

WEB SERVER

TUJUAN PEMBELAJARAN

1. Mahasiswa memahami prinsip kerja web server.
2. Mahasiswa memahami kegunaan/manfaat web server.
3. Mahasiswa mampu melakukan installasi dan konfigurasi/*setting* web server.

Dasar Teori

Saat ini *web* merupakan salah satu layanan informasi yang banyak diakses oleh pengguna internet di dunia. Sebagai salah satu layanan informasi maka perlu dibangun *web* yang mampu menangani permintaan (*request*) dari banyak pengguna dengan baik (*reliable*) tanpa meninggalkan aspek keamanannya. Masalah keamanan merupakan salah satu aspek yang penting dalam pembangunan *web* karena kelalaian dalam menangani keamanan *web* server dapat berakibat fatal.

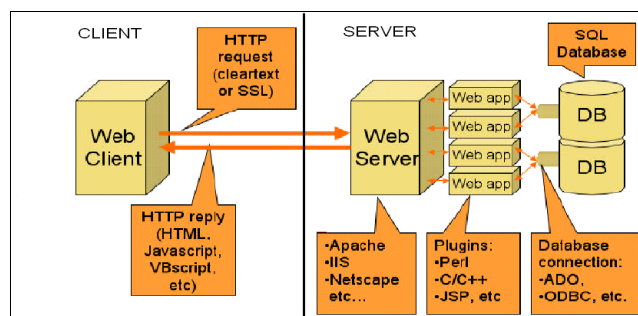
Apache merupakan salah satu distribusi *web server* yang populer dengan dukungan *feature* yang sangat banyak. Perhitungan statistik yang ada saat ini menunjukkan bahwa *Apache* menjadi *web server* yang paling banyak digunakan dalam dunia internet, yaitu mencapai nilai 60 % dari seluruh *web server* yang ada.

Keberhasilan *Apache* mencapai kepopuleran saat ini selain dikarenakan memiliki banyak *feature* yang sering tidak dijumpai pada *web server* yang lain, juga dikarenakan *Apache* merupakan aplikasi gratis yang berjalan dalam berbagai sistem operasi.

Ada beberapa aspek yang perlu diterapkan dalam mengamankan *web server*, antara lain:

1. Layanan *web server* dengan *low privillages*
2. Pengaturan akses terhadap *web server*
3. Meminimalkan layanan publik pada mesin yang menjalankan *web server*
4. Menyediakan *filesystem* khusus untuk layanan *web server*

Secara garis besar, sistem yang ada pada web sistem bisa digambar sebagai berikut ;



Web server bertugas memberikan layanan ke client. Web server berguna memberikan layanan kepada client.

Web application merupakan program tambahan yang ada pada server web yang membuat tampilan web bisa dinamis. Beberapa aplikasi yang dipakai seperti perl, asp, jsp dan php. Selain itu web application juga berfungsi sebagai penghubung dan pengambil data ke database sesuai permintaan client.

Pada web application dikenal istilah server side scripting dan client side scripting. Server side scripting adalah kode yang ditempatkan di server dijalankan oleh server, contoh dari server side misal adalah php. Sedangkan client side scripting adalah kode program yang ditempatkan di sisi client yang dijalankan oleh browser di client. Contohnya adalah javascript, applet, jscript dan vbscripts. Pada client side ini browser harus bisa menterjemahkan bahasa pemrograman script tadi untuk bisa mengeksekusinya

Install Apache

Di terminal (application → accesoris → terminal), masukkan code sebagai berikut :

```
$ sudo apt-get install apache2
```

Untuk memeriksa apakah Apache sudah terinstal dengan benar dan dapat digunakan, buka halaman <http://localhost/> pada browser anda. Jika sudah benar, maka akan muncul halaman putih yang bertuliskan :

It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

Root folder untuk apache, di Ubuntu, ada di `/var/www`. Dan file konfigurasinya ada di `/etc/apache2/apache2.conf`.

Install MySQL

Masukkan perintah berikut di terminal :

```
$ sudo apt-get install mysql-server
```

(Nanti akan ada sesi untuk menginput password root MySQL)

Untuk membuat database, pada terminal masukkan perintah :

```
$ mysqladmin -u root -p create <namadatabase>
```

Untuk masuk ke MySQL :

```
$ mysql -u root -p
```

Setelah masuk ke MySQL. Iseng-iseng cek database yang ada :

```
mysql> show databases
```

Maka database yang baru saja dibuat ada disana. Untuk keluar dari mysql, ketikkan "quit".

Install PHP dan download module untuk MySQL

Masukkan perintah berikut di terminal :

```
$ sudo apt-get install php5-mysql
```

Selanjutnya, restart apache dengan code :

```
$ sudo /etc/init.d/apache2 restart
```

Jika kemudian muncul pesan bahwa :

```
... waiting apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for ServerName
```

Itu karena belum didefinisikan ServerName sebagai localhost. Solusinya sederhana. (Masih di terminal). Ketikkan perintah :

```
$ sudo gedit /etc/apache2/httpd.conf
```

Anda akan masuk ke file httpd.conf. Secara default, isinya kosong. Selanjutnya, ketikkan baris berikut :

ServerName localhost

Lalu simpan file dan keluar dari editor.

Selanjutnya, restart kembali apache. InsyaAllah restart akan berjalan dengan baik.

Untuk memeriksa apakah sudah berjalan dengan baik, buat file php yang berisi :

```
<?php  
phpinfo();  
?>
```

Misalkan nama filenya info.php. Letakkan di folder root dari Apache, /var/www. Kemudian akses file tersebut di web-browser dengan memasukkan alamat URL : *http://localhost/info.php*. Jika keluar halaman yang berisi info-php, maka koneksi antara Apache dengan PHP sudah benar.

Pada info-php tersebut, periksa apakah modul MySQL sudah ada? Jika anda menggunakan perintah instalasi PHP seperti yang saya tulis di atas, pastinya modul tersebut sudah terinstall.