aplikasi Postfix sebagai MTA (Mail Transfer Agent) yang dipenetrasikan pada platform Linux khususnya untuk distribusi Slackware yang notabenehnya merupakan salah satu distro tua dengan performa dan stabilitas yang tidak diragukan lagi khususnya dalam menangani berbagai fungsi server. Slackware sudah ikut berperan dalam pengembangan ilmu komputer khususnya dalam bidang *operating system* sejak lahirnya dan merupakan favorit diantara para “sesepeuh”, pakar dan “purnalinuxwan”. Itu tidak terlepas dari kenyataan bahwa Slackware adalah “super stabil” dan merupakan Linux berpenampilan paling mendekati Unix. Slackware adalah ideal bagi yang mengerti seluk-beluk dan isi perutnya Unix. Sangat populer dikalangnya, karena Slackware selain “powerful” juga hemat memori/spasi harddisk.

Pemilihan *mail transfer agent* (MTA) sebagai *agent* misionaris yang bertugas melakukan manajemen pengiriman dan penerimaan email layaknya sebuah kantor pos pada sistem pengiriman surat konvensional untuk mendapatkan performansi dan stabilitas sistem yang baik. Penulis berinisiatif untuk menggunakan Postfix yang terkenal handal dalam melakukan fungsi-fungsi server email dengan berbagai keunggulan yang dimiliki Postfix dibanding MTA pesaingnya yang lain yang sudah ada sebelumnya seperti Sendmail dan Qmail.

Pada bagian *user interface* (*Email client*) akan diimplementasikan sistem berbasis web (*web based*). Semua *mail* yang ditujukan untuk user dalam sebuah sistem email, akan disimpan di server, sehingga apabila user ingin mengambil dan membaca maupun manajemen email untuknya, ia dapat melakukannya dimanapun dan kapanpun selama terhubung dengan server. Dalam hal ini penulis mengimplementasikan aplikasi UebiMiau yang dapat di peroleh secara gratis dari situs resminya <http://www.uebimiau.org> sebagai MUA (*Mail User Agent*). Sampai saat ini webmail yang dibangun dengan kombinasi bahasa pemrograman PHP ini

memiliki catatan yang baik pada pengguna *email client* berbasis web dalam menangani fungsi-fungsi manajemen email user.

Semua aplikasi yang diperlukan dalam sistem ini merupakan software *free* atau gratis, dapat diperoleh dengan mendownload secara gratis dari internet, dengan kata lain untuk membangun sistem mail server ini hanya diperlukan *cost* yang sedikit.

3.1.2 Implementasi dan Uji Coba Sistem

Pada tahap implementasi dan uji coba sistem yang terbagi dalam dua tahap yaitu implementasi dan pengembangan aplikasi dan uji coba sistem. Tahap implementasi dan pengembangan aplikasi meliputi dua sub tahapan yaitu instalasi dan konfigurasi sistem. Seperti yang di jelaskan pada tahap analisis dan perancangan sistem *mail server* dimana dalam penelitian ini akan diimplementasikan sebuah mail server berbasis Linux menggunakan Postfix sebagai MTA dengan *user agent* berbasis web menggunakan UebiMiau, sehingga secara tidak langsung dalam pembangunan sistem email server, dibangun juga sistem server yang lain yaitu Web Server dan DNS server. Secara struktural runtutan proses instalasi dan konfigurasi yang dilakukan adalah sebagai berikut:

1. Instalasi Linux Slackware 11.
2. Instalasi dan konfigurasi Apache server sebagai web server. Pada umumnya Apache server sudah di-*include*-kan dalam paket instalasi Slackware.
3. Instalasi dan konfigurasi Bind 9 sebagai DNS server. Aplikasi ini juga sudah disertakan dalam paket instalasi Slackware 11.
4. Instalasi dan konfigurasi Postfix sebagai MTA (*Mail Transfer Agent*).
5. Instalasi dan konfigurasi IMAP sebagai protokol untuk manajemen *mail* berbasis web.
6. Instalasi dan konfigurasi UebiMiau sebagai MUA (*Mail User Agent*).

Sebagai tahap pamungkas dari tahapan ini adalah dengan melakukan pengujian terhadap sistem yang telah dibangun. Tahapan uji coba yang dilakukan melalui beberapa jenis tes yaitu pengujian dengan mengirimkan email antar user dengan metode *remote* dari client yang menggunakan SO Windows ke sistem

server email yang menggunakan Linux melalui SSH (*secure shell*) menggunakan software putty. Jika uji coba ini berhasil, maka dilanjutkan dengan pengiriman email dengan modus grafis melalui akses web pada web browser. Gambar 3.1 terlihat umpan balik yang dipertimbangkan untuk mengevaluasi sistem.

3.1.3 Dokumentasi

Sebagai tahap paripurna dari seluruh rangkaian penelitian yang dilakukan adalah dengan melakukan dokumentasi dari apa yang telah dilakukan, sehingga kedepan dari apa yang telah dilakukan bisa diambil manfaatnya baik sebagai acuan atau referensi sistem yang sama maupun literatur pembangunan sistem yang lain.

3.2 Alat dan Bahan

Dalam pelaksanaan penelitian tugas akhir ini, beberapa kebutuhan minimum yang harus dipersiapkan diantaranya:

3.2.1 Kebutuhan Alat

Dalam pembangunan server email ini diperlukan beberapa peralatan, diantaranya:

1. Komputer dengan spesifikasi minimum:

Processor	: 386
Memory	: 16 MB
Harddisk	: 400MB (full instalation)
2. CD Instalasi Slackware 11
3. Lan Card
4. Kabel UTP
5. CDROM
6. Mouse + keyboard

Catatan: Spesifikasi diatas adalah spesifikasi minimum, untuk meningkatkan performa disarankan untuk menggunakan spesifikasi yang lebih tinggi.

3.2.2 Kebutuhan Bahan

Berikut adalah kebutuhan software yang diperlukan dalam pembangunan mail server berbasis Linux:

1. Sistem operasi Linux Slackware 11
2. Apache server + PHP. Pada penelitian ini Penulis menggunakan versi apache-1.3.37-i486-2.tgz dan php-4.4.4-i486-3.tgz.
3. MySQL. Versi yang Penulis gunakan adalah mysql-5.0.24a-i486-1.tgz.
4. BIND. Versi yang Penulis gunakan adalah bind-9.3.2_P1-i486-1.tgz.
5. Postfix. Disini penulis menggunakan versi postfix-2.3.7.tar.
6. IMAP server. Pada penelitian ini Penulis menggunakan IMAP server UW IMAP dengan versi imapd-4.64-i486-3.tgz.
7. UebiMiau sebagai aplikasi web mail. Penulis menggunakan versi uebimiau-2.7.2-any.zip yang bisa di download secara gratis dari situs resminya www.uebimiau.org.

Catatan: Sebagian besar software tersebut sudah disertakan dalam CD instalasi Slackware 11.

3.2.3 Kebutuhan Kualifikasi Administrator

Untuk dapat mengimplementasikan aplikasi mail server ini, diperlukan kemampuan minimal yang harus dimiliki oleh sistem administrator, diantaranya:

1. Pemahaman konsep dasar Linux.
2. Pemahaman tentang perintah-perintah dasar Linux.
3. Pemahaman tentang konsep jaringan komputer.
4. Pemahaman tentang sistem email.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Perancangan Sistem

Perancangan sistem merupakan langkah awal yang harus dilakukan dalam proses pembangunan sebuah sistem server email berbasis Linux, khususnya untuk distro Slackware 11 dengan mengimplementasikan sebuah aplikasi MTA (*Mail Transfer Agent*) Postfix yang bertugas menangani manajemen lalu-lintas penerimaan dan pengiriman pesan dalam suatu jaringan komputer. Sebagai media interaksi dengan *user*, pada sistem ini di implementasikan aplikasi email client berbasis web (*web based*), yang artinya aplikasi ini mendukung model *client server*, sehingga integritas dan manajemen email dapat dilakukan secara tersentral pada sebuah sistem email server. Seluruh email yang ditujukan untuk sebuah user dalam suatu sistem *mail server* akan disimpan disisi server, dan user dapat mengambil dan membaca pesan untuknya dari mana saja dan kapan saja selama terkoneksi dengan server. Hal ini berbeda apabila menggunakan *email client* berbasis *desktop* seperti MS Outlook, dimana seluruh email untuk sebuah user akan di download ke komputer client.

Konektifitas dengan user sebagai sebuah interaksi langsung antara user dengan server, dalam hal ini kaitannya dengan manajemen email baik proses penerimaan maupun pengiriman email serta proses-proses yang lain dari sisi *client*, mensyaratkan sebuah server yang dapat menghubungkan *email client* dengan *email server* sebagai sistem penterjemah perintah-perintah yang diberikan user melalui sistem *email client* untuk dapat dimengerti dan dieksekusi oleh sistem server. Server ini yang biasa disebut dengan POP3 atau IMAP. Dalam penelitian ini, penulis menggunakan software UW IMAP sebagai salah satu standar protokol IMAP untuk Linux Slackware. Sebagai aplikasi *email client*, penulis mengimplementasikan software UebiMiau.

4.2 Hasil Penelitian

Dari penelitian ini dihasilkan sebuah sistem *mail server* berbasis Linux menggunakan Postfix yang memiliki kecepatan dan keamanan yang baik dalam menangani manajemen penerimaan maupun pengiriman email yang di penetrasikan pada varian Slackware 11 dengan kombinasi Apache sebagai web server dan BIND9 sebagai DNS server dengan memberikan nama domain www.firdaus.net. Sebagai user interface, di implementasikan *mail client* berbasis web yang mendukung tipe client server, sehingga user dapat mengambil, membaca maupun mengirim email dari mana saja dan kapan saja selama terkoneksi dengan server. Lebih dari itu, model jaringan seperti ini akan lebih memudahkan dari sisi server dalam melakukan manajemen sistem untuk menjaga stabilitas dan performa sistem.

4.3 Pembahasan

Langkah penting dalam pembangunan *mail server* adalah proses implementasi dan aplikasi. Dalam proses ini terdiri dari 3 tahapan yaitu instalasi, konfigurasi dan uji coba.

4.3.1 Instalasi Linux Slackware

Linux Slackware merupakan salah satu varian Linux yang dibangun lebih ditujukan untuk keperluan server dengan kehandalan dan performa yang tinggi. Pada beberapa sistem administrator server Linux, pengguna atau administrator tidak memerlukan tampilan grafis, sehingga instalasi hanya dilakukan pada *base system* dan paket-paket instalasi tertentu yang di perlukan. Alasan ini lebih pada personal masing-masing, dimana untuk sistem server, sistem administrator lebih menyukai modus teks dari pada modus grafis, hal ini juga yang Penulis lakukan, dimana untuk sistem administratornya hanya menggunakan modus teks.

Langkah-langkah instalasi Linux Slackware dapat dituliskan sebagai berikut:

1. Dalam hal ini proses instalasi akan dilakukan dari CD ROM, sehingga langkah awal adalah setting BIOS untuk *booting* pertama dari CD ROM.

2. Selanjutnya jalankan komputer dengan terlebih dahulu masukkan CD instalasi Slackware 11 kedalam CD ROM.
3. Setelah berhasil *booting* dari CD, pada layar monitor akan ditampilkan halaman muka dari instalasi Linux Slackware.

```

ISOLINUX 2.13 2004-12-14 Copyright (C) 1994-2004 H. Peter Anvin

Welcome to Slackware version 11.0 (Linux kernel 2.4.33.3)!

If you need to pass extra parameters to the kernel, enter them at the prompt
below after the name of the kernel to boot (scsi.s etc). NOTE: In most cases
the kernel will detect your hardware, and parameters are not needed.

Here are some examples (and more can be found in the BOOTING file):
  hdx=cyls,heads,sects,upcom,irq (needed in rare cases where probing fails)
or hdx=cdrom (force detection of an IDE/ATAPI CD-ROM drive)
where hdx can be any of hda through hdt.

In a pinch, you can boot your system from here with a command like:

For example, if the Linux system were on /dev/hda1.

boot: sata.i root=/dev/hda1 noinitrd ro

This prompt is just for entering extra parameters. If you don't need to enter
any parameters, hit ENTER to boot the default kernel "sata.i" or press [F2]
for a listing of more kernel choices.

boot:

```

Gambar 4.1 Tampilan awal instalasi Slackware 11

4. Tekan enter untuk melanjutkan proses instalasi yang secara default akan menggunakan jenis kernel “sata.i” yang mendukung untuk penggunaan *harddisk* jenis SATA, sehingga akan ditampilkan halaman login.

```

- If you do not have a color monitor, type: TERM=vt100
  before you start 'setup'.

You may now login as 'root'.

slackware login:

```

Gambar 4.2 Tampilan login instalasi

Tekan enter untuk login secara default sebagai root.

5. Langkah selanjutnya adalah membuat partisi harddisk. Untuk membuat partisi harddisk digunakan perintah *fdisk* yang sudah *build in* pada CD instalasi Linux Slackware 11, yaitu dengan mengetikkan perintah *fdisk* pada *shell* Linux diikuti tombol enter, maka akan ditampilkan program partisi ala Linux. Dalam hal ini minimal diperlukan 2 partisi yaitu partisi untuk swap dan base

system Linux. Besar kecilnya kapasitas harddisk tergantung dari kebutuhan. Berikut adalah alokasi harddisk yang penulis berikan untuk membangun sistem ini:

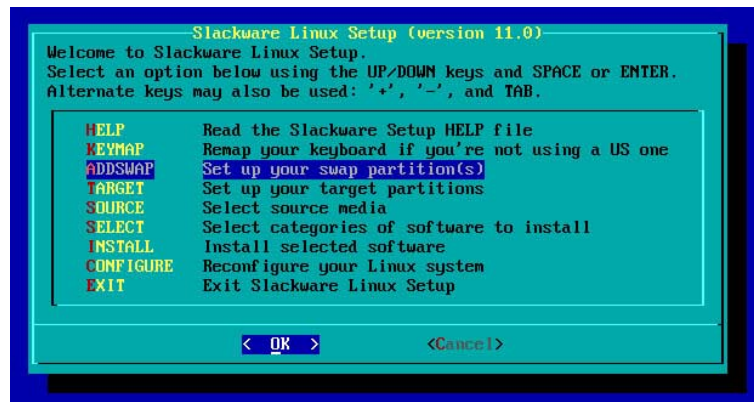
Tabel 4.1 Tabel partisi harddisk

Nama Partisi	Type File Sistem	Besar(MB)
/dev/hda1	Ext3	4000
/dev/hda2	SWAP	512



Gambar 4.3 Program partisi *cfdisk*

- Setelah proses partisi selesai, langkah selanjutnya adalah memulai proses instalasi dan konfigurasi sistem operasi Linux Slackware 11. Proses ini dimulai dengan mengetikkan perintah *setup* pada prompt yang ada diikuti tombol *enter* maka secara otomatis akan tampil menu program instalasi *wizard* Slackware 11.



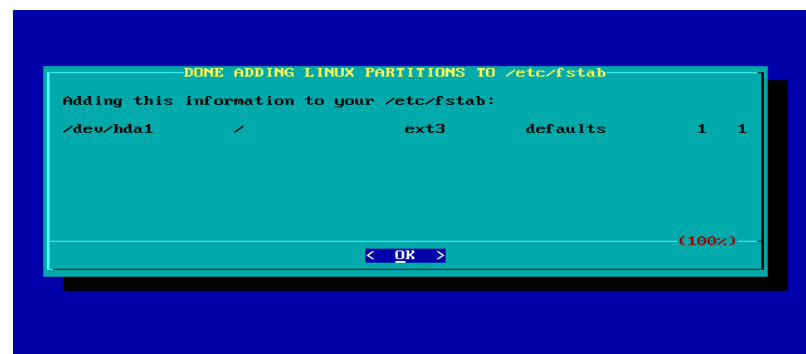
Gambar 4.4 Program Instalasi Wizard Slackware 11

7. Pada tahap ini langsung pilih menu ADDSWAP yang digunakan untuk menambah partisi SWAP pada sistem operasi Linux.
8. Pada langkah awal, partisi SWAP sudah dibuat sehingga bisa langsung sistem dapat langsung mengenalinya, sehingga tinggal memilih tekan tombol *Yes* untuk mulai mengkonfigurasi partisi tersebut sebagai partisi SWAP. Tunggu sampai proses konfigurasi SWAP ini selesai dan kemudian tinggal memilih *option OK*.



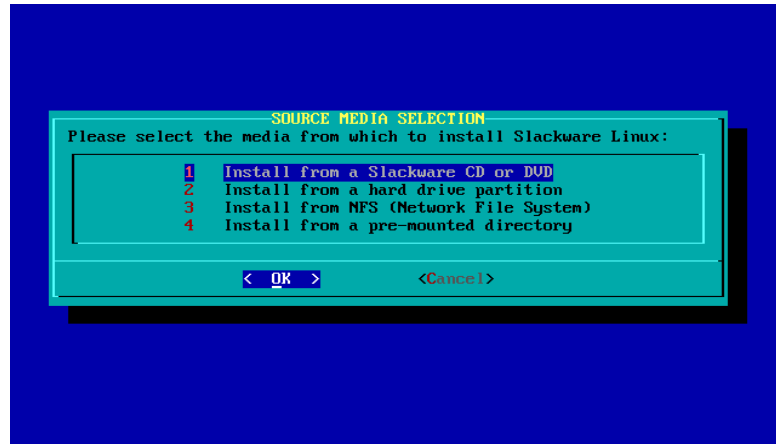
Gambar 4.5 Konfigurasi Partisi SWAP

9. Penambahan partisi root(/), pada langkah ini dilakukan pemilihan partisi bertipe Linux yang sebelumnya telah dibuat, kemudian pilih *file* sistem bertipe ext3 untuk partisi Linux ini, setelah itu pilih *Inod Density* pada pilihan *default* yaitu: 4096 (1 node per 4096 byte). Tunggu sampai proses *formatting* selesai dan tinggal memilih *option OK* untuk melanjutkan langkah selanjutnya.



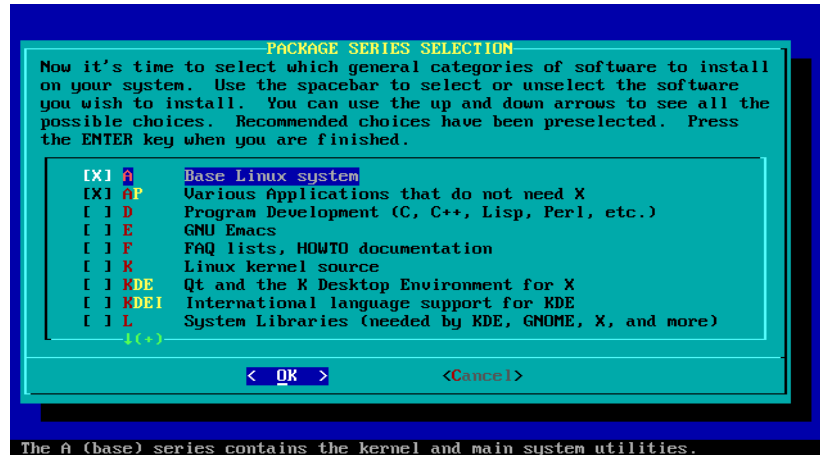
Gambar 4.6 Penambahan Partisi root(/)

10. Selanjutnya adalah pemilihan sumber instalasi Linux Slackware 11 ini. Pilih option pertama untuk melakukan instalasi dari CD, dan akan diteruskan dengan proses *scanning source media* yang telah dipilih tersebut, tunggu sampai proses *scanning* selesai yang menandakan bahwa CD instalasi dapat terdeteksi oleh sistem.



Gambar 4.7 Pemilihan media sumber instalasi

11. Proses selanjutnya adalah menentukan paket-paket instalasi yang diperlukan. Pembangunan *mail server* ini dilakukan menggunakan modus teks yang berarti tidak membutuhkan aplikasi grafis. Beberapa *series package* yang perlu diinstall adalah A, AP, D dan N. *Series A* adalah paket yang berisi *base linux system* yang harus ada pada setiap sistem operasi Linux, AP berisi paket atau aplikasi yang tidak membutuhkan *X windows system*, D berisi paket-paket *development* atau paket-paket utiliti untuk mengembangkan sistem Linux, sedangkan *series N* berisi paket atau aplikasi yang berhubungan *networking*. Berikan tanda cek dengan menekan tombol *space*. Setelah memilih keempat kumpulan paket di tersebut, tekan tombol OK untuk melanjutkan proses instalasi.



Gambar 4.8 Pemilihan *Series Packages* Instalasi

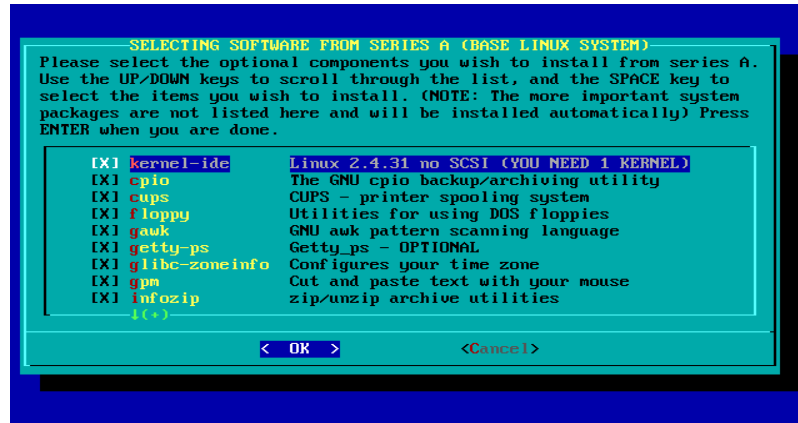
12. Pada langkah ini dilakukan pemilihan model instalasi paket (*prompting mode*). Tidak semua paket yang ada dalam *series* paket yang telah dipilih sebelumnya akan diinstal, hanya pada paket-paket tertentu yang diperlukan, sehingga disini pilih option Menu.



Gambar 4.9 Pemilihan Mode Instalasi Paket

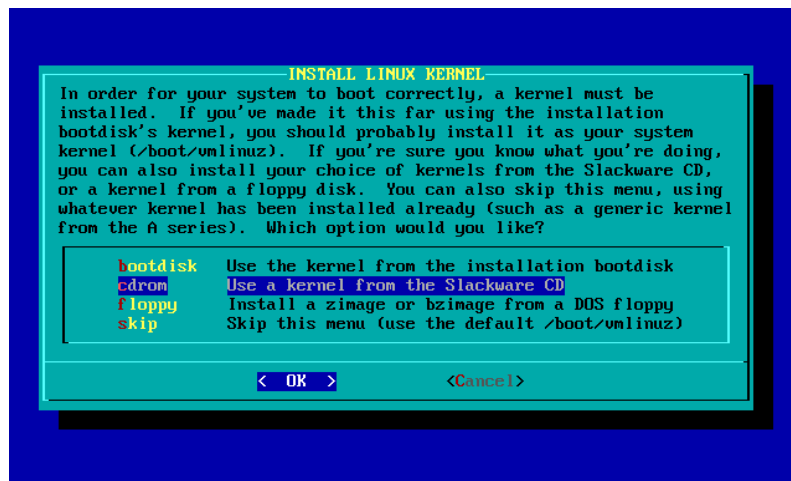
13. Langkah selanjutnya adalah pemilihan paket instalasi dari masing-masing *series* paket. Pemilihan paket dilakukan dengan menekan tombol *space* pada paket tersebut yang ditandai dengan tanda " x ". Tekan tombol Ok untuk melanjutkan ke daftar paket selanjutnya. Setelah semua paket dipilih, akan dilanjutkan dengan proses instalasi paket-paket tersebut. Proses ini memakan

waktu yang cukup lama, tergantung dari spesifikasi komputer dan jumlah paket yang dipilih.



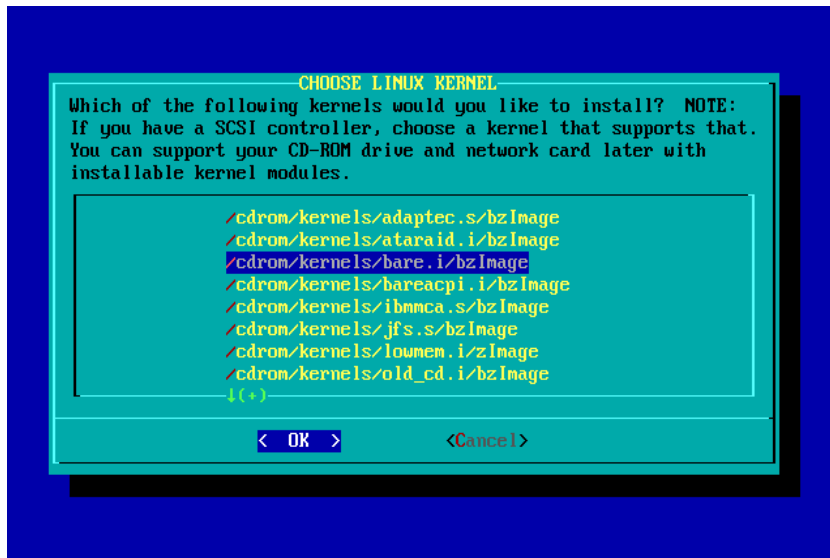
Gambar 4.10 Pemilihan Paket Instalasi

14. Setelah proses instalasi paket sistem Linux selesai, dilanjutkan dengan instalasi kernel. Dalam hal ini dipilih *cdrom* karena kernel yang digunakan berasal dari CD instalasi Linux Slackware 11.



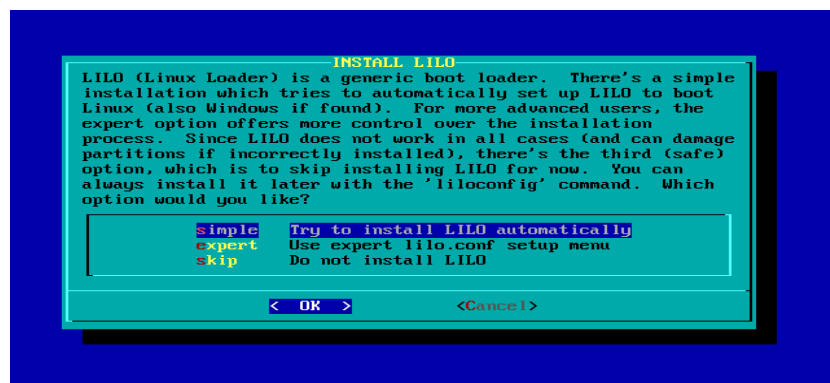
Gambar 4.11 Pemilihan media sumber instalasi Kernel

15. Selanjutnya pilih kernel yang sesuai dengan spesifikasi *hardware* dimiliki.



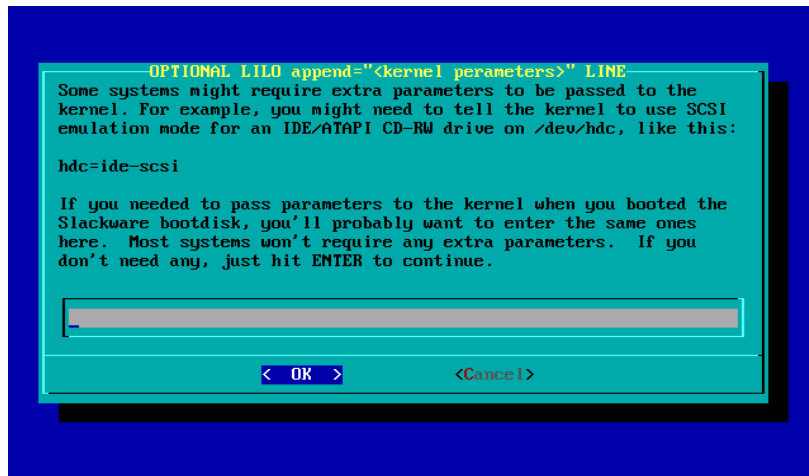
Gambar 4.12 Pemilihan Kernel Linux

16. Pembuatan *bootdisk*, langkah ini diabaikan dengan memilih *option* skip.
17. Langkah selanjutnya adalah konfigurasi modem, langkah ini diabaikan dengan memilih *option* no modem karena komputer ini tidak menggunakan modem. Lanjutkan dengan menekan tombol OK.
18. Selanjutnya adalah penentuan *subsystem hot plug* akan diaktifkan atau tidak, proses ini berfungsi untuk mendeteksi dan menyesuaikan semua perubahan yang terjadi pada sistem seperti penambahan *hardware* atau *peripheral* baru pada komputer. Pilih *option* yes untuk mengaktifkannya.
19. Dilanjutkan instalasi LILO (*Linux Loader*), untuk model instalasi *default* pilih *option* simple, untuk menginstal LILO secara otomatis.



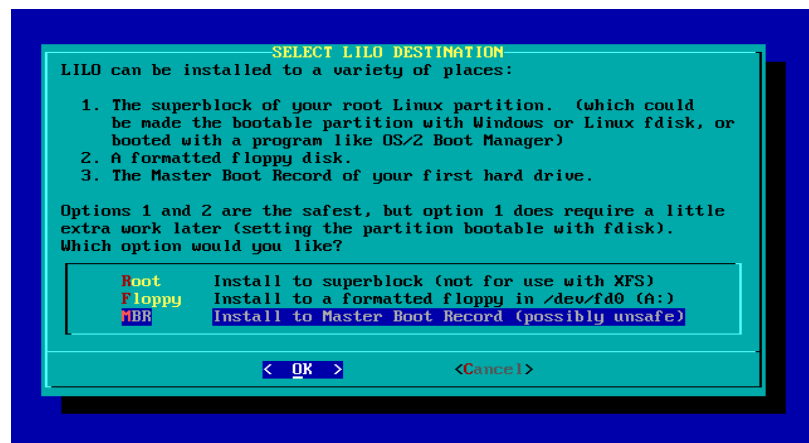
Gambar 4.13 Konfigurasi mode instalasi LILO

20. Kemudian pilih *frame buffer console default* yaitu sebesar 1024x768x256 karena monitor yang digunakan menggunakan resolusi 1024x768, setting ini tergantung resolusi monitor yang digunakan
21. Selanjutnya untuk isian *optional lilo append* dikosongkan.



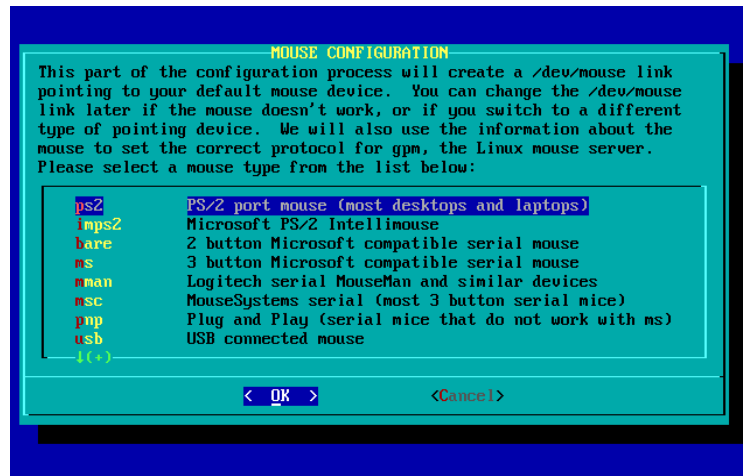
Gambar 4.14 Konfigurasi *Frame Buffer Console*

22. Langkah selanjutnya adalah pemilihan lokasi dimana LILO akan diinstal. Disini dipilih *option MBR* agar LILO diinstal pada *master boot record* pada *hardisk* komputer.



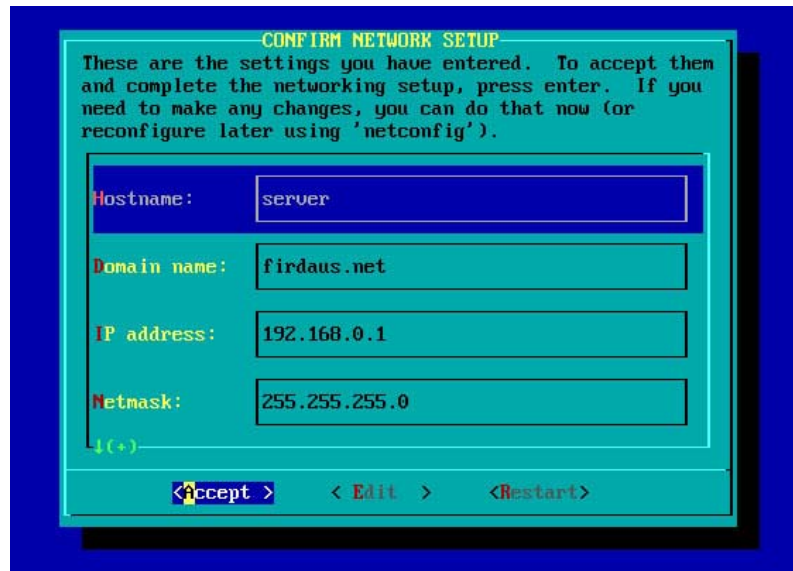
Gambar 4.15 Konfigurasi *LILO Destination*

23. Dilanjutkan konfigurasi jenis mouse yang digunakan. Disini dipilih *ps2* karena mouse yang digunakan menggunakan port PS2.



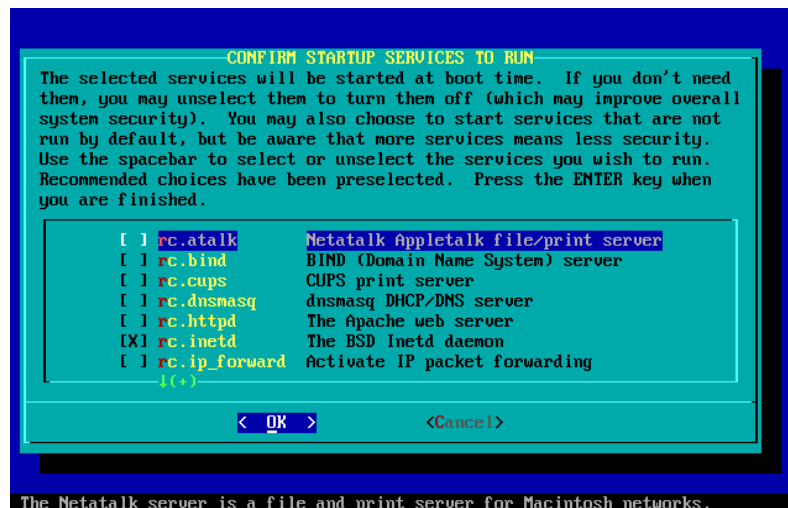
Gambar 4.16 Konfigurasi Mouse

24. Langkah selanjutnya adalah mengkonfigurasi GPM program, program ini berfungsi agar *mouse* bisa melakukan fungsi *cut-paste* pada layar *virtual console*. Pilih *yes*.
25. Selanjutnya adalah konfigurasi jaringan komputer. Pilih *Yes* untuk melanjutkan proses konfigurasi. Pada tahap awal konfigurasi ini penulis memasukkan nama *hostname* yaitu server dan domain adalah *firdaus.net*. Langkah selanjutnya adalah konfigurasi *IP Address*, karena komputer server ini menggunakan *IP Static* maka pilih *static IP*, kemudian masukkan IP komputer server yaitu *192.168.0.1*, dilanjutkan dengan menentukan *netmask* komputer server, isi dengan *netmask default IP* kelas C yaitu *255.255.255.0*. Langkah berikutnya adalah menentukan *gateway* dan *name server*, masing-masing isi dengan *192.168.0.1*.
26. Setelah proses konfigurasi jaringan selesai, maka akan muncul *Confirm Network Setup*, pilih *Accept* jika konfigurasi sudah benar.



Gambar 4.17 Konfirmasi *Network Setup*

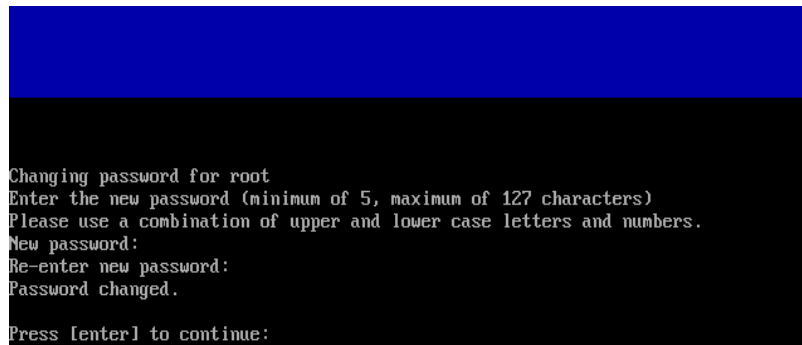
27. Langkah selanjutnya adalah mengkonfigurasi servis apa saja yang akan dijalankan setiap kali komputer dinyalakan. Langkah ini diatur dalam jendela *Confirm StarUp Services To Run*. Tandai pada option *rc.bind*, *rc.httpd*, dan *rc.inetd*, selanjutnya tekan OK.



Gambar 4.18 Konfigurasi *Startup Services To Run*

28. Pengkonfigurasian jenis huruf yang digunakan oleh sistem, gunakan *font default* dengan memilih *option No*.

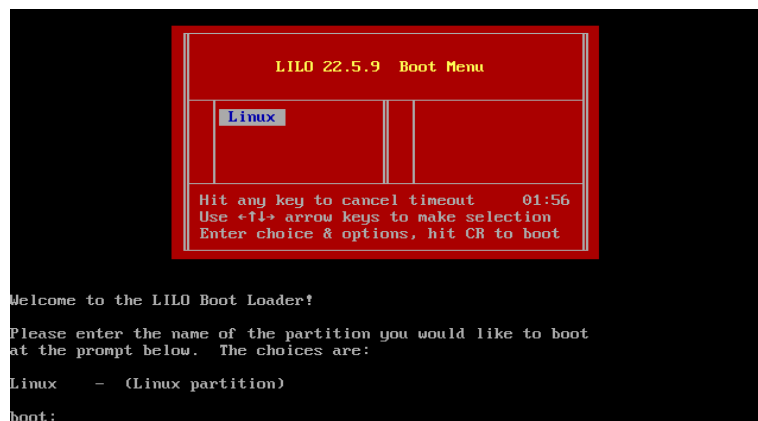
29. *Setting* waktu sistem, pilih No agar waktu sistem diset secara lokal, kemudian pilih lokasi waktu yaitu Asia/Jakarta.
30. Sebagai langkah terakhir adalah setting *password* root, pada langkah ini penulis bisa memasukkan password sesuai keinginan penulis.]



```
Changing password for root
Enter the new password (minimum of 5, maximum of 127 characters)
Please use a combination of upper and lower case letters and numbers.
New password:
Re-enter new password:
Password changed.
Press [enter] to continue:
```

Gambar 4.19 Konfigurasi *Password root*

31. Setelah semua proses di atas selesai maka akan muncul jendela *Setup Complete*, yang menandakan proses instalasi Linux Slackware 11 telah selesai. Setelah itu akan kembali ke menu utama instalasi dan pilih Exit kemudian menekan Ctrl + Alt + Del untuk melakukan *reboot* sistem. Keluarkan CD instalasi dari CD ROM.
32. Berikut tampilan LILO dan proses *booting* serta tampilan login Slackware 11 hasil instalasi. Lanjutkan dengan menekan *enter*.



```
LILO 22.5.9 Boot Menu
Linux
Hit any key to cancel timeout 01:56
Use +1+ arrow keys to make selection
Enter choice & options, hit CR to boot

Welcome to the LILO Boot Loader!
Please enter the name of the partition you would like to boot
at the prompt below. The choices are:
Linux - (Linux partition)
boot:
```

Gambar 4.20 Tampilan LILO

4.3.2 Konfigurasi Web Server (Apache Server)

Apache merupakan salah satu aplikasi web server standar Linux dengan unjuk kerja yang tidak diragukan lagi untuk memberikan respon terhadap setiap permintaan dari user akan suatu halaman web. Sistem email yang dibangun menggunakan aplikasi *email client* berbasis web, untuk itu layanan web server juga di perlukan untuk melakukan manajemen layanan website.

Pada umumnya Apache sudah disertakan dalam paket instalasi pada CD Linux. Sehingga apabila dalam proses instalasi Linux dipilih tipe *full installation* yang artinya akan melakukan instalasi untuk seluruh paket-paketnya, secara otomatis Apache juga sudah terinstal. Namun demikian, instalasi Apache juga dapat dilakukan secara manual setelah sistem berhasil di instal.

Untuk mendapatkan sistem yang sesuai, maka perlu dilakukan beberapa konfigurasi terhadap file konfigurasi Apache. Pada distro Slackware konfigurasi tersebut terletak pada direktori `/etc/apache/httpd.conf`. Gunakanan teks editor untuk melakukan editing file tersebut, dalam hal ini Penulis menggunakan teks editor *pico*. Adapun langkah-langkah instalasi dan konfigurasi secara manual aplikasi Apache adalah sebagai berikut:

1. Instalasi akan dilakukan dari CD instalasi Slackware. Masukkan CD instalasi ke dalam CD ROM, selanjutnya lakukan *mounting* CD ROM.

```
# mount /mnt/cdrom/
```

```
# cd /mnt/cdrom/
```

Catatan: Tanda # menandakan bahwa user login sebagai *root*

2. Paket instalasi standar Slackware adalah file dengan ekstensi **.tgz*. Selanjutnya lakukan proses instalasi dengan memberikan perintah:

```
# installpkg /mnt/cdrom/slackware/n/apache-1.3.37-i486-2.tgz
```

3. Lakukan konfigurasi Apache web server. Edit file `/etc/apache/httpd.conf`

```
# pico /etc/apache/httpd.conf
```

Edit file tersebut seperti berikut:

```
ServerType standalone
```

```
ServerName server.firdaus.net
```

```
ServerAdmin servadmin@firdaus.net
```

Servername berisikan parameter nama host komputer diikuti nama domain dari mesin server tersebut. Dalam hal ini diberikan nilai parameter Servername server sebagai nama komputer dan firdaus.net sebagai domainnya.

```
ServerRoot "/usr"
PidFile /var/run/httpd.pid
ScoreBoardFile /var/run/httpd.scoreboard
Timeout 300
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15
MinSpareServers 5
MaxSpareServers 10
StartServers 5
MaxClients 150
MaxRequestsPerChild 0
Port 80
User nobody
Group nobody
HostnameLookups Off
UseCanonicalName On
Listen 80
Listen 443
```

Berikut adalah daftar modul yang akan di panggil ketika service apache dijalankan.

```
LoadModule vhost_alias_module libexec/apache/mod_vhost_alias.so
LoadModule env_module libexec/apache/mod_env.so
LoadModule define_module libexec/apache/mod_define.so
LoadModule config_log_module libexec/apache/mod_log_config.so
LoadModule mime_magic_module libexec/apache/mod_mime_magic.so
LoadModule mime_module libexec/apache/mod_mime.so
LoadModule negotiation_module libexec/apache/mod_negotiation.so
LoadModule status_module libexec/apache/mod_status.so
LoadModule info_module libexec/apache/mod_info.so
LoadModule includes_module libexec/apache/mod_include.so
```

```

LoadModule autoindex_module    libexec/apache/mod_autoindex.so
LoadModule dir_module          libexec/apache/mod_dir.so
LoadModule cgi_module          libexec/apache/mod_cgi.so
LoadModule asis_module         libexec/apache/mod_asis.so
LoadModule action_module       libexec/apache/mod_actions.so
LoadModule speling_module      libexec/apache/mod_speling.so
LoadModule userdir_module      libexec/apache/mod_userdir.so
LoadModule alias_module        libexec/apache/mod_alias.so
LoadModule rewrite_module      libexec/apache/mod_rewrite.so
LoadModule access_module       libexec/apache/mod_access.so
LoadModule auth_module         libexec/apache/mod_auth.so
LoadModule expires_module      libexec/apache/mod_expires.so
LoadModule headers_module      libexec/apache/mod_headers.so
LoadModule usertrack_module    libexec/apache/mod_usertrack.so
LoadModule log_forensic_module
libexec/apache/mod_log_forensic.so
LoadModule setenvif_module     libexec/apache/mod_setenvif.so
LoadModule php4_module         libexec/apache/libphp4.so

ClearModuleList
AddModule mod_vhost_alias.c
AddModule mod_env.c
AddModule mod_define.c
AddModule mod_log_config.c
AddModule mod_mime_magic.c
AddModule mod_mime.c
AddModule mod_negotiation.c
AddModule mod_status.c
AddModule mod_info.c
AddModule mod_include.c
AddModule mod_autoindex.c
AddModule mod_dir.c
AddModule mod_cgi.c
AddModule mod_asis.c
AddModule mod_actions.c
AddModule mod_speling.c
AddModule mod_userdir.c
AddModule mod_alias.c

```

```

AddModule mod_rewrite.c
AddModule mod_access.c
AddModule mod_auth.c
AddModule mod_expires.c
AddModule mod_headers.c
AddModule mod_usertrack.c
AddModule mod_log_forensic.c
AddModule mod_so.c
AddModule mod_setenvif.c
AddModule mod_php4.c

DirectoryIndex index.html index.htm index.php index.shtml
default.htm
AddType application/x-httpd-php .php
AddType application/x-httpd-php-source .phps
AccessFileName .htaccess
<Files ~ "\.ht">
    Order allow,deny
    Deny from all
    Satisfy All
</Files>
<Directory />
    Options FollowSymLinks Includes
    AllowOverride All
</Directory>
DocumentRoot "/var/www/htdocs"
<Directory "/var/www/htdocs">
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

<IfModule mod_watch.c>
    <Location /watch-info>
        SetHandler watch-info
    </Location>
</IfModule>

```

```

<IfModule mod_dir.c>
    DirectoryIndex index.html
</IfModule>
<IfModule mod_mime.c>
    TypesConfig /etc/apache/mime.types
</IfModule>
DefaultType text/plain

<IfModule mod_mime_magic.c>
    MIMEMagicFile /etc/apache/magic
</IfModule>

ErrorLog /var/log/apache/error_log
LogLevel warn
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
CustomLog /var/log/apache/access_log common
ServerSignature On
<IfModule mod_alias.c>
    Alias /icons/ "/var/www/icons/"
    <Directory "/var/www/icons">
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>
</IfModule>

<IfModule mod_autoindex.c>
    IndexOptions FancyIndexing
    AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-
gzip
    AddIconByType (TXT,/icons/text.gif) text/*
    AddIconByType (IMG,/icons/image2.gif) image/*
    AddIconByType (SND,/icons/sound2.gif) audio/*

```



```

AddIconByType (VID,/icons/movie.gif) video/*
AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/world2.gif .wrl .wrl.gz .vrml .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
AddIcon /icons/a.gif .ps .ai .eps
AddIcon /icons/layout.gif .html .shtml .htm .pdf
AddIcon /icons/text.gif .txt
AddIcon /icons/c.gif .c
AddIcon /icons/p.gif .pl .py
AddIcon /icons/f.gif .for
AddIcon /icons/dvi.gif .dvi
AddIcon /icons/uuencoded.gif .uu
AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
AddIcon /icons/tex.gif .tex
AddIcon /icons/bomb.gif core
AddIcon /icons/back.gif ..
AddIcon /icons/hand.right.gif README
AddIcon /icons/folder.gif ^^DIRECTORY^^
AddIcon /icons/blank.gif ^^BLANKICON^^
DefaultIcon /icons/unknown.gif
ReadmeName README
HeaderName HEADER
IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t
</IfModule>

```

```

<IfModule mod_mime.c>
AddCharset ISO-8859-8 .iso8859-8
AddCharset ISO-8859-2 .iso-pl
AddCharset Big5 .Big5 .big5
AddCharset WINDOWS-1251 .cp-1251
AddCharset CP866 .cp866
AddCharset UTF-8 .utf8
AddType application/x-tar .tgz
AddEncoding x-compress .Z
AddEncoding x-gzip .gz .tgz
</IfModule>

```

```

<IfModule mod_setenvif.c>
    BrowserMatch "Mozilla/2" nokeepalive
    BrowserMatch "MSIE 4\.0b2;" nokeepalive downgrade-1.0
force-response-1.0
    BrowserMatch "RealPlayer 4\.0" force-response-1.0
    BrowserMatch "Java/1\.0" force-response-1.0
    BrowserMatch "JDK/1\.0" force-response-1.0
</IfModule>

<IfModule mod_ssl.c>
SSLPassPhraseDialog builtin
SSLSessionCache dbm:/var/log/apache/ssl_cache
SSLSessionCacheTimeout 300
SSLMutex file:/var/log/apache/ssl_mutex
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
SSLLog /var/log/apache/ssl_engine_log
SSLLogLevel info
</IfModule>
<IfDefine SSL>
AddType application/x-x509-ca-cert .crt
AddType application/x-pkcs7-crl .crl
</IfDefine>

```

Berikut adalah inisialisasi host dimana web server dijalankan

```
NameVirtualHost server.firdaus.net
```

Berikut adalah konfigurasi virtual domain untuk alamat domain www.firdaus.net sebagai alamat utama website, yang artinya jika ada permintaan untuk mengakses alamat www.firdaus.net maka script ini yang akan di jalankan. *Document root* berisikan parameter lokasi dimana file-file web site yang akan di tampilkan di *web browser* diletakkan. Secara *default* pada Linux Slackware, lokasi ini berada pada direktori `/var/www/htdocs`, namun dalam penelitian ini lokasi dokumen root dari domain ini akan diletak dalam folder `/home/firdaus.net/www/web` yang artinya jika ada permintaan

untuk mengakses alamat www.firdaus.net, maka semua file dalam folder ini yang akan dieksekusi oleh web server. Semua catatan aktivitas dari servis Apache akan di simpan dalam file /home/firdaus.net/www/logs/error.

```
<VirtualHost server.firdaus.net>
    ServerName www.firdaus.net
    ServerAdmin servadmin@localhost
    DocumentRoot '/home/firdaus.net/www/web'
    ErrorLog /home/firdaus.net/www/logs/error
    CustomLog /home/firdaus.net/www/logs/access combined
    ServerAlias firdaus.net
    php_admin_flag engine On
    AddType application/x-httpd-php .php .php4 .php3 .phtml
    AddType application/x-httpd-php-source .phps
    AddType text/html .shtml
    AddHandler server-parsed .shtml
    AddHandler cgi-script .cgi .pl
    <Directory '/home/firdaus.net/www/web'>
        Options +ExecCGI
    </Directory>
</VirtualHost>
```

Berikut adalah konfigurasi untuk virtual host dengan alamat mail.firdaus.net sebagai alamat dari webmail. File-file web untuk webmail tersebut disimpan pada direktori /home/firdaus.net/www/mail

```
<VirtualHost server.firdaus.net>
    ServerName mail.firdaus.net
    ServerAdmin servadmin@localhost
    DocumentRoot '/home/firdaus.net/www/mail'
    ErrorLog /home/firdaus.net/www/logs/mail_error
    CustomLog /home/firdaus.net/www/logs/access combined
</VirtualHost>
```

4. Setelah proses editing selesai, langkah selanjutnya adalah membuat direktori yang akan menyimpan seluruh file website yang akan dieksekusi oleh web server.

```
# mkdir -p /home/firdaus.net/www/web
```

```
# mkdir -p /home/firdaus.net/www/mail
```

```
# mkdir -p /home/firdaus.net/www/logs
```

5. Setelah melakukan editing file konfigurasi Apache, langkah selanjutnya adalah menjalankan service Apache.

```
# chmod +x /etc/rc.d/rc.httpd
```

```
# /etc/rc.d/rc.httpd start
```

6. Sampai disini proses instalasi dan konfigurasi web server sudah selesai, selanjutnya dapat dilakukan uji coba awal dengan membuat sebuah file `index.html` atau `index.php` kemudian simpan di direktori `/home/firdaus.net/www/web/`. Kemudian melalui jendela browser masukkan alamat `http://192.168.0.1/`

Catatan : alamat 192.168.0.1 merupakan IP *address* dari server tersebut.

4.3.3 Konfigurasi DNS Server (BIND 9)

Domain Name Server adalah suatu sistem penamaan sebuah host dalam suatu jaringan komputer dengan mengubah IP address menjadi sebuah rangkaian huruf yang mengidentifikasikan nama dari komputer tersebut. Dalam penelitian ini akan dibuat DNS server dengan nama domain www.firdaus.net sebagai identitas dari server tersebut.

Seperti halnya Apache, BIND sebagai software aplikasi DNS server, pada distro Slackware aplikasi ini sudah disertakan dalam paket instalasinya. Namun proses instalasi dapat dilakukan secara manual dari CD instalasi Linux maupun dari *source* yang lain. Adapun langkah-langkah instalasi dan konfigurasi DNS server adalah sebagai berikut:

1. Setelah CD instalasi Slackware dimasukkan, kaitkan sistem dengan CD ROM.

```
# mount /mnt/cdrom/
```

```
# cd /mnt/cdrom/
```

2. Install paket BIND

```
# installpkg /mnt/cdrom/slackware/n/bind-9.3.2_P1-i486-1.tgz
```

3. Lakukan konfigurasi DNS server dengan mengedit file `/etc/named.conf`. File ini berisi konfigurasi nama-nama domain yang akan digunakan oleh mesin server. Berikut adalah konfigurasi yang dilakukan:

```
acl "xfer" {
    127.0.0.1;
    192.168.0.1;
    192.168.0.0/24;
};

options {
    directory "/var/named";
    pid-file "/var/run/named/named.pid";
    allow-transfer { xfer; };
    notify yes;
    recursion yes;
    allow-recursion { xfer; };
    allow-query { any; };
};
```

Konfigurasi diatas menyatakan bahwa semua file konfigurasi untuk semua zona yang didenisikan dalam file ini, tersimpan dalam direktori `/var/named/`

Berikut adalah inisialisasi daftar nama-nama top level domain (TLD) yang ada di seluruh dunia yang tersimpan dalam file `/var/named/named.ca`.

```
zone "." IN {
    type hint;
    file "named.ca";
};
```

Berikut ini pendefinisian untuk zona localhost atau komputer itu sendiri yang berisikan konfigurasi untuk memetakan nama FQDN dari mesin server ke alamat IP. File konfigurasi ini tersimpan dalam file `/var/named/localhost.zone`.

```
zone "localhost" IN {
    type master;
```

```
file "localhost.zone";
allow-update { none; };
};
```

Berikut ini pendefinisian untuk zona 0.0.127.in-addr.arpa sebagai *reserve lookup* dari mesin localhost.

```
zone "0.0.127.in-addr.arpa" IN {
    type master;
    file "named.local";
    allow-update { none; };
};

controls {
    inet 127.0.0.1 port 953
        allow { 127.0.0.1; } keys { "rndc-key"; };
};
include "/etc/rndc.key";
```

Berikut adalah inisialisasi zona domain untuk server email yang akan dibangun yaitu firdaus.net sebagai domain yang akan menangani semua halaman website utama dari domain tersebut. File-file konfigurasinya disimpan pada file /var/named/firdaus.net.zone.

```
zone "firdaus.net" {
    type master;
    file "/var/named/firdaus.net.zone";
    allow-update { none; };
};
```

Berikut ini pendefinisian untuk zona 0.168.192.in-addr.arpa sebagai *reserve lookup* dari mesin firdaus.net.

```
zone "0.168.192.in-addr.arpa"{
    type master;
    file "/var/named/0.168.192.in-addr.arpa.zone";
    allow-update { none; };
};
```

4. Langkah selanjutnya adalah membuat zona yang berisi konfigurasi untuk masing-masing zona yang di definisikan dalam file /etc/named.conf.

```
# pico /var/named/firdaus.net.zone
```

```
$TTL 86400
```

```
@    IN      SOA      server.firdaus.net. servadmin.firdaus.net. (
                                2007061803      ;
                                10800          ;
                                3600           ;
                                604800        ;
                                86400 )       ;

                                NS      server.firdaus.net.
                                MX      10 server.firdaus.net.
```

```
firdaus.net.  IN      A      192.168.0.1 ; Nohost
www           IN      A      192.168.0.1
server       IN      A      192.168.0.1
mail         IN      A      192.168.0.1
```

```
#pico /var/named/0.168.192.in-addr.arpa.zone
```

```
$TTL      86400
```

```
$ORIGIN localhost.
```

```
@    IN      SOA      server.firdaus.net. servadmin.firdaus.net. (
                                20070629      ; serial
                                3H            ; refresh
                                15M          ; retry
                                1W           ; expiry
                                1D )         ; minimum

                                IN      NS      firdaus.net.
1    IN      PTR     firdaus.net.
```

```

# pico /var/named/named.ca
;      last update:      Jan 29, 2004
;      related version of root zone:      2004012900
;      ftp://ftp.rs.internic.net/domain/named.root

.              3600000  IN  NS      A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.  3600000      A    198.41.0.4
.              3600000      NS    B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET.  3600000      A    192.228.79.201
.              3600000      NS    C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET.  3600000      A    192.33.4.12
.              3600000      NS    D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET.  3600000      A    128.8.10.90
.              3600000      NS    E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET.  3600000      A    192.203.230.10
.              3600000      NS    F.ROOT-SERVERS.NET.
F.ROOT-SERVERS.NET.  3600000      A    192.5.5.241
.              3600000      NS    G.ROOT-SERVERS.NET.
G.ROOT-SERVERS.NET.  3600000      A    192.112.36.4
.              3600000      NS    H.ROOT-SERVERS.NET.
H.ROOT-SERVERS.NET.  3600000      A    128.63.2.53
.              3600000      NS    I.ROOT-SERVERS.NET.
I.ROOT-SERVERS.NET.  3600000      A    192.36.148.17
.              3600000      NS    J.ROOT-SERVERS.NET.
J.ROOT-SERVERS.NET.  3600000      A    192.58.128.30
.              3600000      NS    K.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET.  3600000      A    193.0.14.129
.              3600000      NS    L.ROOT-SERVERS.NET.
L.ROOT-SERVERS.NET.  3600000      A    198.32.64.12
.              3600000      NS    M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET.  3600000      A    202.12.27.33
; End of File

```

File diatas adalah pendefinisian daftar top level domain (TLD) yang ada diseluruh dunia.


```
# pico /var/named/named.local
```

```
$TTL      86400
@         IN      SOA      localhost. root.localhost. (
                                1997022700 ; Serial
                                28800      ; Refresh
                                14400      ; Retry
                                3600000    ; Expire
                                86400 )    ; Minimum

                                IN      NS      localhost.

1         IN      PTR      localhost.
```

```
# pico /var/named/localhost.zone
```

```
$TTL      86400
$ORIGIN   localhost.
@         1D IN SOA      @ root (
                                42          ; serial
                                3H         ; refresh
                                15M        ; retry
                                1W         ; expiry
                                1D )       ; minimum

                                1D IN NS    @
                                1D IN A     127.0.0.1
```

5. Langkah selanjutnya adalah menjalankan service BIND dengan terlebih dulu mengubah mode aksesnya agar dapat di eksekusi.

```
# chmod +x /etc/rc.d/rc.bind
```

```
# /etc/rc.d/rc.bind start
```

6. Langkah akhir dalam konfigurasi DNS server adalah uji coba. Untuk melakukannya dapat menggunakan salah satu perintah berikut ini:

```
# dig @localhost firdaus.net
```

```
; <<>> DiG 9.3.2-P1 <<>> @localhost firdaus.net
; (1 server found)
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 38560
```

```

;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1,
ADDITIONAL: 1

;; QUESTION SECTION:
;firdaus.net.                IN      A

;; ANSWER SECTION:
firdaus.net.                86400  IN      A      192.168.0.1

;; AUTHORITY SECTION:
firdaus.net.                86400  IN      NS
server.firdaus.net.

;; ADDITIONAL SECTION:
server.firdaus.net.        86400  IN      A      192.168.0.1

;; Query time: 39 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Fri Jun 29 17:21:27 2007
;; MSG SIZE rcvd: 82

```

Atau

```
# nslookup firdaus.net
```

```
Server:          192.168.0.1
Address:         192.168.0.1#53
```

```
Name:   firdaus.net
Address: 192.168.0.1
```

Jika respon yang diterima seperti diatas, berarti server DNS telah berjalan dengan baik.

7. Agar server mengidentifikasi sistem dengan prioritas nama DNS nya dibanding *hostname*, ubah file */etc/host.conf* seperti berikut

```
# pico /etc/host.conf
order bind, hosts
multi on
```

4.3.4 Instalasi dan Konfigurasi Postfix

Dalam proses instalasi Postfix ini penulis menggunakan format file *.tgz sebagai format standar untuk Linux Slackware. Pada proses instalasi ini diperlukan servis MySQL, sehingga sebelum memulai instalasi Postfix perlu dilakukan konfigurasi MySQL. Berikut adalah langkah-langkah yang dilakukan.

1. Instalasi paket MySQL. Secara default paket MySQL sudah disertakan pada proses instalasi Linux Slackware, namun instalasi MySQL bisa dilakukan secara manual.

```
# mount /mnt/cdrom/  
# cd /mnt/cdrom/  
# installpkg /mnt/cdrom/slackware/ap/mysql-5.0.24a-i486-1.tgz
```

2. Setelah proses instalasi selesai, lanjutkan dengan proses konfigurasi.

```
# su mysql  
$ mysql_install_db
```

Kembali ke shell root dengan mengetikkan exit atau tekan Ctrl+D.

```
# chown -R mysql:mysql /var/lib/mysql
```

3. Jalankan servis MySQL

```
# chmod +x /etc/rc.d/rc.mysql  
# /etc/rc.d/rc.mysql start
```

4. Selanjutnya adalah proses instalasi MTA Postfix. Dalam penelitian ini penulis menggunakan versi postfix-2.3.7.tar. Sebelumnya letakkan file master Postfix di direktori /usr/src/. Sebelum melakukan instalasi Postfix perlu dipastikan bahwa tidak ada MTA yang lain yang berjalan pada sistem. Langkah pertama adalah membuat sebuah *account* dan *group* yang diperlukan untuk menjalankan *daemon* Postfix.

```
# groupadd postfix  
# useradd -c "Postfix Daemon User" -d /dev/null -g postfix \  
-s /bin/false -u 932 postfix  
# groupadd postdrop
```

5. Lanjutkan dengan proses ekstraksi file instalasi Postfix.

```
# cd /usr/src/
```

```
# tar -xvzf postfix-2.3.7.tar.gz
```

6. Lanjutkan dengan proses instalasi Postfix.

```
# make makefiles \
```

```
CCARGS='-DUSE_TLS -DUSE_SASL_AUTH -
```

```
DDEF_DAEMON_DIR=\"/usr/lib/postfix\" \
```

```
-DDEF_MANPAGE_DIR=\"/usr/share/man\" -I/usr/include/openssl -
```

```
I/usr/include/sasl' \
```

```
AUXLIBS='-L/usr/lib -lssl -lcrypto -lsasl2' \
```

```
CCARGS='-DHAS_MYSQL -I/usr/include/mysql' \
```

```
AUXLIBS='-L/usr/lib -lmysqlclient -lz -lm'
```

```
# make
```

```
# make install
```

Tunggu sampai proses instalasi selesai.

7. Setelah proses instalasi selesai, langkah selanjutnya adalah menset-up file aliases dan diikuti dengan membuat file DBM-nya.

```
# cp /etc/postfix/aliases /etc/
```

```
# newaliases
```

8. Selanjutnya melakukan editing file konfigurasi Postfix. File konfigurasi Postfix pada umumnya terdiri dari dua file yaitu main.cf dan master.cf yang masing-masing terdapat pada direktori /etc/postfix.

```
# pico /etc/postfix/main.cf
```

Ubah file konfigurasi tersebut sehingga menjadi seperti berikut.

```
queue_directory = /var/spool/postfix
```

```
command_directory = /usr/sbin
```

```
daemon_directory = /usr/libexec/postfix
```

```
mail_owner = postfix
```

```
myhostname = firdaus.net
```

```
mydomain = firdaus.net
```

```
myorigin = $mydomain
```

```
inet_interfaces = all
```

```
mydestination = $myhostname, localhost.$mydomain
```

```

unknown_local_recipient_reject_code = 450
mynetworks_style = host
relay_domains =
transport_maps = hash:/etc/postfix/transport
local_transport = local

alias_maps = hash:/etc/aliases
alais_database = hash:/etc/aliases
mail_spool_directory = /var/spool/mail
smtpd_banner = $myhostname ESMTP $mail_name
local_destination_concurrency_limit = 2

default_destination_concurrency_limit = 20
debug_peer_level = 2

debugger_command =
    PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin
    xxd $daemon_directory/$process_name $process_id & sleep 5
sendmail_path = /usr/sbin/sendmail
newaliases_path = /usr/bin/newaliases
mailq_path = /usr/bin/mailq
setgid_group = postdrop
html_directory = /usr/doc/postfix/html
manpage_directory = /usr/share/man
sample_directory = /etc/postfix/samples
readme_directory = /etc/postfix/readme

```

9. Selanjutnya adalah melakukan konfigurasi file master.cf sehingga menjadi seperti berikut.

```

#=====
# service tipe privat unpriv  chroot wakeup maxproc
perintah+arg
#          (ya)   (ya)      (ya) (jangan) (100)
#=====
smtp      inet  n       -       y       -       -       smtpd
#628     inet  n       -       y       -       -       qmqpd
pickup   fifo  n       -       y       60      1       pickup
cleanup  unix  n       -       y       -       0
cleanup
qmgr     fifo  n       -       y       300    1       qmgr

```

```

rewrite    unix    -    -    y    -    -
trivial-rewrite
bounce     unix    -    -    y    -    0    bounce
defer     unix    -    -    y    -    0    bounce
trace     unix    -    -    y    -    0    bounce
verify    unix    -    -    y    -    1    verify
flush     unix    n    -    y    1000?  0    flush
proxymap  unix    -    -    y    -    -
proxymap
smtp       unix    -    -    y    -    -    smtp
relay     unix    -    -    y    -    -    smtp
showq     unix    n    -    y    -    -    showq
error     unix    -    -    y    -    -    error
local     unix    -    n    n    -    -    local
virtual   unix    -    n    n    -    -
virtual
lmtpl     unix    -    -    y    -    -    lmtpl

maildrop  unix    -    n    n    -    -    pipe
  flags=DRhu user=vmail argv=/usr/local/bin/maildrop -d
  ${recipient}
old-cyrus unix    -    n    n    -    -    pipe
  flags=R user=cyrus argv=/cyrus/bin/deliver -e -m ${extension}
  ${user}
cyrus     unix    -    n    n    -    -    pipe
  user=cyrus argv=/cyrus/bin/deliver -e -r ${sender} -m
  ${extension} ${user}
uucp      unix    -    n    n    -    -    pipe
  flags=Fqhu user=uucp argv=uux -r -n -z -a$sender -
  $nexthop!rmail ($recipient)
ifmail    unix    -    n    n    -    -    pipe
  flags=F user=ftn argv=/usr/lib/ifmail/ifmail -r $nexthop
  ($recipient)
bsmtp     unix    -    n    n    -    -    pipe
  flags=Fq. user=foo argv=/usr/local/sbin/bsmtp -f $sender
  $nexthop $recipient

```

10. Selanjutnya adalah mengedit file transport yang terdapat pada `/etc/postfix/transport`.

```
# pico /etc/postfix/transport
```

File ini berperan mengidentifikasi kemana merutekan email yang masuk. Server ini akan menerima semua mail untuk domain `firdaus.net` dan merutekannya keserver pertukaran (*exchange server*), maka perlu ditambahkan baris berikut:

```
firdaus.net smtp:[192.168.0.1]
```

Ini akan meyakinkan bahwa server dapat berhasil melakukan *lookup record* DNS mesin bersangkutan.

- Langkah selanjutnya adalah menjalankan perintah *postmap* yang akan mengkonversi file tersebut ke dalam format database.

```
# postmap /etc/postfix/transport
```

- Kemudian jalankan servis Postfix.

```
# chmod +x /etc/rc.d/rc.postfix
```

```
# /etc/rc.d/rc.postfix start
```

Agar service Postfix dapat langsung dijalankan setiap kali sistem *reboot*, maka diberikan perintah:

```
# echo "/usr/sbin/postfix start" >> /etc/rc.d/rc.local
```

- Tahap akhir dari proses instalasi Postfix adalah melakukan pengujian. Hal ini digunakan untuk memastikan bahwa user dapat terhubung ke server SMTP pada port 25.

```
# telnet localhost 25
```

```
Trying 127.0.0.1...
```

```
Connected to localhost.
```

```
Escape character is '^]'.
```

```
220 server.firdaus.net ESMTP Postfix
```

Jika memberikan respon seperti diatas, maka sistem berhasil. Tekan Ctrl+] dan ketikkan *quit* untuk keluar.

4.3.5 Instalasi dan Konfigurasi IMAP server (UW IMAP)

IMAP merupakan protokol yang digunakan agar *webmail* sebagai MUA dapat berhubungan dengan sistem dalam pengelolaan email. Dalam penelitian ini penulis menggunakan UW IMAP sebagai standar untuk Linux Slackware. Paket ini juga sudah di sertakan pada CD intalasinya. Disini penulis menggunakan versi `imapd-4.64-i486-3.tgz`. Berikut adalah langkah-langkah instalasi dan konfigurasi UW IMAP. Berikut adalah langkah-langkah dilakukan.

1. Lakukan instalasi dari CD instalasi Slackware 11.

```
# mount /mnt/cdrom/  
# cd /mnt/cdrom/  
# installpkg /mnt/cdrom/slackware/n/imapd-4.64-i486-3.tgz
```

2. Pastikan pada file `/etc/services` terdapat baris berikut:

```
pop3      110/tcp      pop-3  
pop3      110/udp      pop-3
```

3. Edit file `/etc/inetd.conf`

```
# pico /etc/inetd.conf
```

Tambahkan baris-baris berikut

```
pop      stream tcp    nowait root    /usr/sbin/ipop2d    ipop2d  
pop3     stream tcp    nowait root    /usr/sbin/ipop3d    ipop3d  
imap     stream tcp    nowait root    /usr/sbin/imapd     imapd
```

4. Restart service `inetd`

```
# chmod +x /etc/rc.d/rc.inetd  
# /etc/rc.d/rc.inetd restart
```

5. Lakukan pengujian pada service tersebut.

```
# telnet localhost 110  
Trying 127.0.0.1...  
Connected to localhost.  
Escape character is '^]'.  
+OK POP3 localhost 2004.89 server ready  
user yudi  
+OK User name accepted, password please
```



```
pass yudi
+OK Mailbox open, 4 messages
```

Jika respon yang diperoleh seperti diatas, berarti konfigurasi sudah berhasil, untuk keluar tekan tombol Ctrl+] dan berikan perintah *quit*.

4.3.6 Instalasi dan Konfigurasi UebiMiau

Pada pembangunan *mail server* ini, sebagai *webmail* menggunakan aplikasi UebiMiau. Disini penulis menggunakan versi uebimiau-2.7.2-any.zip. Berikut adalah langkah-langkah dalam melakukan instalasi dan konfigurasi UebiMiau.

1. Salin file master UebiMiau ke *document root* untuk *virtual host* yang menangani aplikasi *email client* yang akan di akses dari *client* melalui *web browser* sebagai mail.firdaus.net

```
# cp uebimiau-2.7.2-any.zip /home/firdaus.net/www
```

2. Ekstrak file master UebiMail.

```
# cd /home/firdaus.net/www/mail
```

```
# unzip uebimiau-2.7.2-any.zip
```

Karena direktori penyimpanan file webmail didefinisikan sebagai /mail dalam file konfigurasi *virtual host*- nya, sehingga perlu disalin nama folder *default*- nya.

```
# mv uebimiau-2.7.2-any/ mail
```

3. Buat folder yang akan menampung *file temporary*, kemudian berikan *write-permission* pada folder tersebut. Untuk keluarga Unix dan Linux umumnya terdapat pada folder /tmp/uebimiau.

```
# mkdir /tmp/uebimiau
```

```
# chmod 755 /tmp/uebimiau
```

4. Langkah selanjutnya adalah melakukan konfigurasi UebiMiau yang berada pada file inc/config.php.

```
# cd mail/inc/config.php
```

```
# pico config.php
```

Edit file tersebut dengan mengubah beberapa parameter yang ada dan pastikan nilainya adalah seperti berikut:

```
define("yes",1);
define("no",0);
$themes          = Array();
$languages       = Array();
```

Berikut konfigurasi untuk menentukan folder penyimpanan *file temporary* sesuai dengan yang telah dibuat sebelumnya.

```
$temporary_directory = "/tmp/uebimiau";

$smtp_server = "localhost"; #YOU NEED CHANGE IT !!
$mail_protocol = "pop3";
$mail_port = 110;
$server_time_zone = "-0000";
```

Konfigurasi berikut untuk menentukan *space* hardisk yang diberikan untuk menyimpan email.

```
$quota_limit = 2048; // in KB, eg. 4096 Kb = 4MB

$use_password_for_smtp = no;
$check_first_login = yes;
$allow_html = yes;
$allow_scripts = no;
$allow_modified_from = yes;
$mail_server_type = "ONE-FOR-EACH";
$mail_detect_remove = "www.";
$mail_detect_prefix = "mail.";
$mail_detect_login_type = "%user%@%domain%";
```

Konfigurasi inisialisai *mail server*.

```
$mail_servers[] = Array(
    "domain" => "firdaus.net",
    "server" => "server.firdaus.net",
    "login_type" => "%user%"
);
```

```

$default_mail_server = "server.firdaus.net";
$one_for_all_login_type = "%user%@%domain%";
require("../inc/config.languages.php");
$use_sendmail = no;
$path_to_sendmail = "/usr/sbin/sendmail";
$mail_use_top = yes;
$appversion = "2.7.2";
$appname = "UebiMiau";
$footer = "

```

Message sent using \$appname \$appversion

```

";
$enable_debug = no;
$block_external_images = no;
$idle_timeout = 10; //minutes
$default_sortby = "date";
$default_sortorder = "DESC";
$default_preferences = Array(
    "send_to_trash_default" => yes,
    "st_only_ready_default" => yes,
    "save_to_sent_default" => yes,
    "empty_trash_default" => yes,
    "sortby_default" => "date",
    "sortorder_default" => "DESC",
    "rpp_default" => 20,
    "add_signature_default" => no,
    "signature_default" => "",
    "timezone_default" => "+0000",
    "display_images_deafult" => yes,
    "editor_mode_default" => "html",
    "refresh_time_default" => 10
);

```

5. Setelah semua proses instalasi dan konfigurasi *webmail* UebiMiau selesai, selanjutnya lakukan tes dengan mengakses alamat <http://mail.firdaus.net>

4.3.7 Uji coba

Langkah terakhir adalah melakukan uji coba sistem yang telah dibangun. Berikut adalah langkah-langkah yang dilakukan untuk pengujian sistem.

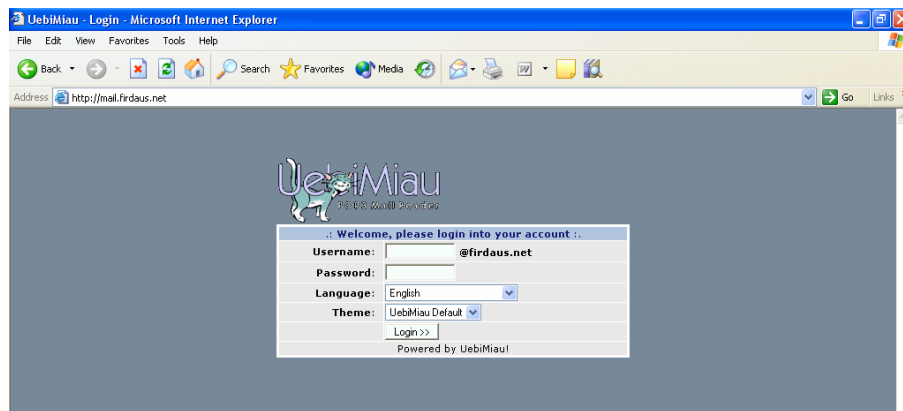
1. Buat sebuah account user melalui sisi server.

```
# useradd -m yudi
```

```
# passwd yudi
```

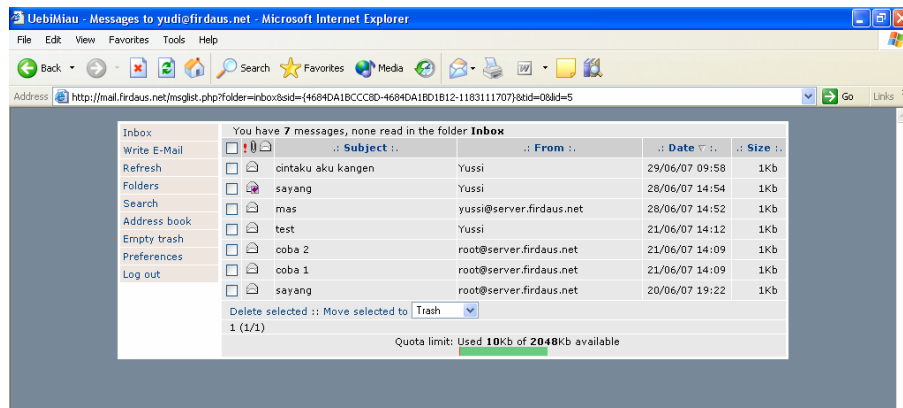
Masukkan password, misalnya “yudi”

2. Pastikan service Apache, BIND, MySQL, dan Postfix sudah dijalankan. Selanjutnya akses alamat *webmail* pada alamat <http://mail.firdaus.net> melalui *web browser*.



Gambar 4.21 Tampilan halaman Login UebiMiau

3. Masukkan *username* dan *password* yang telah dibuat sebelumnya kemudian klik Login.
4. Jika berhasil akan ditampilkan halaman utama dari aplikasi UebiMiau.



Gambar 4.22 Tampilan halaman utama UebiMiau

5. Selanjutnya proses manajemen email dari sisi *client* dapat dilakukan. Dengan demikian proses pembangunan mail server berbasis Linux menggunakan Postfix telah selesai.

4.4 Pengujian Keseluruhan Sistem

Sebagai tahap akhir dari seluruh rangkaian penelitian yang dilakukan adalah tahap evaluasi atau pengujian keseluruhan sistem yang telah dibangun. Sistem mail server yang dibangun pada sistem operasi Linux untuk varian Slackware 11 yang notabeneanya memiliki catatan yang baik dalam hal kehandalan dan kestabilan menangani fungsi-fungsi server. Dengan mengimplementasikan Postfix sebagai *mail server* yang bertugas menangani manajemen pengiriman dan penerimaan email dalam jaringan komputer. Penerapkan sistem *mail client* berbasis web untuk memudahkan user dalam mengakses pesan yang dikirimkan untuknya, mensyaratkan adanya web server yang menangani proses interaksi antara user dengan sistem melalui *interface* web, dan DNS server yang memungkinkan user untuk menggunakan sebuah nama untuk mengidentifikasi suatu alamat IP server, dalam hal ini server yang dibangun menggunakan IP 192.168.0.1 netmask 255.255.255.0 dengan menggunakan *domain name* firdaus.net, sehingga untuk dapat mengakses server, user tidak perlu bersusah payah untuk menghafal nomor IP tersebut, cukup menggunakan nama domain yang lebih mudah untuk diingat. Pada aplikasinya, sistem ini menerapkan Apache sebagai web server dan BIND sebagai DNS server.

Sebagai aplikasi *mail client* berbasis web, pada sistem ini digunakan aplikasi UebiMiau sebagai aplikasi *webmail* yang sudah cukup terkenal di kalangan pengguna Linux.

Dalam pelaksanaannya, pembuatan sistem *mail server* ini mengalami beberapa kendala, diantaranya:

1. Minimnya tutorial mengenai penerapan distribusi *mail server* Postfix pada Linux Slackware menjadi salah satu kendala dalam mencari referensi untuk membangun sistem ini.
2. Minimnya hardware yang digunakan, sehingga menyulitkan dalam proses uji coba.

BAB V

PENUTUP

5.1 Kesimpulan

Setelah melaksanakan penelitian tugas akhir dengan membangun sebuah *mail server* berbasis Linux khususnya distro Slackware 11 dengan menggunakan Postfix sebagai MTA (*Mail Transfer Agent*), dapat di ambil kesimpulan sebagai berikut:

1. Telah dibangun sistem *mail server* berbasis Linux dengan menggunakan Postfix yang bertugas menangani lalu lintas penerimaan dan pengiriman email pada suatu jaringan komputer, sehingga dapat memberikan kemudahan untuk berkorespondensi antar *user* secara *virtual* melalui *intranet*.
2. Postfix merupakan salah satu aplikasi MTA (*Mail Transfer Agent*) yang *free* untuk *platform* Linux yang dirancang untuk lebih cepat, lebih mudah dirawat, dan lebih aman.
3. Postfix merupakan MTA yang dirancang sebagai “*sendmail-compatible*”, untuk lebih memudahkan user melakukan migrasi dari sistem Sendmail ke sistem Postfix.
4. Postfix merupakan MTA yang diakui tiga kali lebih cepat dibanding kompetitor utamanya, Qmail.
5. Postfix menggunakan pertahanan *multilayer* untuk melindungi sistem lokal dari gangguan *intruder*.

5.2 Saran

Beberapa hal yang perlu dikembangkan dalam pembangunan mail server untuk lebih meningkatkan performa sistem terutama dalam hal keamanan sistem, apalagi untuk sistem yang berhubungan langsung dengan internet, diantaranya:

1. Server email merupakan sebuah sistem utama yang menangani proses pengiriman dan penerimaan email dalam sebuah jaringan komputer perlu pengelolaan khusus untuk menjaga stabilitas dan performa dari sistem.
2. Dalam pembangunan sebuah sistem *mail server*, apalagi yang berhubungan langsung dengan internet, perlu dirancang khusus untuk sistem keamanannya, baik sistem autentikasi maupun implementasi antivirus untuk menghalangi serangan virus yang menyusup melalui pesan yang dikirimkan.
3. Sebuah aplikasi user interface yang menarik dan mampu memberikan beragam fasilitas yang dapat memberikan kemudahan bagi user akan memberikan nilai lebih disisi *client* terhadap sistem.

DAFTAR PUSTAKA

- Husni. 2003. *Implementasi Jaringan Komputer dengan Linux Redhat 9*. ANDI Yogyakarta: Yogyakarta.
- Nugraha, Dindin. 2005. *Desknow Mail Server Mudah dan Serbaguna*. Majalah Info Linux edisi Desember 2005: Jakarta.
- Nursyamsu, Arief Fajar. 2002. *Pengantar Internet*. Program Pendidikan Teknologi Informasi Universitas Sebelas Maret: Surakarta.
- Rafiudin, Rahmat. 2006. *Membangun Server E-Mail Berbasis FreeBSD/Linux*. ANDI: Yogyakarta.
- Wiryana.S. Si, SKom, M.Sc, I Made, dan Wicaksana, S.Si, M.Eng, I Wayan. 2002. *Membangun server dengan Open Source*. www.ilmukomputer.com: Jakarta.
- <http://dadanhatori.net/?p=6> , 06-06-2007,13:47
- <http://gudanglinux.net/gmc/content/view/278/2/>, 06-06-2007,13:48
- <http://jamm.sourceforge.net/howto/html/architecture.html>, 15-06-2007
- <http://ms.wikipedia.org/wiki/Linux>, 06-06-2007,13:48
- <http://ms.wikipedia.org/wiki/Slackware> ,06-06-2007,13:48
- <http://postfix.vlsm.org/website/basic.html>, 15-06-2007
- <http://vhcs.hostinghacks.net/slackware/> , 12-06-2007 12:49
- <http://workaround.org/articles/ispmail-sarge/>, 15-06-2007
- <http://www.cs.nott.ac.uk/TSG/IMAP/imapd.html>, 15-06-2007
- <http://www.gudanglinux.or.id/content1slack.html>, 06-06-2007,13:48
- <http://www.pocock.com.au/linux-doc/mbox-maildir.html>, 15-06-2007
- <http://www.postfixvirtual.net/>, 12-06-2007 13:45
- <http://www.squirrelmail.org/wiki/SpeedWithUW>, 15-06-2007
- <http://www.washington.edu/imap/documentation/CONFIG.html>, 15-06-2007