

Membangun Web Server (Apache+PHP+MySQL) menggunakan FreeBSD

Pendahuluan.

Internet sudah menjadi bagian dari kehidupan manusia saat ini. Internet merupakan sumber data dan informasi yang sangat lengkap. Internet dapat juga digunakan sebagai media promosi yang relatif murah. Oleh karena itu suatu perusahaan atau instansi yang ingin dikenal oleh masyarakat luas dan dunia, dapat menggunakan internet. Untuk mewujudkan hal tersebut, maka diperlukan sebuah perangkat khusus, yaitu *Web Server*.

Sebagai *hardware*, *Web Server* merupakan sebuah komputer yang didesain khusus sebagai server oleh suatu perusahaan pembuat komputer (misalnya IBM, HP, Dell) atau sebuah komputer biasa yang disetel sebagai server. Komputer biasa disini berarti sebuah komputer yang dirakit sendiri atau dirakit oleh suatu toko komputer lokal.

Untuk menjadi *Web Server*, komputer tersebut harus dipasang suatu Sistem Operasi server dan *software-software* aplikasi yang mendukung terbentuknya suatu *Web Server*.

Sistem Operasi server yang digunakan adalah FreeBSD dan *software* aplikasi yang mendukung terbentuknya *Web Server* adalah **Apache**. Disamping itu, diperlukan juga *software database* sebagai pengelolaan datanya, yaitu **MySQL**.

Makalah ini memanfaatkan *software-software* yang dapat didownload langsung dan *legal* tanpa harus membayar kepada perusahaan pembuatnya karena *software-software* tersebut bersifat *open source*.

Dalam membangun sebuah *Web Server*, harus pula dibangun layanan-layanan pendukungnya. Layanan-layanan pendukung ini sebenarnya adalah suatu *software* aplikasi server juga, hanya saja letaknya secara fisik jadi satu dengan *Web Server*. Layanan-layanan tersebut adalah :

- FTP (*File Transfer Protocol*)
- SSH (*Secure Shell*)
- PHP *interpreter*
- *Database*

Beberapa Pengertian Dasar.

a. *FreeBSD*.

Menurut situs www.freebsd.org, FreeBSD merupakan suatu *project* yang bertujuan untuk menyediakan suatu *software* yang dapat digunakan untuk masyarakat banyak tanpa adanya suatu kompensasi yang harus dibayar oleh masyarakat tersebut.

FreeBSD merupakan turunan dari UNIX versi 4.4.BSD-Lite untuk komputer Intel (x86), DEC Alpha, dan Sun Ultra SPARC. FreeBSD dibuat oleh Computer Systems Research Group (CSRG) di *University of California at Berkeley*.

Menurut *FreeBSD Handbook* (2003), FreeBSD dikatakan sebagai Sistem Operasi server karena mempunyai fitur-fitur seperti berikut ini:

- *Preemptive Multitasking.*
Merupakan pembagian kerja proses yang dapat disela, sehingga mengakibatkan *sharing* sumberdaya komputer antar aplikasi maupun antar *user* terjadi dengan adil dan baik, walaupun sistem sedang dalam beban yang berat dan penuh.
- *Multi-user facilities.*
Fasilitas ini memungkinkan banyak *user* menggunakan sistem FreeBSD secara bersama-sama dan bersamaan untuk mengerjakan hal yang saling berbeda.
- *TCP/IP Networking.*
TCP/IP Networking di FreeBSD mendukung standar-standar industri, misalnya DHCP dan NFS. FreeBSD juga mendukung penomoran IP versi 6. Ini berarti sistem FreeBSD dapat beroperasi lintas sistem dengan mudah dan tetap dapat digunakan ketika IP versi 6 harus digunakan. Sehingga FreeBSD dapat digunakan untuk enterprise server yang menyediakan layanan vital seperti WWW, FTP, *routing*, dan *firewall*.
- *Memory Protection.*
Menjamin bahwa aplikasi atau *user* tidak dapat saling menginterferensi satu dengan lainnya. Ini berarti, satu aplikasi yang *crash* tidak akan mempengaruhi aplikasi lainnya.
- *Mendukung Symetric Multi Processor.*
Mendukung komputer yang mempunyai lebih dari satu prosesor yang simetris.
- *Binary Compatibility.*
Kompatibilitas dengan program-program lain yang dibuat untuk Linux, SCO, SVR4, BSDI, dan NetBSD.

Pada dasarnya, FreeBSD bekerja dengan banyak *Daemon*. *Daemon* merupakan *software* atau modul kecil yang residen di memori untuk menjalankan layanan tertentu. Menurut situs www.freebsd.org, ucapan kata *daemon* sering disalahartikan menjadi setan sehingga logo FreeBSD adalah setan kecil.

Menurut *FreeBSD Handbook* (2003), perusahaan-perusahaan besar yang ada di internet dengan menggunakan FreeBSD antara lain adalah:

- Yahoo! (www.yahoo.com)
- Apache (www.apache.org)
- Netcraft (www.netcraft.com)
- Sony Japan (www.sony.co.jp)
- Sophos Anti-Virus (www.sophos.com)

b. *Web Server.*

Halaman-halaman *web* yang diakses melalui *web browser* seperti Internet Explorer atau Netscape, sebenarnya terletak di suatu server yang disebut dengan *Web Server*. Dengan kata lain, *Web Server* adalah suatu server yang menyimpan halaman-halaman *web* dari suatu instansi atau perusahaan tertentu yang dapat diakses dengan *web browser*. *Web Server* ini diciptakan

oleh suatu *software* aplikasi, salah satunya adalah **Apache**. *Software* ini yang akan digunakan pada makalah.

Apache berkembang dengan cepat dan banyak digunakan karena mendukung bahasa *scripting*, seperti PERL dan PHP, dan juga merupakan produk *open source* yang notabene berharga murah.

Apache bekerja dengan menggunakan *Hyper Text Transfer Protocol*. Sehingga **Apache** sering disebut dengan **httpd**, yang berarti *daemon http Protocol* ini bekerja di *port* 80, 81, dan 8080. Pada makalah ini, *port* yang akan digunakan adalah *port* 80 yang merupakan *port default Apache*.

Web server dapat diukur kinerjanya walaupun ukuran kinerja tersebut tidak benar-benar tepat, tetapi sudah dapat mewakili. Ukuran kinerja *web server* adalah sebagai berikut:

- Banyaknya *user* yang dapat dilayani dalam satu waktu.
- Keandalan dan kestabilan dalam melayani *user*.
- Banyaknya proses yang dapat diciptakan dan diselesaikan dalam satu waktu.

c. **FTP Server.**

FTP (*File Transfer Protocol*) merupakan suatu *protocol* yang digunakan untuk transfer data secara *remote*. Layanan ini sangat berguna dan banyak digunakan di internet selain e-mail, telnet, dan SSH.

Kegunaan utama layanan ini pada *Web Server* adalah untuk meng-*update* halaman *web* yang ada di *Web Server* tersebut, karena peng-*update*-an isi halaman *web* ini biasanya dilakukan dari komputer lain.

FTP menggunakan dua *port* untuk bekerja, yaitu *port* 20 untuk mengirim data yang dibutuhkan, dan *port* 21 untuk perintah-perintah kontrol dan status informasi.

FTP ini biasanya sudah ter-*install* secara *default* dari Sistem Operasi FreeBSD, sehingga tidak diperlukan lagi *software* pihak ketiga.

d. **SSH.**

SSH (*Secure Shell*) merupakan suatu *protocol login* yang *text mode*. Artinya *administrator* dapat login dan bekerja dengan servernya tanpa harus berada di depan server tersebut, hal ini disebut dengan *remote login* atau *remote administration*.

SSH disebut *secure* karena menggunakan enkripsi untuk semua data yang dikirim maupun diterima ketika terhubung dengan server. Oleh karena itu dapat melindungi *password* atau data-data sensitif lainnya dari penyadapan jaringan, sehingga SSH banyak digunakan untuk *remote administration*, jika *administrator* tidak berada di depan server. SSH bekerja menggunakan *port* 22.

SSH ini biasanya sudah ter-*install* secara *default* dari Sistem Operasi FreeBSD, sehingga tidak diperlukan lagi *software* pihak ketiga.

e. **PHP.**

Pada awalnya PHP merupakan singkatan dari *Personal Home Page tools*. PHP sebenarnya merupakan suatu *script* yang bersifat *server-side* yang ditambahkan ke dalam file HTML. *Script* ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML sehingga suatu halaman web menjadi

dinamis dan interaktif. Sifat *server-side* berarti pengerjaan *script* dilakukan di server, kemudian hasilnya dikirimkan ke *browser*. Karena sifat inilah, maka suatu *web server* harus mendukung PHP. Keuntungan sifat *server-side* adalah:

- Tidak harus menggunakan *browser* tertentu, karena server yang akan mengerjakan *script* PHP tersebut. Hasil yang dikirimkan umumnya berupa teks atau gambar yang dapat dikenali oleh *browser* apapun.
- Dapat memanfaatkan aplikasi lain yang dimiliki oleh server, misalnya koneksi ke *database*.
- *Script* dapat dilihat tetapi tidak dapat diketahui secara jelas dengan menggunakan fasilitas *view source* yang ada di *browser*, sehingga ada proteksi keamanan disana.

f. **Database Server.**

Suatu halaman *web* yang interaktif akan memberikan kesempatan pada *user* untuk mengisikan data, misalnya halaman *web* buku tamu. Data-data yang sudah diisikan oleh *user* tersebut akan disimpan oleh *Web Server* dengan menggunakan *software database*. Oleh karena itu dalam suatu *Web Server*, sebaiknya juga dibangun suatu *Database Server* sebagai pendukungnya.

Salah satu *software* yang banyak digunakan untuk membangun *Database Server* adalah **MySQL**. *Software* ini yang akan digunakan pada makalah.

MySQL berkembang dengan cepat karena mendukung pemrograman dengan bahasa SQL yang notabene mudah dan *powerfull*, serta merupakan produk *open source* sehingga berharga murah.

Obyek makalah adalah FreeBSD versi 5.1 beserta *software-software* yang diperlukan dan satu unit komputer. Penentuan obyek ini karena belum ada buku atau referensi lain yang membahas tentang membangun *Web Server* dengan FreeBSD secara jelas dan mendalam. Spesifikasi *Hardware* yang digunakan adalah :

Processor Intel Pentium III 600 MHz

Main Board ASUS chipset VIA

RAM 128 MB

AGP 4MB

Sound Card Yamaha PCI

HDD 20 GB

CDROM 52x

LAN Card PCI Realtek 3189

Pengujian Web Server.

Pengujian dilakukan dengan *Web Server* yang sudah berisi *content web* beserta data-data pendukungnya. Pengujian ini menggunakan suatu *software* yang *download* dari Internet dan dijalankan dari *client*.

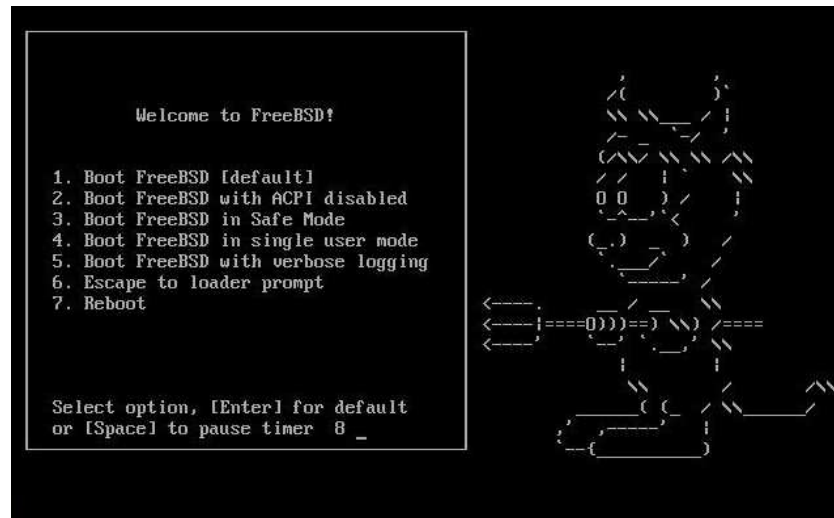
Software tersebut adalah **Webstress versi 6.0** dari Paessler Software Solution untuk *client* berbasis Windows dan **httperf versi 0.8** yang dibuat oleh David Mosberger dari Hewlett-Packard untuk *client* berbasis Linux.

Metode ini menguji *Web Server* tersebut dengan simulasi banyaknya *user* yang terhubung dan dilayani. Kemudian dihitung pula banyaknya kesalahan layanan serta kecepatan layanan dalam satuan detik.

Instalasi FreeBSD versi 5.1-RELEASE.

Makalah ini dimulai dengan penginstallan Sistem Operasi FreeBSD versi 5.1-RELEASE. Media sumber Sistem Operasi ini adalah CDROM yang dibeli dari sebuah toko buku di Yogyakarta. Sebenarnya sumber instalasi dapat juga *download* dari situs resminya, yaitu www.freebsd.org, tetapi tidak dapat dilakukan, karena keterbatasan *bandwith* yang ada. Sumber instalasi ini berkapasitas 640MB. Berikut ini langkah-langkah instalasi FreeBSD:

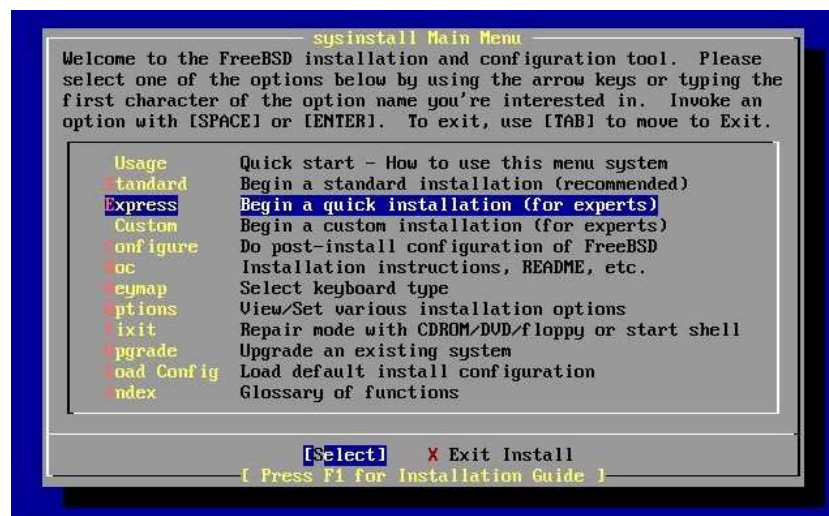
- a. *Booting* dengan CD FreeBSD versi 5.1-RELEASE, kemudian akan muncul tampilan sebagai berikut:



Gambar 1. Menu awal FreeBSD.

Kemudian dipilih nomor 1 (*default*) atau menekan tombol Enter supaya proses *booting* dan instalasi berlanjut.

- b. Setelah proses *booting* selesai, maka akan muncul menu utama instalasi untuk memilih metode instalasi dan keperluan lain. Tampilannya sebagai berikut:



Gambar 2. Menu Utama instalasi.

Dipilih metode *Express*, supaya instalasi berlangsung cepat dan tidak terlalu banyak pilihan.

- c. Setelah itu mengatur partisi *Harddisk*. Pada instalasi ini, keseluruhan kapasitas *Harddisk* akan digunakan. Tampilannya sebagai berikut:

```

Disk name:      ad0      FDISK Partition Editor
DISK Geometry: 33288 cyls/16 heads/63 sectors = 33554304 sectors (16383MB)

Offset      Size(MB)      End      Name  PType      Desc  Subtype  Flags
-----
0           16383      33554303      -    12      unused      0

The following commands are supported (in upper or lower case):

A = Use Entire Disk    G = set Drive Geometry    C = Create Slice    F = 'DD' mode
D = Delete Slice      Z = Toggle Size Units    S = Set Bootable    I = Wizard m.
T = Change Type      U = Undo All Changes    Q = Finish

Use F1 or ? to get more help, arrow keys to select.

```

Gambar 3. Editor partisi *fdisk*.

Kemudian memilih menggunakan keseluruhan *Harddisk* dengan menekan tombol A (*Use Entire Disk*). Pemilihan penggunaan keseluruhan *Harddisk* berarti membuat satu *slice* saja.

```

Disk name:      ad0      FDISK Partition Editor
DISK Geometry: 33288 cyls/16 heads/63 sectors = 33554304 sectors (16383MB)

Offset      Size(MB)      End      Name  PType      Desc  Subtype  Flags
-----
0           0           62      -    12      unused      0
63          16383      33554303      ad0s1  8      freebsd      165

The following commands are supported (in upper or lower case):

A = Use Entire Disk    G = set Drive Geometry    C = Create Slice    F = 'DD' mode
D = Delete Slice      Z = Toggle Size Units    S = Set Bootable    I = Wizard m.
T = Change Type      U = Undo All Changes    Q = Finish

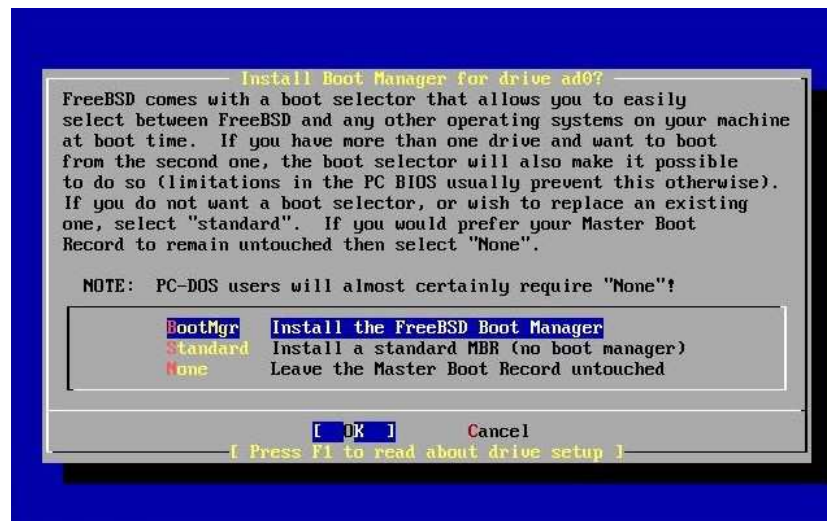
Use F1 or ? to get more help, arrow keys to select.

```

Gambar 4. Menggunakan semua kapasitas yang ada di *Harddisk*.

Setelah selesai dengan konfigurasi ini maka tombol Q (*Finish*) ditekan untuk melanjutkan ke proses berikutnya.

- d. Proses berikutnya adalah instalasi *boot manager*. *Boot Manager* ini harus ada, jika tidak ada maka FreeBSD tidak dapat *booting*. Tampilannya sebagai berikut:



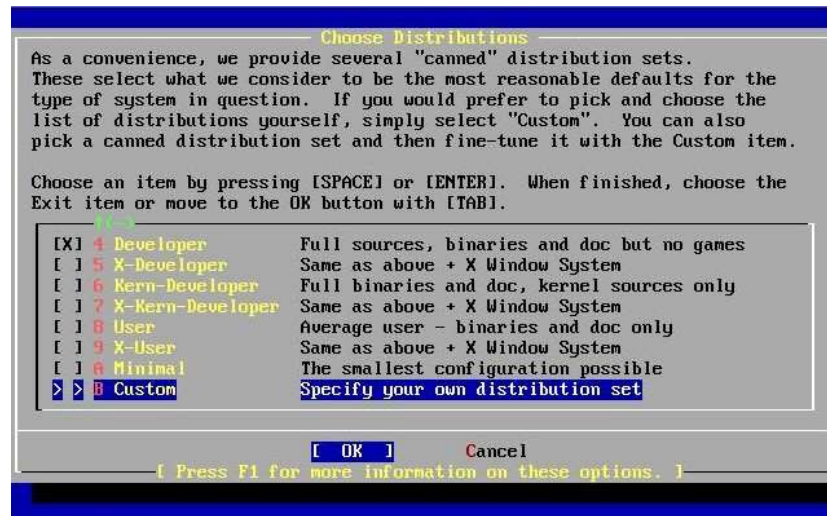
Gambar 5. Instalasi *Boot Manager*.

- e. Kemudian dilanjutkan dengan pembuatan partisi dari *slice* yang sudah ada, untuk memudahkan pembuatan partisi dipilih *Auto Defaults* dengan menekan tombol A. Artinya FreeBSD membuat secara otomatis partisi dengan ukuran yang dibutuhkan sendiri. Tampilannya sebagai berikut:



Gambar 6. Editor pembagian *slice* (*Disklabel*).

- f. Setelah pembuatan partisi selesai, maka dilanjutkan dengan pemilihan komponen *software* yang akan diinstall ke dalam FreeBSD. Di sini penulis memilih sebagai *Developer*, sehingga *software-software* akan diinstallkan secara penuh dengan *sourcenya*, tanpa adanya *software game*. Tampilannya sebagai berikut:



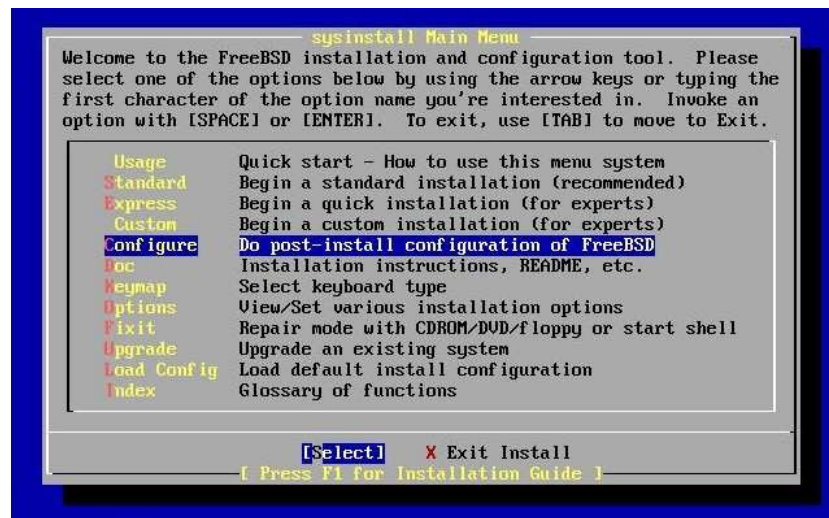
Gambar 7. Pemilihan program yang akan diinstall.

- g. Pilihan berikutnya adalah memilih media sumber untuk instalasinya, seperti pada gambar berikut:



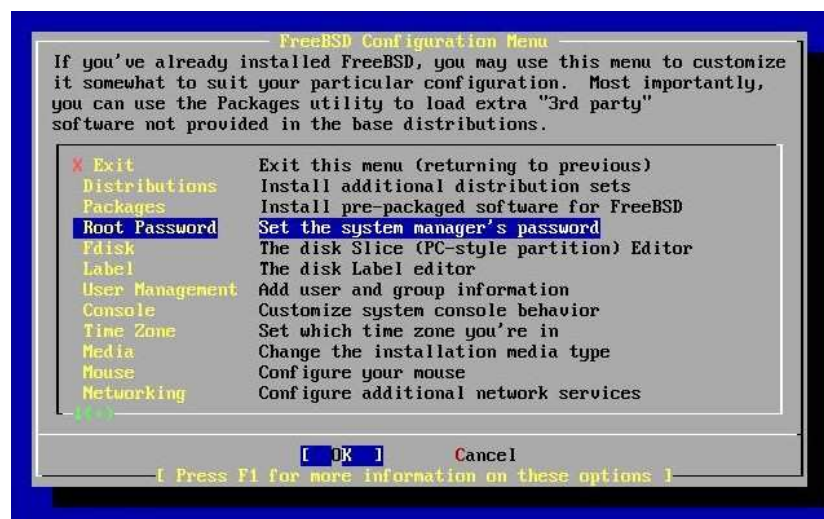
Gambar 8. Pemilihan media sumber.

- h. Setelah pemilihan media sumber untuk instalasinya, maka proses instalasi dimulai.
- i. Setelah instalasi selesai dan tidak ada kesalahan ataupun kerusakan *software* pada saat instalasi, maka proses instalasi awal FreeBSD sudah selesai. Kemudian dilanjutkan dengan instalasi lanjutan (*Post-install*). Tampilannya sebagai berikut:



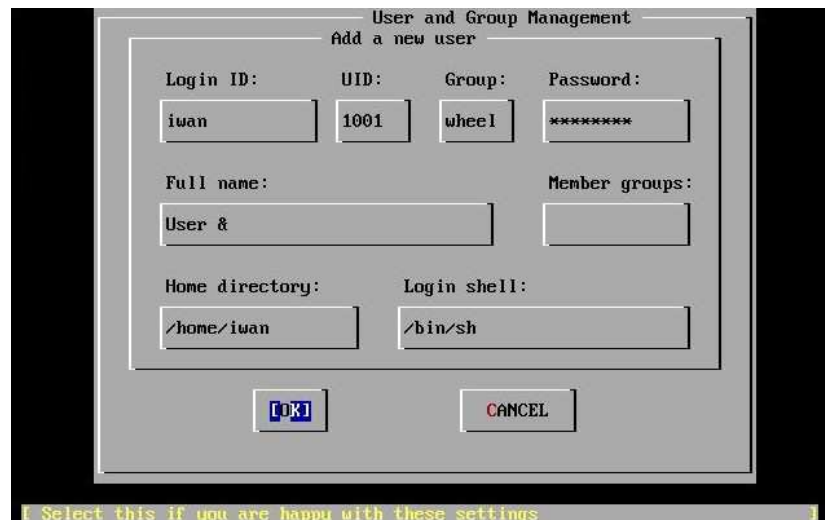
Gambar 9. Post-install configuration.

- j. *Post-install* ini dimulai dengan pembuatan *password* untuk *user root*. Tampilannya sebagai berikut:



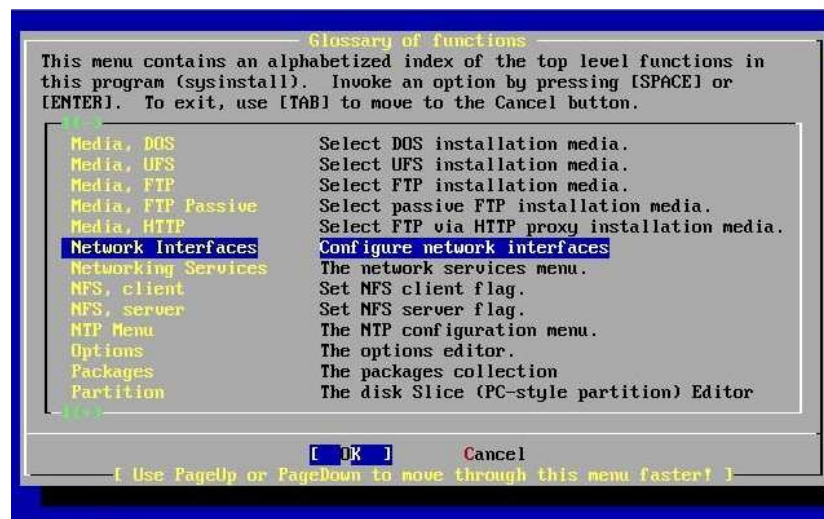
Gambar 10. Pembuatan password *user root*.

- k. Kemudian dilanjutkan dengan pembuatan *user login*, dimana *user* ini akan digunakan sebagai *login* untuk memelihara dan memeriksa *web server* ini. Oleh karena itu, *groupnya* disamakan dengan *group* dari *user root* yaitu **wheel**. Tampilannya sebagai berikut:



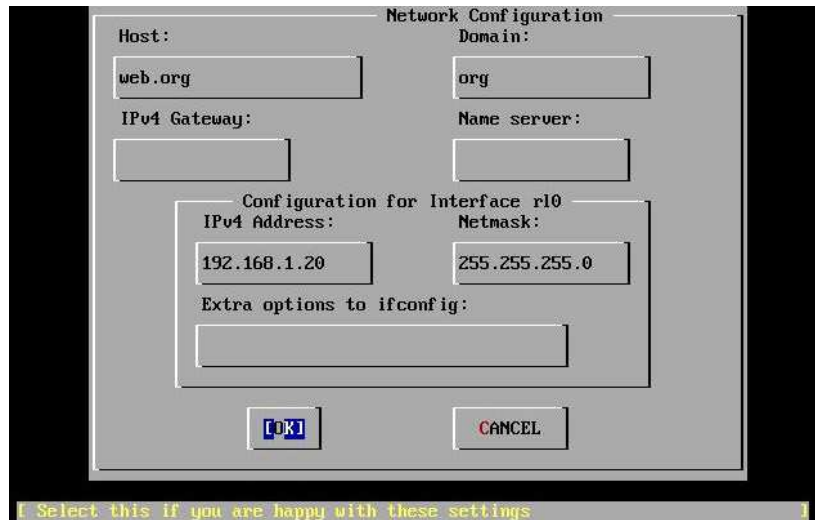
Gambar 11. Membuat *user login*.

- l. Setelah pembuatan *user login* selesai, maka akan kembali ke menu utama. Kemudian dilanjutkan dengan konfigurasi *network* yang ada di server, dengan memilih **Index** dilanjutkan dengan memilih **Network Interface** pada menu yang sudah disediakan, seperti pada tampilan berikut ini:



Gambar 12. Pilihan untuk konfigurasi *Network*.

- m. Pada langkah selanjutnya, FreeBSD sudah mengenali *network card* yang ada, sehingga tinggal dipilih *network card* tersebut, karena menggunakan Realtek, maka terdeteksi sebagai **rl10**.
- n. Kemudian konfigurasi *network* dimulai, dengan memberi nama server tersebut dengan **web.org** dengan IP address adalah 192.168.1.20 netmask 255.255.255.0 yang merupakan nomor IP lokal kelas C. Nomor ini dipilih karena untuk sementara *web server* ini hanya akan diakses secara lokal dan belum dihubungkan dengan internet. Tampilannya sebagai berikut:



Gambar 13. Konfigurasi *Network*.

Jika akan dihubungkan dengan internet, maka IP *address* harus diganti dengan IP *address* publik. IP *address* publik biasanya didapat dari ISP.

- o. Setelah langkah di atas, maka instalasi Sistem Operasi FreeBSD sudah selesai, sehingga dapat dimulai instalasi paket-paket *software* yang mendukung pembuatan *web server*.

Instalasi Apache versi 2.0.52.

Instalasi paket *software* ini dilakukan secara manual, yaitu dengan kompilasi sendiri. FreeBSD secara *default* sudah menyediakan *compilernya*, yaitu *compiler* bahasa C. Penulis tinggal menggunakannya. Langkah-langkah yang digunakan adalah sebagai berikut:

- a. *Download source code Apache* versi 2.0.52 dari situs resminya www.apache.org atau *mirror server* yang disediakan. Nama filenya adalah **httpd-2.0.52.tar.gz**. *Source code* ini masih terkompresi sebagai **.tar.gz**, sehingga perlu diekstrak.
- b. Login sebagai *user root* untuk mengekstrak file tersebut, dengan perintah:

```
# tar -zxvf httpd-2.0.52.tar.gz
```
- c. Setelah terekstrak, maka langkah kompilasi dan instalasi **Apache** dapat dimulai. Langkah pertama adalah masuk ke *directory* hasil ekstraksi; biasanya sama dengan nama filenya tanpa ekstensi (**httpd-2.0.52**), kemudian mengonfigurasi **Apache** supaya diarahkan ke *directory* **/usr/local/apache** yang akan menyimpan *software* ini dan dapat menerima modul PHP. Perintah yang digunakan adalah:

```
# ./configure --prefix=/usr/local/apache --enable-module=so
```
- d. Setelah konfigurasi di atas, maka dilanjutkan dengan kompilasi dan instalasi *software Apache*, dengan perintah:

```
# make; make install
```

- e. Jika tidak ada kesalahan atau *error* pada proses konfigurasi, kompilasi, dan instalasi, maka **Apache** sudah berhasil diinstall. Untuk memeriksa dan mencobanya dilakukan dengan cara:

```
# cd /usr/local/apache/bin
# ./apachectl start
# ps -ax
```

Baris terakhir dari perintah di atas adalah perintah untuk melihat proses yang sedang aktif saat itu. Jika proses **httpd** ada, maka **Apache** berhasil diinstall dan bekerja, seperti ditunjukkan pada gambar berikut:

```
35 ?? IL 0:00.00 (nfsiod 1)
36 ?? IL 0:00.00 (nfsiod 2)
37 ?? IL 0:00.00 (nfsiod 3)
227 ?? Ss 0:02.86 /usr/sbin/syslogd -s
361 ?? Is 0:00.77 /usr/sbin/sshd
366 ?? Ss 0:05.83 sendmail: accepting connections (sendmail)
369 ?? Is 0:00.19 sendmail: Queue runner@00:30:00 for /var/spool/client
383 ?? Ss 0:03.42 /usr/sbin/cron
41918 ?? Ss 0:00.27 /usr/local/apache/bin/httpd -k start
41919 ?? I 0:00.06 /usr/local/apache/bin/httpd -k start
41920 ?? I 0:00.08 /usr/local/apache/bin/httpd -k start
41921 ?? I 0:00.05 /usr/local/apache/bin/httpd -k start
41922 ?? I 0:00.06 /usr/local/apache/bin/httpd -k start
41923 ?? I 0:00.05 /usr/local/apache/bin/httpd -k start
420 ?? Is 0:01.08 /usr/sbin/lpd
428 v0 S 0:02.56 -csh (csh)
41945 v0 R+ 0:00.03 ps -ax
19891 v1 Is+ 0:00.22 /usr/libexec/getty Pc ttyv1
422 v2 Is+ 0:00.17 /usr/libexec/getty Pc ttyv2
423 v3 Is+ 0:00.17 /usr/libexec/getty Pc ttyv3
424 v4 Is+ 0:00.15 /usr/libexec/getty Pc ttyv4
425 v5 Is+ 0:00.15 /usr/libexec/getty Pc ttyv5
426 v6 Is+ 0:00.16 /usr/libexec/getty Pc ttyv6
427 v7 Is+ 0:00.16 /usr/libexec/getty Pc ttyv7
web#
```

Gambar 14. Proses **httpd** sedang aktif.

- f. Untuk menghentikan proses **Apache** yang sedang berjalan, dapat dilakukan dengan memberikan perintah:

```
# cd /usr/local/apache/bin
# ./apachectl stop
```

- g. File konfigurasi **Apache** bernama **httpd.conf** yang berada di **/usr/local/apache/conf** tidak perlu dirubah, sehingga secara *default* dokumen-dokumen *web* yang akan dipublikasikan berada di *directory* **/usr/local/apache/htdocs**.

Instalasi MySQL versi 4.0.21.

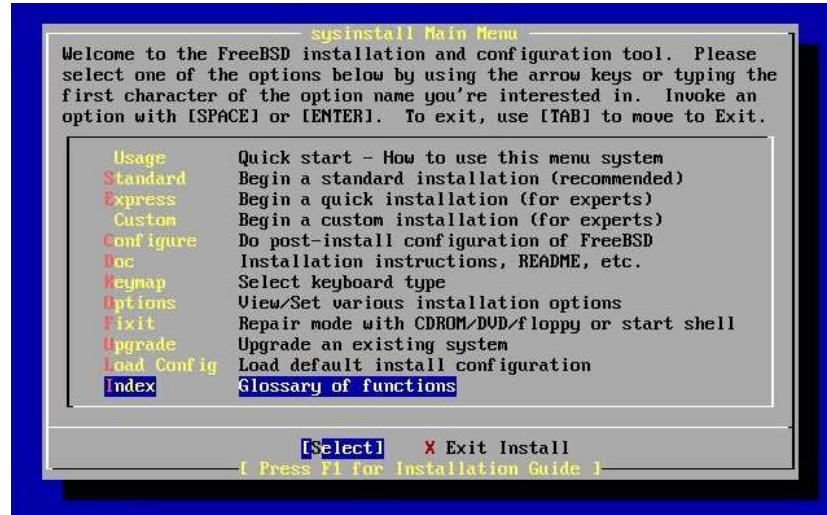
Instalasi paket *software* ini dilakukan secara manual, yaitu dengan kompilasi sendiri. Langkah-langkah yang digunakan adalah sebagai berikut:

- Download source code* **MySQL** versi 4.0.21 dari situs resminya www.mysql.com atau *mirror server* yang disediakan. Nama filenya adalah **mysql-4.0.21.tar.gz**. *Source code* ini masih terkompresi sebagai **.tar.gz**, sehingga perlu diekstrak.
- Login sebagai *user* **root** untuk mengekstrak file tersebut, dengan perintah:
tar -zxvf mysql-4.0.21.tar.gz
- Setelah terekstrak, maka sebelum memulai langkah kompilasi dan instalasi **MySQL**, dibuat lebih dulu *group* dan *user* dengan nama **mysql** yang

nantinya akan digunakan sebagai *user* untuk menjalankan *software* tersebut. FreeBSD menyertakan *tool* yang memudahkan *administrator* untuk membuat *user* baru maupun *group* baru, *tool* tersebut adalah **sysinstall** yang berada di *directory* **/stand**. Langkah-langkah untuk membuat *group* dan *user* baru adalah sebagai berikut:

```
# cd /stand
# ./sysinstall
```

Perintah yang terakhir akan memanggil *tool* tersebut, seperti pada gambar berikut ini:



Gambar 15. Menu Utama **sysinstall**.

Kemudian dipilih **Index > User Management** seperti pada tampilan berikut ini:



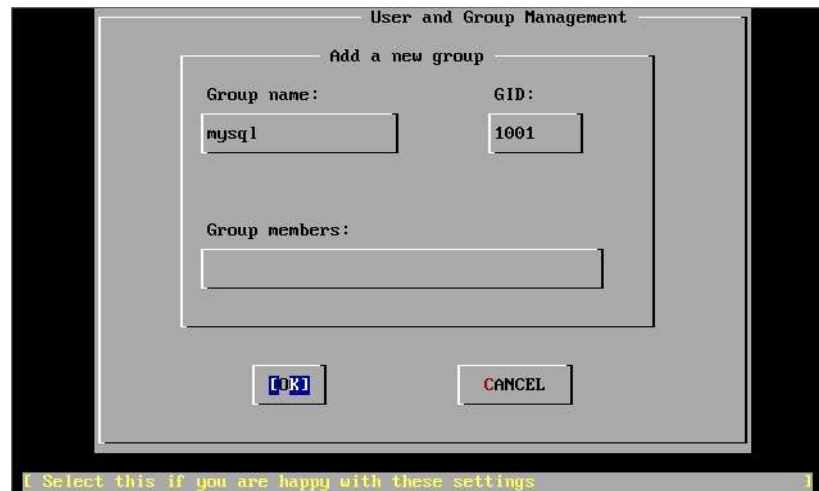
Gambar 16. Pilihan untuk membuat *user login* baru.

Sehingga terdapat tampilan sebagai berikut:



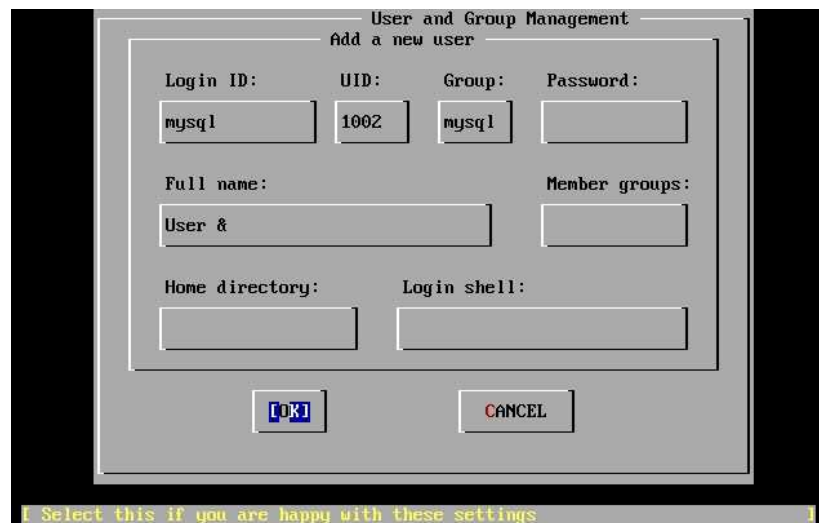
Gambar 17. Membuat *user* atau *group* baru.

Kemudian pembuatan *group* dikerjakan, seperti pada tampilan berikut ini:



Gambar 18. Membuat *group* baru.

Demikian juga halnya dengan pembuatan *user* **mysql**, dikerjakan seperti pada tampilan berikut ini:



Gambar 19. Membuat *user login* baru.

- d. Setelah berhasil membuat *user* dan *group* **mysql**, maka dilanjutkan dengan konfigurasi, kompilasi, dan instalasi. Langkahnya adalah masuk ke

directory hasil ekstraksi; biasanya sama dengan nama filenya tanpa ekstensi (**mysql-4.0.21**), kemudian mengonfigurasi **MySQL** supaya diarahkan ke *directory* **/usr/local/mysql** yang akan menyimpan *software* ini dan akan digunakan oleh modul PHP. Perintah yang digunakan adalah:

```
# ./configure --prefix=/usr/local/mysql
```

- e. Setelah konfigurasi di atas, dilanjutkan dengan kompilasi dan instalasi *software* **MySQL**, dengan perintah:

```
# make; make install
```

- f. Jika tidak ada kesalahan atau *error* pada proses konfigurasi, kompilasi, dan instalasi, maka **MySQL** sudah berhasil diinstall. Tetapi **MySQL** belum dapat berjalan karena masih dibutuhkan konfigurasi-konfigurasi lain supaya **MySQL** dapat berjalan dengan baik.

- g. **MySQL** membutuhkan file **.cnf** untuk inisialisasi pada saat dipanggil, nama filenya adalah **my.cnf** yang seharusnya terletak di **/etc**. Pada saat selesai instalasi, file **.cnf** ini belum ada, sehingga perlu dibuat. Pembuatan file ini tidak sulit, karena tinggal *mengcopy*kan dari file **my-medium.cnf** yang berada di dalam *directory* **/usr/local/mysql/share/mysql** ke *directory* **/etc** dengan nama **my.cnf**. Hal ini dilakukan dengan perintah:

```
# cd /usr/local/mysql/share/mysql
# cp my-medium.cnf /etc/my.cnf
```

- h. Kemudian dibuat *database* awal **MySQL** dengan perintah:

```
# cd /usr/local/mysql/bin
# ./mysql_install_db --user=mysql
```

- i. Setelah itu akan tercipta sebuah *directory* **var** di dalam **/usr/local/mysql** yang merupakan *directory* tempat *database* **MySQL** tersimpan. Hak akses pada *directory* ini perlu diset ulang, karena **MySQL** akan bekerja dengan *user* **mysql**. *Directory* **/usr/local/mysql/var** dan isinya akan diset kepemilikannya dengan *user* **mysql** dan *group* **mysql** dengan cara:

```
# cd /usr/local/mysql
# chown -R mysql var
# chgrp -R mysql var
```

- j. Setelah konfigurasi-konfigurasi di atas, maka **MySQL** sudah siap untuk dijalankan. Untuk menjalankannya dilakukan dengan cara:

```
# cd /usr/local/mysql/bin
# ./mysqld_safe -user=mysql &
# ps -ax
```

Baris terakhir dari perintah di atas adalah perintah untuk melihat proses yang sedang aktif saat itu. Jika proses **mysqld** ada, maka **MySQL** berhasil diinstall dan bekerja, seperti ditunjukkan pada gambar berikut:


```

24 ?? WL 0:04.06 (irq1: atkbd0)
26 ?? WL 0:00.03 (irq6: fdc0)
33 ?? DL 0:02.02 (vnlrn)
34 ?? IL 0:00.00 (nfsiod 0)
35 ?? IL 0:00.00 (nfsiod 1)
36 ?? IL 0:00.00 (nfsiod 2)
37 ?? IL 0:00.00 (nfsiod 3)
227 ?? Is 0:04.00 /usr/sbin/syslogd -s
361 ?? Is 0:00.77 /usr/sbin/sshd
366 ?? Ss 0:11.07 sendmail: accepting connections (sendmail)
369 ?? Is 0:00.31 sendmail: Queue runner00:30:00 for /var/spool/client
383 ?? Is 0:06.57 /usr/sbin/cron
420 v0 Is 0:01.08 login [pam] (login)
428 v0 R 0:05.38 -csh (csh)
84282 v0 S 0:00.32 /bin/sh ./mysqld_safe --user=mysql
84252 v0 S 0:13.64 /usr/local/mysql/libexec/mysqld --basedir=/usr/local/
84255 v0 R+ 0:00.01 ps -ax
42283 v1 Is+ 0:00.16 /usr/libexec/getty Pc ttyv1
422 v2 Is+ 0:00.17 /usr/libexec/getty Pc ttyv2
423 v3 Is+ 0:00.17 /usr/libexec/getty Pc ttyv3
424 v4 Is+ 0:00.15 /usr/libexec/getty Pc ttyv4
425 v5 Is+ 0:00.15 /usr/libexec/getty Pc ttyv5
426 v6 Is+ 0:00.16 /usr/libexec/getty Pc ttyv6
427 v7 Is+ 0:00.16 /usr/libexec/getty Pc ttyv7
web#

```

Gambar 20. Proses `mysqld` sedang berjalan.

Instalasi PHP versi 4.3.9.

Instalasi paket *software* ini dilakukan secara manual, yaitu dengan kompilasi sendiri. Langkah-langkah yang digunakan adalah sebagai berikut:

- Download source code* **PHP** versi 4.3.9 dari situs resminya www.php.net atau *mirror server* yang disediakan. Nama filenya adalah **php-4.3.9.tar.gz**. *Source code* ini masih terkompresi sebagai **.tar.gz**, sehingga perlu diekstrak.
- Login sebagai *user root* untuk mengekstrak file tersebut, dengan perintah:

```
# tar -zxvf php-4.3.9.tar.gz
```
- Setelah terekstrak, maka langkah kompilasi dan instalasi **PHP** dapat dimulai. Langkah pertama adalah masuk ke *directory* hasil ekstraksi; biasanya sama dengan nama filenya tanpa ekstensi (**php-4.3.9**), kemudian mengonfigurasi **PHP** supaya diarahkan ke *directory* **/usr/local/apache/bin** yang akan memanggil suatu program kecil **apxs** sehingga **PHP** dapat terintegrasi dengan **Apache**. Disamping itu perlu juga diarahkan ke *directory* **/usr/local/mysql** supaya **PHP** dapat berinteraksi dengan **MySQL**. Perintah yang digunakan adalah:

```
# ./configure --with-apxs2=/usr/local/apache/bin/apxs \
--with-mysql=/usr/local/mysql --with-track-vars \
--with-config-file-path=/etc
```
- Setelah konfigurasi di atas, maka dilanjutkan dengan kompilasi dan instalasi **PHP**, dengan perintah:

```
# make; make install
```
- Jika tidak ada kesalahan atau *error* pada proses konfigurasi, kompilasi, dan instalasi, maka **PHP** sudah berhasil diinstall. Untuk memastikan bahwa **PHP** sudah terinstall, maka perlu diperiksa file **httpd.conf** yang merupakan file inisialisasi **Apache**. Di dalam file tersebut seharusnya tersisipkan suatu baris perintah:

```
LoadModule php4_module modules/libphp4.so
```

Statemen di atas berarti memerintahkan **Apache** untuk *meload* modul **PHP**. Jika statemen tersebut tidak ada, maka harus disisipkan secara manual.

- f. Disamping ada penyisipan otomatis pada waktu penginstallan **PHP**, masih ada yang harus disisipkan secara manual; yaitu pada statemen yang akan mengenali **index.php** sebagai file awal yang harus dipanggil pertama kali. Penyisipan tersebut terletak di statemen:

```
DirectoryIndex index.php index.html index.html.var
```

Serta juga menambahkan statemen berikut ini untuk mengenalkan **Apache** dengan file **PHP**, sehingga file **PHP** dapat dieksekusi:

```
AddType application/x-httpd-php .php
```

- g. Kemudian dibuat file **php.ini** yang merupakan file inisialisasi **PHP**. File ini dicopykan dari file **php.ini-dist** di *directory* sumbernya yaitu di *directory* **php-4.3.9** ke *directory* **/etc** dengan nama **php.ini** dengan perintah:

```
# cd /php-4.3.9
# cp php.ini-dist /etc/php.ini
```

Testing koneksi dengan client.

Sebelum dilakukan serangkaian tes untuk menguji *web server* maka dilakukan konfigurasi supaya paket-paket *software Web Server* yang sudah terinstall tersebut dapat secara otomatis terpanggil pada waktu *booting*. Cara yang dilakukan adalah dengan membuat *script* yang akan diletakkan di *directory* **/usr/local/etc/rc.d**. Nama file *script* bebas tetapi harus berekstensi **.sh**, pada makalah ini file *script* tersebut diberi nama **awal.sh** yang isinya sebenarnya memanggil **Apache** dan **MySQL** seperti berikut ini:

```
./usr/local/apache/bin/apachectl start
./usr/local/mysql/bin/mysqld_safe --user=mysql &
```

File **awal.sh** kemudian diubah modenya menjadi *executable*, supaya dapat dieksekusi oleh sistem dengan perintah:

```
# chmod +x awal.sh
```

Kemudian server *booting* ulang dengan perintah:

```
# reboot
```

Setelah *booting* selesai dan Sistem Operasi berjalan, maka dapat diperiksa apakah pemanggilan otomatis tersebut dapat berjalan dengan baik, dengan mengetikkan perintah:

```
# ps -ax
```

Jika proses **httpd** dan **mysqld** terlihat di layar, maka pembuatan *script* dan instalasi keseluruhan *web server* sudah berjalan dengan baik, tampilan akan terlihat sebagai berikut:

```

377 ?? Is 0:00.10 sendmail: Queue runner@00:30:00 for /var/spool/client
391 ?? Ss 0:00.15 /usr/sbin/cron
403 ?? Ss 0:00.39 /usr/local/apache/bin/httpd -k start
410 ?? I 0:00.04 /usr/local/apache/bin/httpd -k start
411 ?? I 0:00.09 /usr/local/apache/bin/httpd -k start
412 ?? I 0:00.06 /usr/local/apache/bin/httpd -k start
413 ?? I 0:00.07 /usr/local/apache/bin/httpd -k start
414 ?? I 0:00.07 /usr/local/apache/bin/httpd -k start
469 ?? S 0:04.54 /usr/local/mysql/libexec/mysqld --basedir=/usr/local/
459 u0 Is 0:00.28 login (pam) (login)
470 u0 S 0:00.14 -csh (csh)
474 u0 R+ 0:00.00 ps -ax
460 v1 Is+ 0:00.13 /usr/libexec/getty Pc ttyu1
461 v2 Is+ 0:00.20 /usr/libexec/getty Pc ttyu2
462 v3 Is+ 0:00.11 /usr/libexec/getty Pc ttyu3
463 v4 Is+ 0:00.10 /usr/libexec/getty Pc ttyu4
464 v5 Is+ 0:00.21 /usr/libexec/getty Pc ttyu5
465 v6 Is+ 0:00.19 /usr/libexec/getty Pc ttyu6
466 v7 Is+ 0:00.14 /usr/libexec/getty Pc ttyu7
404 con- I 0:00.31 /bin/sh ./usr/local/mysql/bin/mysqld_safe --user=mysql
451 con- I 0:00.01 sh /etc/rc autoboot
453 con- I 0:00.03 sh /etc/rc autoboot
454 con- I 0:00.08 logger -p daemon.notice -t fsck
455 con- I 0:00.02 sleep 60
web#

```

Gambar 21. Proses **httpd** dan **mysqld** sedang berjalan.

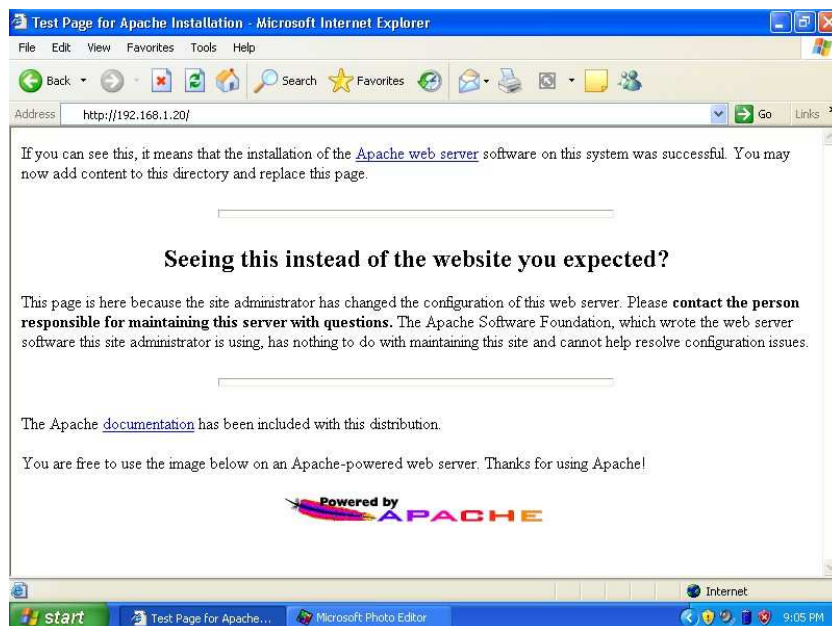
Konfigurasi selanjutnya adalah membuat file **info.php** yang berfungsi untuk mengetes apakah **PHP** sudah dapat berjalan dengan baik. Isi file **info.php** adalah:

```

<?
phpinfo();
?>

```

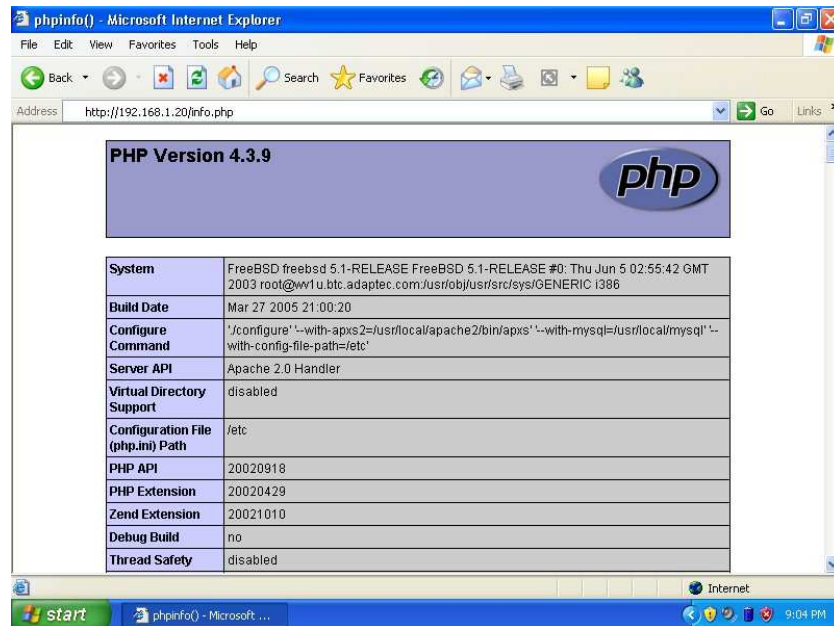
Testing selanjutnya merupakan percobaan awal koneksi dari *client* ke *web server* menggunakan Internet Explorer dengan alamat URLnya dituliskan **http://192.168.1.20**. Jika terdapat tampilan sebagai berikut, maka koneksi dan *web server* sudah berjalan dengan baik:



Gambar 22. Tampilan dari *client* ketika akses pertama.

Langkah berikutnya adalah testing **PHP**, menggunakan Internet Explorer dengan alamat URL yang sama, tetapi ada tambahan nama file **info.php**, yaitu

<http://192.168.1.20/info.php>. Jika terdapat tampilan sebagai berikut, maka koneksi dan *web server* dengan PHP sudah berjalan dengan baik:



Gambar 23. Tampilan dari *client* ketika mengakses [info.php](http://192.168.1.20/info.php).

Testing dengan Webstress versi 6.0 dan httperf versi 0.8.

Pada testing ini, *web server* sudah diberi *content web* lengkap dari www.contoh.com sehingga pengetesan dapat berjalan.

1. Webstress 6.0.

Software ini mensimulasikan kepadatan dan penggunaan *web server* dalam jumlah besar yang dihasilkan oleh banyak *user* secara bersamaan.

Testing ini menggunakan parameter WAKTU TES yaitu selama 5 menit, JUMLAH USER adalah 10, WAKTU antara klik satu dengan klik yang lainnya adalah 5 detik, dan halaman-halaman *web* yang akan diakses. Halaman *web* yang diakses adalah beberapa halaman *web* www.contoh.com yang sudah dicopykan ke server percobaan.

Penulis mengambil kesimpulan dari hasil tes ini bahwa server *web* masih dapat menangani *request* yang dihasilkan oleh *software*, walaupun terlihat banyak *error* yang dihasilkan ketika tes dijalankan. Hasil lengkap pengetesan dengan *software* ini ada dalam lampiran 1.

2. httperf 0.8.

Software ini mensimulasikan kepadatan dan penggunaan *web server* dalam jumlah besar yang dihasilkan oleh banyak *session* dan panggilan secara bersamaan.

Testing menggunakan **httplib 0.8** di *client* Linux tidak menggunakan halaman *web* tetapi dengan parameter saja. Parameter yang digunakan adalah 20 *session* dengan masing-masing *session* akan menciptakan 10 panggilan yang masing-masing dipisahkan dengan waktu 2 detik.

Dari hasil pengetesan ini, penulis dapat mengambil kesimpulan bahwa server web yang dibangun ini dapat menangani 20 *session* dengan 19 koneksi yang bersamaan selama 37 detik. Hasil lengkap pengetesan dengan *software* ini ada dalam lampiran 2.

Implementasi Makalah *web server* untuk www.contoh.com.

Setelah diuji coba dengan *software-software* di atas dan dapat dikatakan berhasil serta layak digunakan, maka *web server* ini diimplementasikan untuk www.contoh.com.

Hardware yang digunakan untuk *web server* ini adalah *hardware* yang dirancang khusus untuk *server*, dengan spesifikasi sebagai berikut:

Processor Intel Pentium IV 3 GHz
Main Board Intel SE7210
RAM 1 GB
AGP 8MB onboard
Sound Card AC'97
HDD 80 GB SCSI
CDROM 52x
LAN Card intel 10/100 Mbps dan 1 Gbps

Konfigurasi dan instalasi FreeBSD beserta paket-paket *software* pendukungnya berjalan dengan lancar karena konfigurasi dan instalasinya tidak berbeda jauh dengan komputer sebelumnya. FreeBSD versi 5.1-RELEASE mempunyai kompatibilitas yang baik terhadap *hardware* yang digunakan ini.

Perbedaan konfigurasi yang menyolok adalah alamat IP lokal yang harus diubah menjadi alamat publik, yaitu 202.xxx.yyy.zzz dengan *netmask* 255.255.255.240 beserta *nameserver*nya, yaitu 202.xxx.yyy.aaa

Disamping itu, harus dibuat sebuah *user login* baru yang akan memelihara dan merawat file-file *web* untuk www.contoh.com. *User login* baru ini diberi nama **webmaster** yang mempunyai *directory home* di **/usr/local/apache/htdocs**.

Kesimpulan dan Saran.

Instalasi Sistem Operasi FreeBSD versi 5.1-RELEASE tidak begitu sulit, hampir sama dengan instalasi Linux dengan *text mode*. Ada perbedaan yang menyolok dengan Linux, yaitu organisasi penyimpanan di FreeBSD mengenal istilah partisi dan *slice*. Partisi merupakan bagian dari *slice*, sehingga satu *slice* dapat terdiri dari banyak partisi.

Instalasi **Apache**, **MySQL**, dan **PHP** dilakukan secara manual sehingga menjadikan instalasi terlihat lebih rumit. Tetapi hal ini membantu penulis mengetahui lebih dalam tentang *software-software* tersebut. Misalnya bagaimana **PHP** diselipkan di dalam **Apache** sehingga **Apache** dapat mengenali dan menjalankan *script PHP* serta **PHP** harus dikenalkan dengan **MySQL** ketika akan dikompilasi.

Pada makalah ini kernel Sistem Operasi FreeBSD versi 5.1-RELEASE tidak perlu dikompilasi ulang, karena sudah dapat bekerja dengan baik. Kernel yang digunakan merupakan kernel *generic*, yaitu kernel yang tidak tergantung dengan suatu *hardware* tertentu. Jika *web server* yang dibangun merupakan suatu *web server* yang mempunyai tingkat kepadatan akses yang tinggi, maka lebih baik jika kernelnya

dikompilasi ulang untuk menyesuaikan dengan *hardware* yang ada supaya lebih efisien.

Makalah ini ketika diimplementasikan untuk www.contoh.com tidak mengalami banyak perubahan. Perubahan yang terjadi merupakan penyesuaian untuk berhubungan dengan internet.

Web server yang dibangun ini sistem keamanannya masih rendah, sehingga untuk kebutuhan mendatang perlu dibangun suatu sistem keamanan yang lebih kuat keamanannya.

Daftar Pustaka.

- a. <http://www.freebsd.org>
- b. <http://www.apache.org>
- c. <http://www.mysql.com>
- d. <http://www.php.net>
- e. <http://www.wikipedia.org>
- f. <http://www.kernel.org>
- g. Kurniawan, Yahya, S.T. 2002. *Singkat Tepat Jelas: Aplikasi Web Database dengan PHP dan MySQL*. Elex Media Komputindo. Jakarta.
- h. Mansfield, Niall. 2003. *Practical TCP/IP: Designing, Using, and Troubleshooting TCP/IP Networks on Linux and Windows*. Pearson Education. UK.
- i. Smith, Roderick W. 2003. *FreeBSD: The Complete Reference*. McGraw-Hill/Osborne. USA.
- j. _____. 2003. *FreeBSD Handbook*. The FreeBSD Documentation Project.