

FAMP:
FreeBSD Apache MySQL PHP

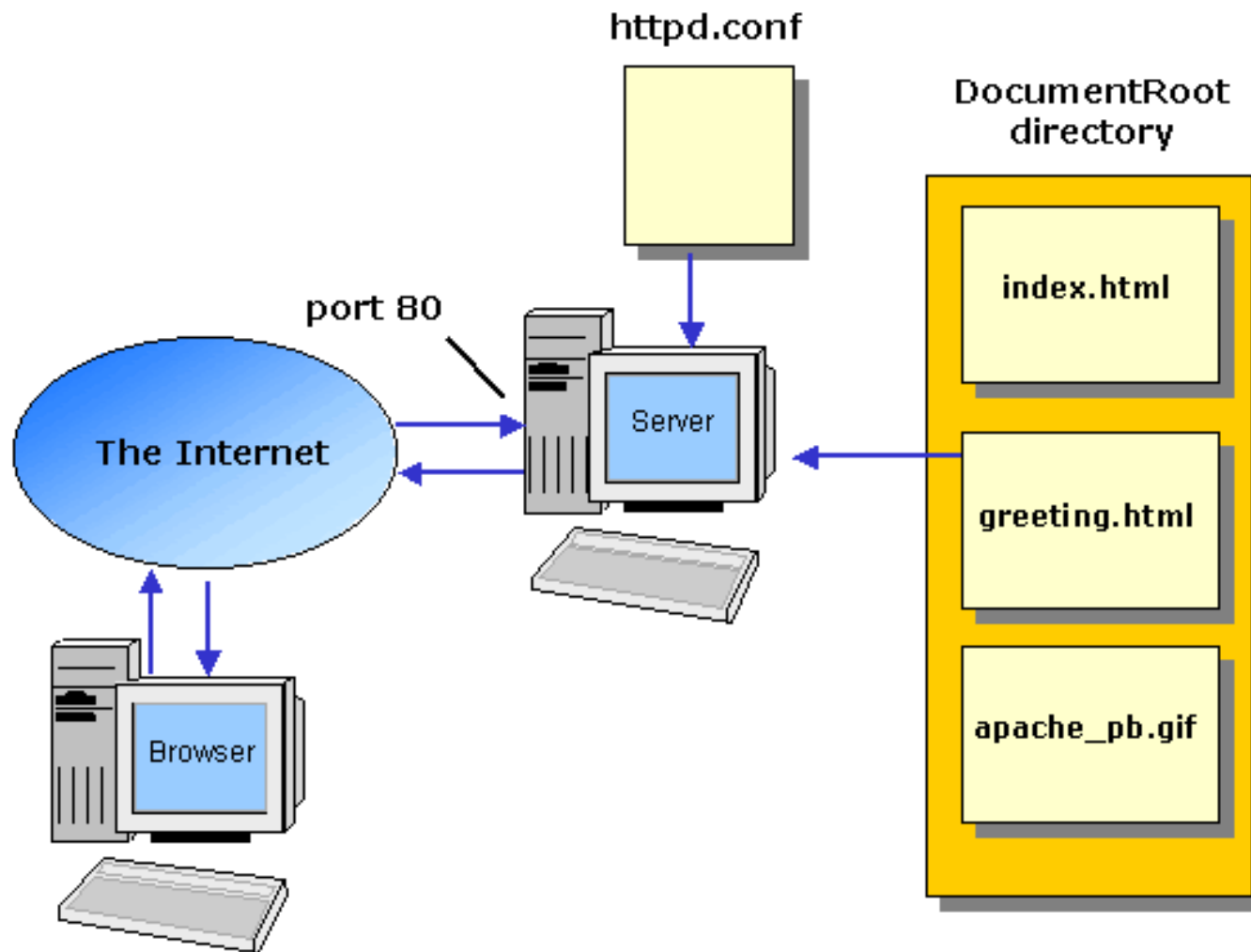
Outline

- Introductions
 - > Apache
 - > MySQL
 - > PHP
- Installation
 - > Apache + MySQL + PHP
- Administration
 - > Apache
 - > MySQL
- Appendix
 - > lighttpd

Apache Software Foundation & Apache HTTP Server Project

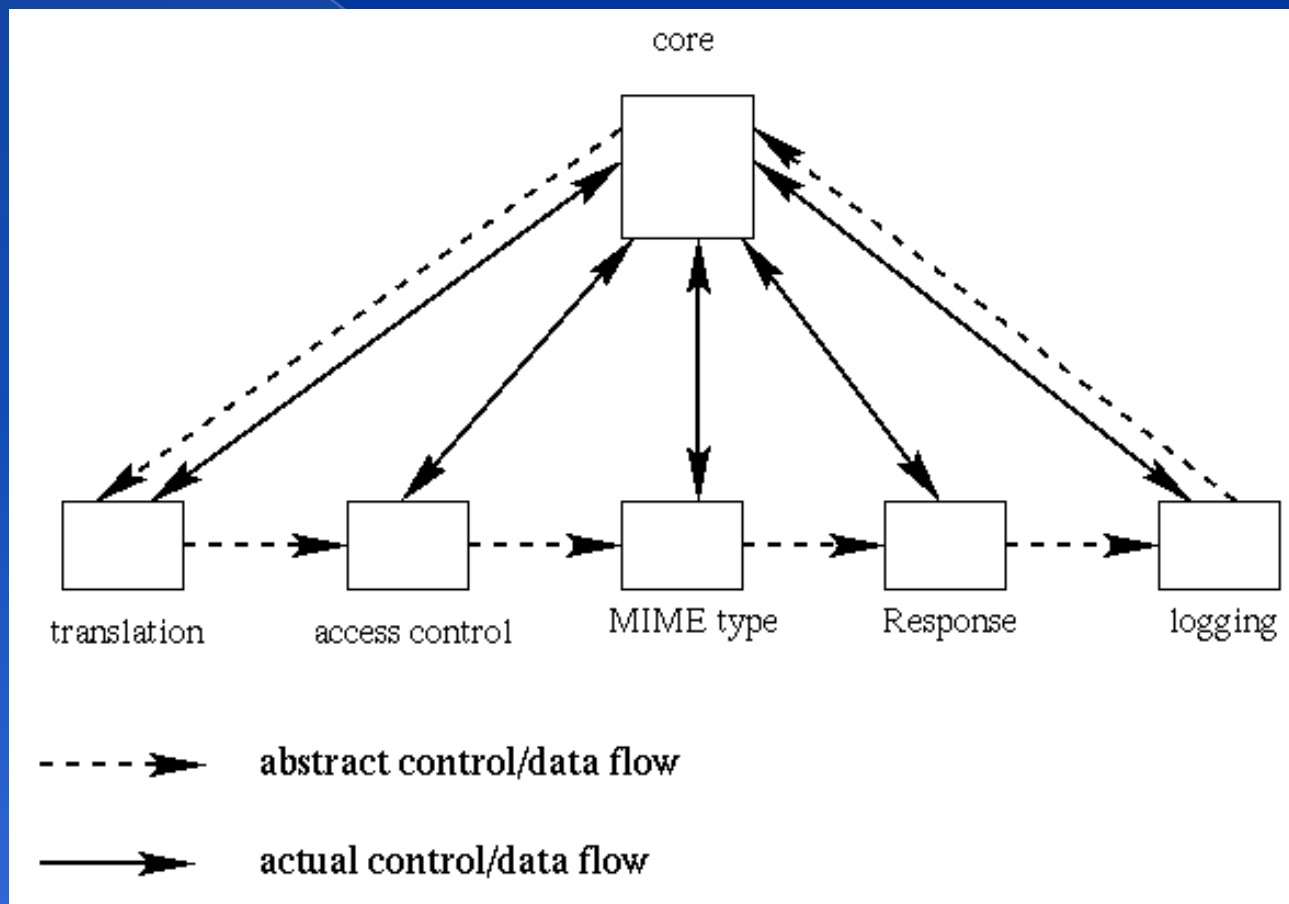
- Apache Software Foundation: <http://www.apache.org/>
- Apache HTTP Server Project: <http://httpd.apache.org/>
- Web httpd server that
 - > HTTP/1.1 compliant web server
 - > Modular design
 - > Can be customized by writing modules using Apache module API
 - > Freely available cross many platforms
- Two main parts
 - > core
 - Implement basic functions
 - > Modules
 - Extend or override the functionality of the server
 - Example:
 - Access control, logging, CGI, proxy, cache control, PHP...

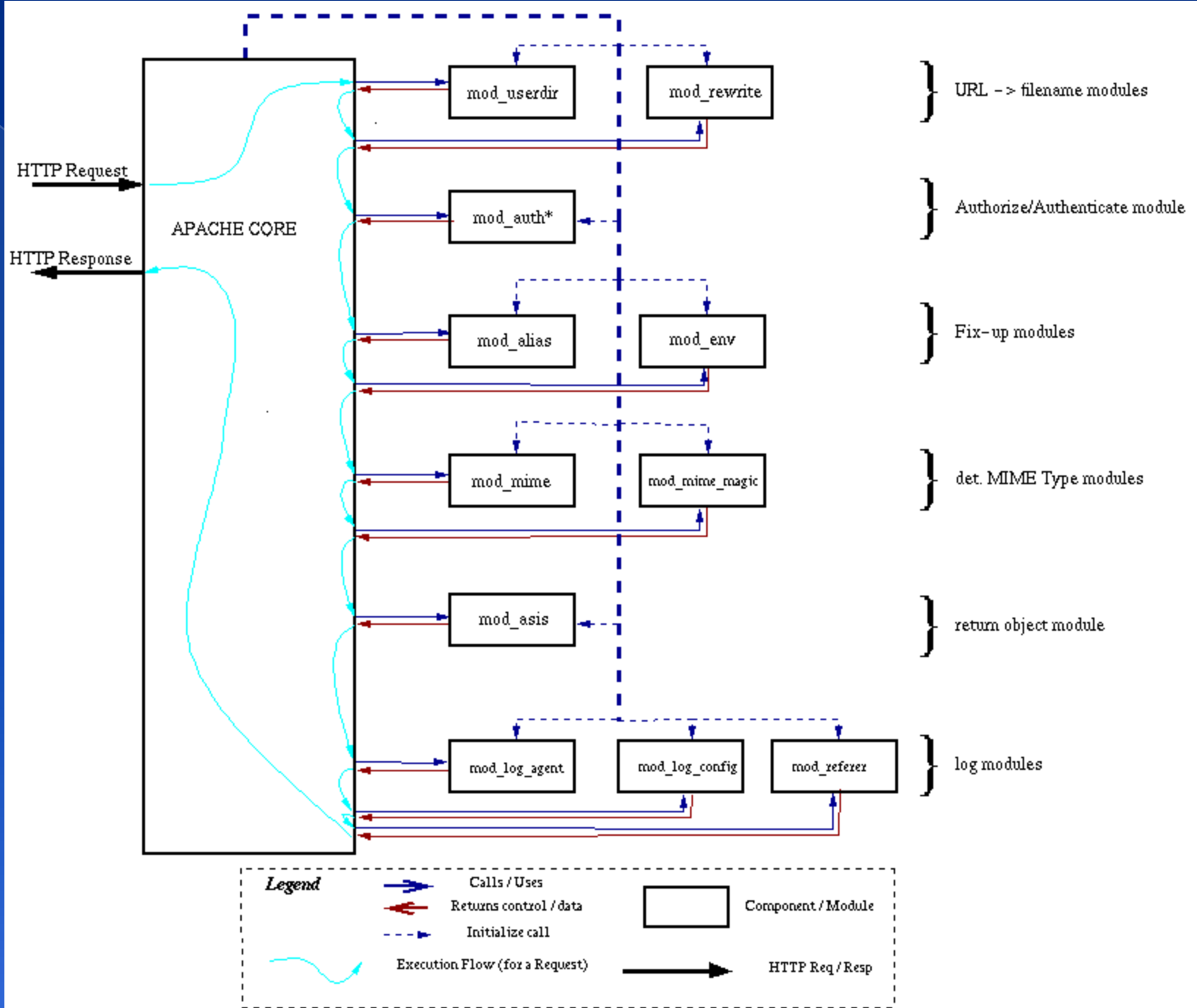
How Apache Works – request and response



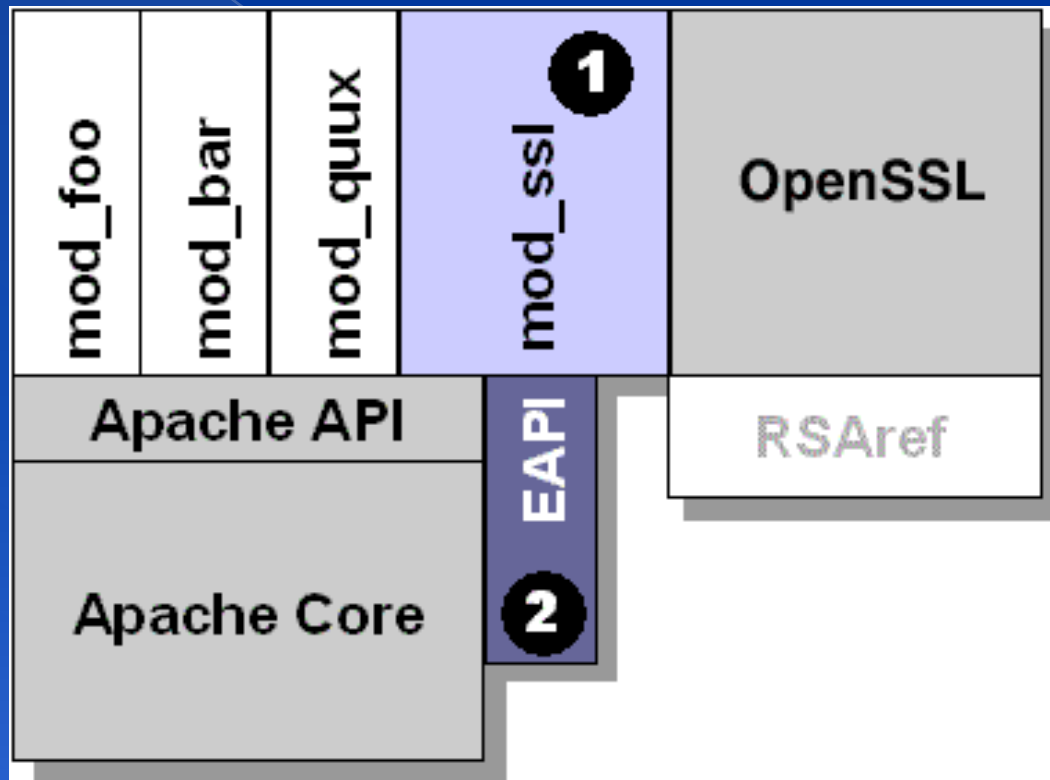
How Apache Works – Each request-response

- Apache breaks client request into several steps which are implemented as modules





Apache with mod_ssl



MySQL (1)

- Official Site: <http://www.mysql.com>
- SQL (Structured Query Language)
 - > The most popular computer language which is used to create, modify, retrieve and manipulate data from **relational database** management systems.
 - > Documents:
 - > <http://dev.mysql.com/doc>
 - > Introduction to SQL: (in Chinese)
<http://www.1keydata.com/tw/sql/sql.html>
- A **multithreaded, multi-user, SQL** Database Management System.
- MySQL is owned and sponsored by a Swedish company **MySQL AB**.

MySQL (2)

○ Characteristics:

- > Writing in C/C++, tested by many compilers, **portable to many platforms**.
 - AIX, FreeBSD, HP-UX, Linux, Mac OS, Solaris, Windows, ...etc.
- > Providing APIs for C/C++, Java, Perl, PHP, Python, Ruby, Tcl, ...etc.
- > **Multi-threaded** kernel, supporting systems with multiple CPUs.
- > Optimized algorithm for **SQL** Query.
- > Multi-Language (coding) Supports.
- > Lots of connecting method: TCP/IP, ODBC, JDBC, Unix domain socket.
- > **Free Software** (GNU General Public License version 2)

PHP

- PHP: Hypertext Preprocessor (<http://php.net/>)
 - > A widely-used Open Source general-purpose scripting language.
 - > Originally designed to create dynamic web pages, PHP's principal focus is server-side scripting.
 - > PHP scripts can be embedded into HTML.
 - > The LAMP architecture has become popular in the Web industry as a way of deploying inexpensive, reliable, scalable, secure web applications.
 - PHP is commonly used as the P in this bundle alongside Linux, Apache and MySQL.
 - FAMP replaces Linux with FreeBSD, WAMP replaces Linux with Windows.

Installation

Install Sequence – MySQL

● Steps

- > # cd/usr/ports/databases/mysql51-server/
- > # make WITH_XCHARSET=all install clean

● Add into rc.conf

- > mysql_enable="YES"

● Start up

- > # /usr/local/etc/rc.d/mysql-server start

Install Sequence – Openssl and Apache

◎ Steps

> `cd /usr/ports/www/apache22/`

> `make config`

- `WITH_CHARSET=utf8`
- `WITH_XCHARSET=all`
- `WITH_MPM=worker`
- `WITH_THREADS=yes`
- `WITH_SUEXEC=yes WITH_BERKELEYDB=db4`
- `WITH_STATIC_SUPPORT=yes`
- `WITH_ALL_STATIC_MODULES=yes`

> `Make install clean`

◎ Add into `/etc/rc.conf`

> `apache22_enable="YES"`

◎ Start up

> `/usr/local/etc/rc.d/apache22 start`

Install Sequence – PHP

◎ Steps

- > # cd /usr/ports/lang/php5
- > # make install clean
 - Remember to choose Apache module

◎ Install php5-extensions

- > # cd /usr/ports/lang/php5-extensions
- > # make install clean
 - Choose what you need
- > Or installing from /usr/ports/*/php5-*

Install Sequence – test PHP with Apache (1)

- Edit httpd.conf to support php
 - > /usr/local/etc/apache22/httpd.conf

```
<IfModule mime_module>
...
AddType application/x-httpd-php .php .phtml .php5
AddType application/x-httpd-php-source .phps
...
</IfModule>
```

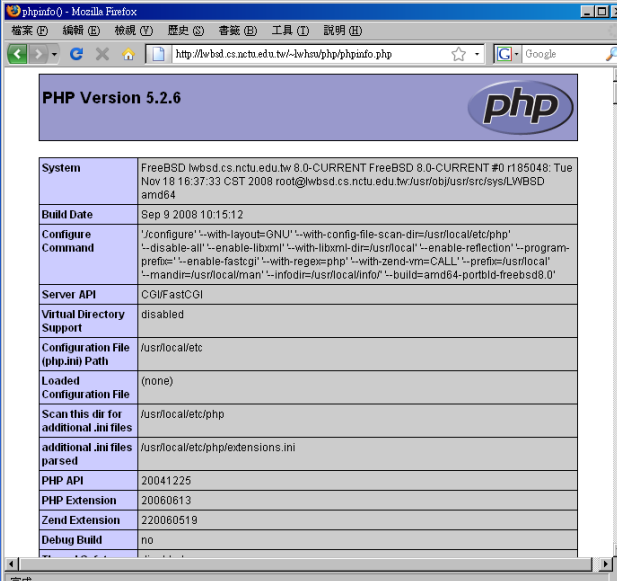
```
ServerName lwbsd.cs.nctu.edu.tw
# DocumentRoot "/usr/local/www/apache22/data"
DocumentRoot "/home/wwwadm/data"
...
# <Directory "/usr/local/www/apache22/data ">
<Directory "/home/wwwadm/data">
```

```
<IfModule mod_dir.c>
    DirectoryIndex index.php index.html index.htm
</IfModule>
```

Install Sequence – test PHP with Apache (2)

- Restart httpd
 - > # /usr/local/etc/rc.d/apache22 restart
- Test PHP with Apache
 - > \$ Edit /home/wwwadm,/data/index.php

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



The screenshot shows a Mozilla Firefox browser window displaying the output of the PHP info() function. The page title is "phpinfo() - Mozilla Firefox" and the URL is "http://lwbtd.cs.nctu.edu.tw/~lwhav/php/phpinfo.php". The page content includes the PHP logo and a table of system and configuration details.

| PHP Version 5.2.6 | |
|---|---|
| System | FreeBSD lwbtd.cs.nctu.edu.tw 8.0-CURRENT FreeBSD 8.0-CURRENT #0 r185048: Tue Nov 18 16:37:33 CST 2008 root@lwbtd.cs.nctu.edu.tw:/usr/obj/usr/src/sys/LWBSD amd64 |
| Build Date | Sep 9 2008 10:15:12 |
| Configure Command | "/configure" "--with-layout=GNU" "--with-config-file-scan-dir=/usr/local/etc/php" "--disable-all" "--enable-libxml" "--with-libxml-dir=/usr/local" "--enable-reflection" "--program-prefix=" "--enable-fastcgi" "--with-regex=php" "--with-zend-vm=CALL" "--prefix=/usr/local" "--mandir=/usr/local/man" "--infodir=/usr/local/info" "--build=amd64-portsdb-freebsd8.0" |
| Server API | CGIFastCGI |
| Virtual Directory Support | disabled |
| Configuration File (php.ini) Path | /usr/local/etc |
| Loaded Configuration File | (none) |
| Scan this dir for additional .ini files | /usr/local/etc/php |
| additional .ini files parsed | /usr/local/etc/php/extensions.ini |
| PHP API | 20041225 |
| PHP Extension | 20060613 |
| Zend Extension | 220060519 |
| Debug Build | no |

Administration

Apache configuration

◎ Location

- > The default location of apache (in ports) is `/usr/local/etc/apache22`
- > Major configuration file: `httpd.conf`
 - Other configuration files could be included. (setting in `httpd.conf`)

◎ Two types

- > Global configurations
 - Global setting
 - Server specific setting
 - Virtual host setting
- > Directory Configuration
 - Local setting for certain directory

Apache configuration – Global Configuration

- Global setting

- > ServerType standalone
- > Timeout 300
- > KeepAlive On
- > KeepAliveRequests 100
- > StartServers 5

- Server configuration

- > Port 80
- > ServerAdmin lwhsu@cs.nctu.edu.tw
- > ServerName lwbsd.cs.nctu.edu.tw
- > DocumentRoot "/home/wwwadm/data"

Apache configuration – Directory Configuration (1)

- Configuration parameters
 - > Options
 - All (turn on all options except multiview)
(To allow executions of
 - ExecCGI AddHandler
 - FollowSymLinks (access files outside this directory)
(generate file-list for browsing)
(when there is no DirectoryIndex files)
 - Indexs (multi-language support)
 - MultiViews
 - > AllowOverride
 - All (Read .htaccess)
 - None (ignoring .htaccess)
 - > Deny/Allow
 - IP/DN (control access to this directory)
 - > Order
 - Solve collision of deny and allow rules

```
<Directory "/home/wwwadm/data">  
    Options Indexes FollowSymLinks MultiViews  
    AllowOverride None  
    Order allow,deny  
    Allow from all  
</Directory>
```

Apache configuration – Directory Configuration (2)

```
# User home directories
#Include etc/apache22/extra/httpd-userdir.conf
```

```
UserDir public_html
UserDir disabled root toor daemon operator bin tty kmem games news man \
      sshd bind proxy _pflogd _dhcp uucp pop www nobody mailnull smmsp
#
# Control access to UserDir directories.  The following is an example
# for a site where these directories are restricted to read-only.
#
<Directory /home/*/public_html>
    AllowOverride FileInfo AuthConfig Limit Indexes
    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
    <Limit GET POST OPTIONS>
        Order allow,deny
        Allow from all
    </Limit>
    <LimitExcept GET POST OPTIONS>
        Order deny,allow
        Deny from all
    </LimitExcept>
</Directory>
```

Apache configuration – Directory Configuration (3)

```
<IfModule alias_module>
  Alias /icons/ "/usr/local/www/apache22/icons/"

  <Directory "/usr/local/www/apache22/icons">
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
  </Directory>

  Alias /manual/ "/usr/local/apache/htdocs/manual/"

  <Directory "/usr/local/apache/htdocs/manual">
    Options Indexes FollowSymlinks MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
  </Directory>
</IfModule>
```

Apache configuration – Virtual Host

○Name-Base

- > Single IP, several hostnames

NameVirtualHost 140.113.51.24

```
<VirtualHost 140.113.51.24>  
ServerName www.snmg.com.tw  
DocumentRoot "/www"  
</VirtualHost>
```

```
<VirtualHost 140.113.51.24>  
ServerName mail.snmg.com.tw  
DocumentRoot "/home/sywang"  
</VirtualHost>
```

```
<VirtualHost 140.113.51.24>  
ServerName csie.snmg.com.tw  
Redirect / http://www.csie.nctu.edu.tw/  
</VirtualHost>
```

○IP-Base

- > several IPs

<VirtualHost 140.113.50.33:80>

```
Port 80  
ServerAdmin webmaster@sun3.csie.nctu.edu.tw  
DocumentRoot /www/csie  
ServerName sun3.csie.nctu.edu.tw  
ErrorLog logs/csie-error_log  
TransferLog logs/csie-access_log  
</VirtualHost>
```

<VirtualHost 140.113.70.25:80>

```
Port 80  
ServerAdmin webmaster@sun3.ee.nctu.edu.tw  
DocumentRoot /www/ee  
ServerName sun3.ee.nctu.edu.tw  
ErrorLog logs/ee-error_log  
TransferLog logs/ee-access_log  
</VirtualHost>
```

Apache configuration – .htaccess (1)

- ◎ .htaccess

- > Allow admin to use one file to control access to certain directory

- ◎ Usage

- > Modify httpd.conf
- > Create .htaccess file
- > Generate password database
- > Test

Apache configuration – .htaccess (2)

- Example
 - > Modify httpd.conf
 - > Create .htaccess file
 - > Generate password file

```
<Directory "/www/data/test1">  
  Options Indexes FollowSymLinks MultiViews ExecCGI  
  AllowOverride All  
  Order allow,deny  
  Allow from all  
</Directory>
```

```
lwhsu@sabsd[3:02pm]/www/data/test1> cat .htaccess  
AuthName "SA-test1"  
AuthType "Basic"  
AuthUserFile "/www/data/test1/.htpasswd"  
require valid-user
```

```
lwhsu@sabsd[2:58pm]/> /usr/local/apache/bin/htpasswd -c ./htpasswd SA-user1  
New password:  
Re-type new password:  
Adding password for user SA-user1
```

Apache configuration - .htaccess (3)



Administering MySQL (1)

- Config file

- > Copy config file

- `% cd /usr/local/share/mysql`
 - `% sudo cp my-huge.cnf /etc/my.cnf`

- > Edit `/etc/my.cnf`

- Start up

- > Add into `rc.conf`

- `mysql_enable="YES"`

- > `# /usr/local/etc/rc.d/mysql-server start`

Administering MySQL (2)

◎ Test

> % `mysql -u root -p`

- The initial password for root is empty

```
lwhsu@sabsd:/var/log> mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2 to server version: 4.1.7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> show databases;
+-----+
| Database|
+-----+
| mysql   |
| test    |
+-----+
2 rows in set (0.27 sec)

mysql> exit
Bye
```

Administrating MySQL (3)

- Securing initial accounts
 - > Two initial accounts
 - root
 - anonymous

```
mysql> SELECT Host, User From mysql.user;
+-----+-----+
| Host          | User |
+-----+-----+
| localhost     |      |
| localhost     | root |
| sabsd.cs.nctu.edu.tw |      |
| sabsd.cs.nctu.edu.tw | root |
+-----+-----+
```

```
lwhsu@sabsd:~> mysql -u root -p
```

```
Enter password:
```

```
Welcome to the MySQL monitor. Commands end with ; or \g.
```

```
Your MySQL connection id is 4 to server version: 4.1.7-log
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
```

```
mysql> UPDATE mysql.user SET Password = PASSWORD('user123') WHERE User = '';
```

```
Query OK, 2 rows affected (0.26 sec)
```

```
Rows matched: 2  Changed: 2  Warnings: 0
```

```
mysql> UPDATE mysql.user SET Password = PASSWORD('root123') WHERE User = 'root';
```

```
Query OK, 2 rows affected (0.00 sec)
```

```
Rows matched: 2  Changed: 2  Warnings: 0
```

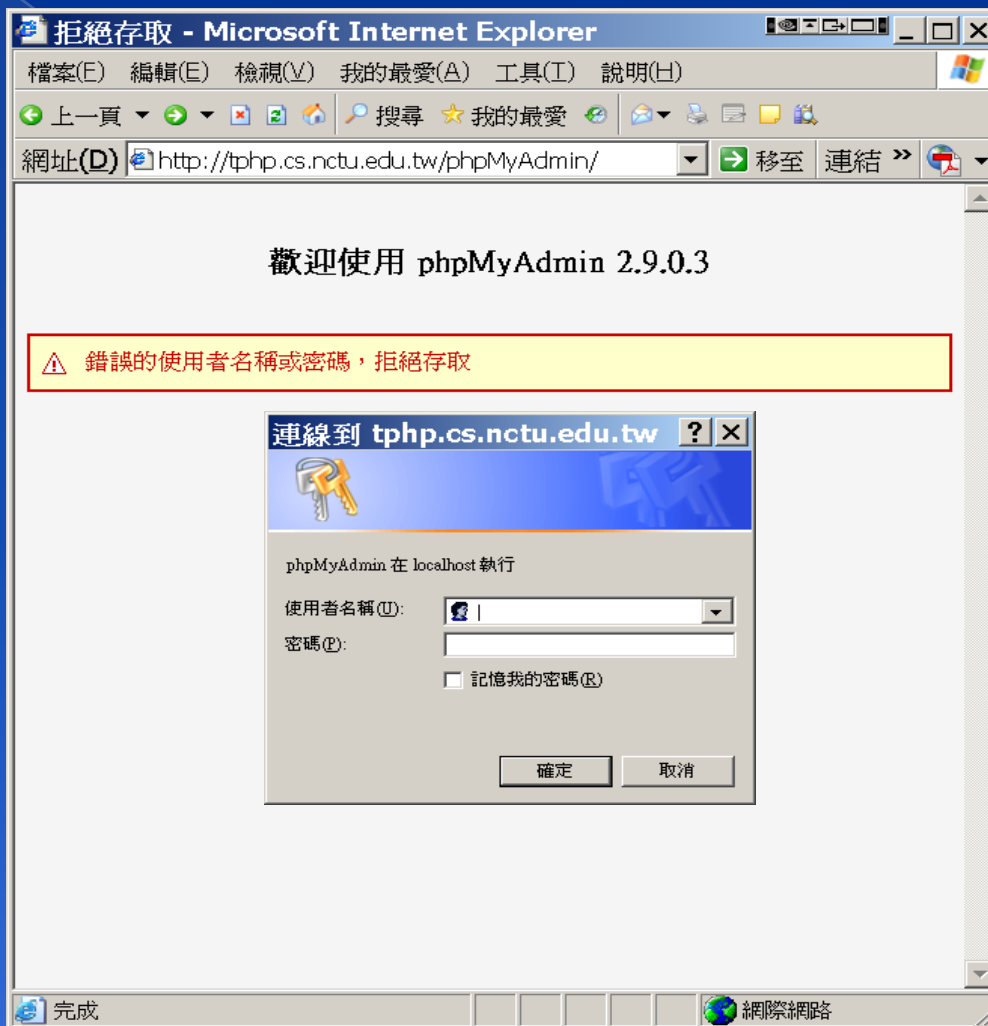
```
mysql> FLUSH PRIVILEGES;
```

```
Query OK, 0 rows affected (0.00 sec)
```

Administrating MySQL – Using phpMyAdmin (1)

- phpMyAdmin can manage a whole MySQL server as well as a single database.
- Official Site: <http://www.phpmyadmin.net/>
- Characteristics
 - > Browser-based, Supporting PHP5, MySQL 4.1 and 5.0, Open Source
- Installation Steps
 1. Download latest version from official site
 2. Unzip the archived file.
 3. Read documents: Documentation.html
 4. copy config.sample.inc.php → config.inc.php
 - Change auth type to http
 - Remove configuration about Advanced Feature (something start with 'pma')
 - > Browse the phpMyAdmin, and login.

Administrating MySQL – Using phpMyAdmin (2)



Administrating MySQL – Using phpMyAdmin (3)

localhost

- 伺服器版本: 5.0.27
- 通訊協定版本: 10
- 伺服器: Localhost via UNIX socket
- 使用者: chonsi@localhost
- MySQL 文字編碼: UTF-8 Unicode (utf8)
- MySQL 連線校對: utf8_unicode_ci
- 建立新資料庫
- 校對 建立
- 顯示 MySQL 執行狀態
- 顯示 MySQL 系統變數
- 處理
- 文字編碼及校對
- 儲存引擎
- 重新讀取權限
- 權限
- 資料庫
- 輸出

phpMyAdmin - 2.9.0.3

- MySQL 客戶端版本: 5.0.27
- 已使用 PHP 擴充附件: mysql
- Language: 中文 - Chinese traditional
- 款式: Original
- Font size: 100%
- phpMyAdmin 說明文件
- phpMyAdmin 官方網站
- [ChangeLog] [CVS] [Lists]

Administrating MySQL – Using phpMyAdmin (4)

◎ Create another user with limited privilege

The screenshot shows the phpMyAdmin 2.9.0.3 interface in Microsoft Internet Explorer. The browser address bar shows the URL `http://tphp.cs.nctu.edu.tw/phpMyAdmin/`. The page title is `localhost | phpMyAdmin 2.9.0.3 - Microsoft Internet Explorer`. The interface includes a navigation menu with options like `資料庫`, `SQL`, `狀態`, `資訊`, `文字編碼`, `引擎`, `權限`, `處理`, `輸出`, and `載入`. The main content area is titled `新增使用者` (Add New User) and contains the following sections:

- 登入資訊** (Login Information):
 - 使用者名稱 (Username): `文字輸入` (Text input)
 - 主機 (Host): `任何主機` (Any host)
 - 密碼 (Password): `文字輸入` (Text input)
 - 確認密碼 (Confirm Password): [Text input]
 - 產生密碼 (Generate Password): `產生` (Generate) / `複製` (Copy)
- Database for user**:
 - None
 - Create database with same name and grant all privileges
 - Grant all privileges on wildcard name (username_%)
- 整體權限 (全選 / 全部取消)** (Global Privileges):
 - 注意: MySQL 權限名稱會以英語顯示 (Note: MySQL privilege names will be displayed in English)
 - 資料** (Data):
 - SELECT
 - INSERT
 - UPDATE
 - DELETE
 - FILE
 - 結構** (Structure):
 - CREATE
 - ALTER
 - INDEX
 - DROP
 - CREATE TEMPORARY TABLES
 - CREATE VIEW
 - SHOW VIEW
 - CREATE ROUTINE
 - ALTER ROUTINE
 - EXECUTE
 - 系統管理** (System Management):
 - GRANT
 - SUPER
 - PROCESS
 - RELOAD
 - SHUTDOWN
 - SHOW DATABASES
 - LOCK TABLES
 - REFERENCES
 - REPLICATION CLIENT
 - REPLICATION SLAVE
 - CREATE USER
 - 資源限制** (Resource Limits):
 - 註: 設定這些選項為 0 (零) 可解除限制 (Note: Setting these options to 0 (zero) will remove the limit)
 - MAX QUERIES PER HOUR: [0]
 - MAX UPDATES PER HOUR: [0]
 - MAX CONNECTIONS PER HOUR: [0]
 - MAX_USER_CONNECTIONS: [0]

Appendix: Installing lighttpd

Installing lighttpd (1)

- Official: <http://www.lighttpd.net/>
- Installation:
 - > # cd /usr/ports/www/lighttpd
 - > # make install clean
- Supporting PHP
 - Remove the '#' before "mod_fastcgi"
 - Remove the '#' before these lines:

```
fastcgi.server= ( ".php"=>
    ( "localhost" =>
        (
            "socket" => "/tmp/php-fastcgi.socket",
            "bin-path" => "/usr/local/bin/php-cgi"
        )
    )
)
```

Installing lighttpd (2)

- SSL support

- > ##### SSL engine
- > ssl.engine = "enable"
- > ssl.pemfile = "/path/server.pem"

- Virtual Hosting

- > Simple Virtual-Hosting

```
#simple-vhost.server-root =  
    "/home/weigon/wwwroot/servers/"  
#simple-vhost.default-host = "grisu.home.kneschke.de"  
#simple-vhost.document-root = "/pages/"
```

- > Enhanced Virtual-Hosting

- <http://trac.lighttpd.net/trac/wiki/Docs%3AModEVhost>

Installing lighttpd (3)

- ◉ /etc/rc.conf
 - > lighttpd_enable="YES"
- ◉ Start up
 - > /usr/local/etc/rc.d/lighttpd start