

# Apache/PHP/MySql



Relatore: Peppe Merola

Autori: Peppe Merola e  
MajagLUG

Evento: Linux Day 2006

Location: Genzano – RM Biblioteca  
Comunale



# Web Server

- Quando apriamo il nostro browser e digitiamo un indirizzo (tipo <http://www.majaglug.net>), facciamo una richiesta ad un server remoto
- Chiediamo, in pratica al server di farci vedere un file
- La comunicazione tra il browser e il server avviene tramite un canale, il protocollo HTTP (Hyper Text Transfer Protocol)

# Web Server

- Un Server Web quindi non è altro che un programma in esecuzione su una macchina (PC) collegata ad internet, che riceve una richiesta di una risorsa, e la restituisce al client (il browser)
- Un server web prende un file (.htm, .php, .asp, .jsp ...) lo elabora (se necessario), e restituisce il risultato al client sotto forma di testo formattato.
- Restituisce HTML.

# Web Server

- HTML restituito viene formattato dal browser.
- Il browser visualizza il contenuto del file richiesto.

# APACHE

- La storia di Apache inizia nel mese di febbraio del 1994, quando inizia il progetto del Gruppo Apache, basato sul server httpd della NCSA (National Center of Supercomputing Applications), Università dell'Illinois.
- Il nome, secondo la leggenda, nasce dal fatto che inizialmente il server era semplicemente una raccolta di patch da applicare al server NCSA.
- Apache = "a patchy server"

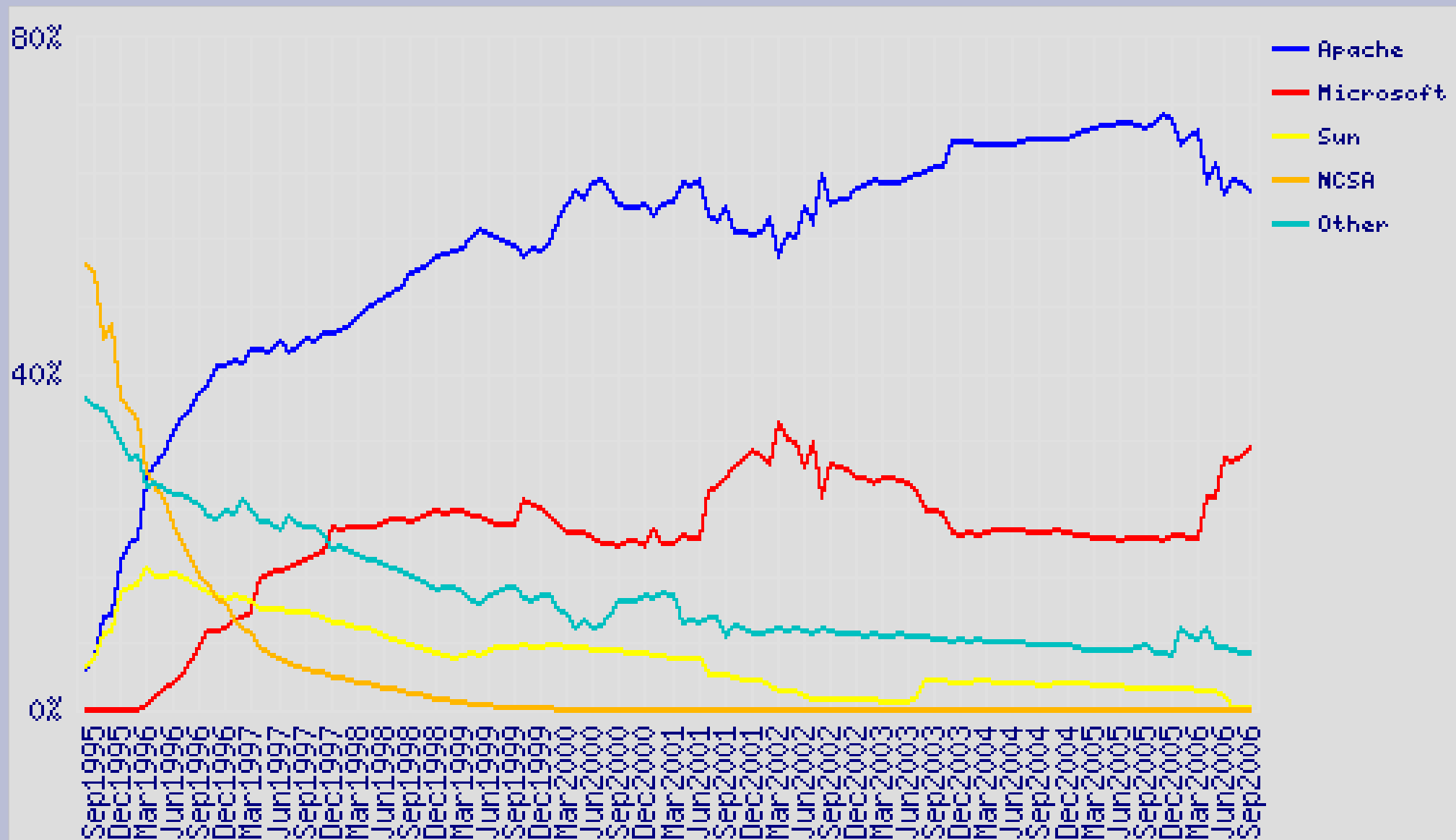
# APACHE

## Caratteristiche (1)

- **HTTP 1.1**
- **CGI**
- **SSL**
- **SSI**
- **.htaccess**
- **Alias e Redirect.**
- **VirtualHost.**
- **moduli dinamici (DSO  
Dynamic shared objects)**
- **multiprocess/multithreaded  
(standart POSIX)**

# Apache

## Diffusione secondo Netcraft





# PHP

- PHP è un linguaggio di scripting interpretato utilizzato principalmente per la realizzazione di applicazioni web **lato server** e pagine web dinamiche. Da qui il suo nome: un acronimo ricorsivo che sta per PHP: Hypertext Preprocessor (PHP: preprocessore di ipertesti).
- Nato nel 1994 ad opera del danese Rasmus Lerdorf, PHP era in origine una raccolta di script CGI (Personal Home Page)

# PHP - Caratteristiche

- PHP riprende per molti versi la sintassi C
- È un linguaggio a tipizzazione debole
- Supporta la programmazione ad oggetti (dalla ver 5).
- Supporta molti database tra cui MySQL, PostgreSQL, Oracle, Firebird, IBM DB2, Microsoft SQL Server
- Supporta numerose tecnologie, come XML, SOAP, IMAP, FTP, CORBA.

# PHP – Caratteristiche (2)

- Si integra anche con altri linguaggi e/o piattaforme quali Java e .NET.
- si può dire che esista un wrapper per ogni libreria esistente, come CURL, GD, Gettext, GMP, Ming, OpenSSL ed altro.
- E' possibile, grazie all'estensione php-gtk, sviluppare applicazioni con interfaccia grafica lato client.



# MySQL - Storia

- Nel 1996 un'azienda scandinava, la TcX, aveva bisogno di un database relazionale che potesse gestire grandi quantità di dati, ma che fosse allo stesso tempo molto veloce e sicuro e che potesse essere usato con poche risorse hardware. In quel periodo, trovare un prodotto che avesse queste caratteristiche pareva impossibile. Allora l'azienda decise di creare un proprio DBMS (DataBase Management System), partendo da un altro database ( mSql )

# MySQL - Teoria

- Un database e' una collezione di dati che viene gestita e organizzata da un software specifico chiamato DBMS (DataBase Management System, Sistema di Gestione di Basi Dati). Questo software permette la ricerca e la gestione delle informazioni in una struttura, che é definita con la banca dati.
- L'idea é simile ad un foglio di calcolo, i dati sono suddivisi in tabelle.

# MySQL – Teoria (2)

- Un DBMS ha delle funzioni di ricerca potentissime, che gli danno la capacità di trovare degli elementi (celle) in pochissimo tempo.
- Attualmente quasi tutti i DBMS si basano sul modello relazionale
- Per l'estrazione e manipolazione dei dati si utilizza un linguaggio chiamato SQL (Structured Query Language).

# MySQL – Teoria (3)

- MySQL è un RDBMS (Relational database management system - sistema relazionale per la gestione di basi di dati).

# MySQL - Caratteristiche

- Select, Insert, Delete, Update
- Viste
- Stored Procedures
- Trigger e Eventi
- una cache per le query
- ClusterDB
- Indici

# MySQL

## Storage Ufficiali

- **MyISAM**
- **InnoDB** (transazionale, svilup. da InnoDB Oy)
- **Memory** (una volta si chiamava Heap)
- **Merge**
- **NDB, o ClusterDB** (introdotta nella 5.0)
- **CSV** (introdotta nella 5.0)
- **Federated** (introdotta nella 5.0)
- **Archive** (introdotta nella 5.0)
- **Blackhole** (introdotta nella 5.1)

# MySQL

## Storage Non Ufficiali

- **PrimeBase XT Engine** - Motore transazionale progettato per applicazioni web ad alta concorrenza
- **solidDB** - Motore transazionale
- **RitmarkFS** - Permette di accedere al filesystem in lettura e in scrittura tramite comandi SQL, supporta anche la replica del filesystem
- **Distributed Data Engine** - Motore per dati distribuiti, per gestire meglio il carico di lavoro
- **mdbtools** - Permette di leggere e scrivere un file .mdb (Access)

# INSTALLAZIONE

La strada semplice:

- binari (.deb, .rpm, .tgz): da uno dei repository della vostra distribuzione

La strada meno semplice:

- ricompilare i sorgenti: [www.apache.org](http://www.apache.org), [www.mysql.com](http://www.mysql.com), [www.php.net](http://www.php.net)

# Installazione da binari

- Debian (deb):

```
# apt-get install apache2 mysql-server php5 php5-mysql
```

- RedHat (rpm):

```
# rpm -ivh apache2-xxxx.rpm php5-xxxx.rpm  
mysql-xxxx.rpm php5-mysql-xxx.rpm
```

```
# yum install httpd mysql-server php php-mysql
```

- Client grafico per l'installazione dei pacchetti (synaptic, yast, etc..)

# Installazione da binari (2)

The screenshot shows the Synaptic Package Manager window. The title bar reads "Gestore di pacchetti Synaptic". The menu bar includes "File", "Modifica", "Pacchetto", "Impostazioni", and "Ajuto". The toolbar contains icons for "Ricarica", "Marca tutti gli aggiornamenti", "Applica", "Proprietà", and "Cerca".

The main window is divided into two panes. The left pane, titled "Tutti", shows a list of packages with "apache2" selected. The right pane displays a table of installed and available packages.

S	Pacchetto	Versione installata	Ultima versione	Dim.
<input checked="" type="checkbox"/>	apache2	2.2.3-2	2.2.3-2	86
<input checked="" type="checkbox"/>	apache2.2-common	2.2.3-2	2.2.3-2	34
<input type="checkbox"/>	apache2-common		2.0.55-4.1	
<input type="checkbox"/>	apache2-doc		2.2.3-2	
<input type="checkbox"/>	apache2-mpm-event		2.2.3-2	
<input type="checkbox"/>	apache2-mpm-perchild		2.2.3-2	
<input checked="" type="checkbox"/>	apache2-mpm-prefork	2.2.3-2	2.2.3-2	6
<input type="checkbox"/>	apache2-mpm-threadpool		2.0.54-5sarge1	
<input type="checkbox"/>	apache2-mpm-worker		2.2.3-2	
<input type="checkbox"/>	apache2-prefork-dev		2.2.3-2	
<input type="checkbox"/>	apache2-threaded-dev		2.2.3-2	
<input checked="" type="checkbox"/>	apache2-utils	2.2.3-2	2.2.3-2	5

Below the table, there are tabs for "Descrizione", "Comune", "Dipendenze", "File installati", and "Versioni". The "Descrizione" tab is active, showing the text "Nessun pacchetto selezionato."

At the bottom of the window, there are buttons for "Sezioni", "Stato", "Search Results", and "Custom Filters". A status bar at the very bottom indicates: "71 pacchetti elencati, 1364 installati, 0 difettosi. 0 da installare/aggiornare, 0 da rimuovere".

# Installazione da sorgenti

- Si scompatta il file in una directory ( di solito con il comando `tar zxvf nomefile.tar.gz`)
- Si entra nella directory
- Si danno i comandi:
  - `./configure --opzione1 --opzione2`
  - `make`
  - `make install`

# Installazione da sorgenti (2)

## Esempio:

- `# tar zxvf httpd-2.2.3.tar.gz`
- viene creata la directory `httpd-2.2.3`
- `# cd httpd-2.2.3/`
- `# ./configure --prefix=/usr/lib/apache2 --enable-rewrite=shared`
- `# make`
- `# make install`

# Configurare Apache

## Directory

- /etc/apache2
- /usr/local/apache2/conf

I file di configurazione sono tutti **file di testo**.

- **f - apache2.conf**
- d - conf.d
- f - envvars
- f - httpd.conf
- f - magic
- **d - mods-available**
- **d - mods-enabled**
- f - ports.conf
- **d - sites-available**
- **d - sites-enabled**

# apache2.conf

- Contiene le direttive base che riguardano tutto il server web.

ServerRoot "/etc/apache2"

LockFile /var/lock/apache2/accept.lock

PidFile /var/run/apache2.pid

Timeout 300

KeepAlive On

MaxKeepAliveRequests 100

KeepAliveTimeout 15

User www-data

Group www-data

StartServers 5

MinSpareServers 5

MaxSpareServers 10

MaxClients 150

MaxRequestsPerChild 0

# sites-available/sites-enabled

- **sites-available:** contiene i file di configurazione del sito base e dei server virtuali.
- **sites-enabled:** contiene dei link simbolici ai file contenuti nella directory sites-available. Tutti i link presenti corrispondono a server virtuali attivi

# VirtualHost

## sites-available/default

**NameVirtualHost \***

**<VirtualHost \*>**

ServerAdmin webmaster@localhost

**DocumentRoot /var/www/**

**<Directory />**

Options FollowSymLinks

AllowOverride None

**</Directory>**

**<Directory /var/www/>**

Options Indexes FollowSymLinks MultiViews

AllowOverride None

Order allow,deny

allow from all

**</Directory>**

.....

**</VirtualHost>**



specificare la  
directory dove sono  
i nostri file html

# Avviare apache

- Dopo aver configurato un VirtualHost e impostato la directory dove posizioneremo il nostro sito, possiamo avviare apache con:

```
# /etc/init.d/apache2 start
```

oppure

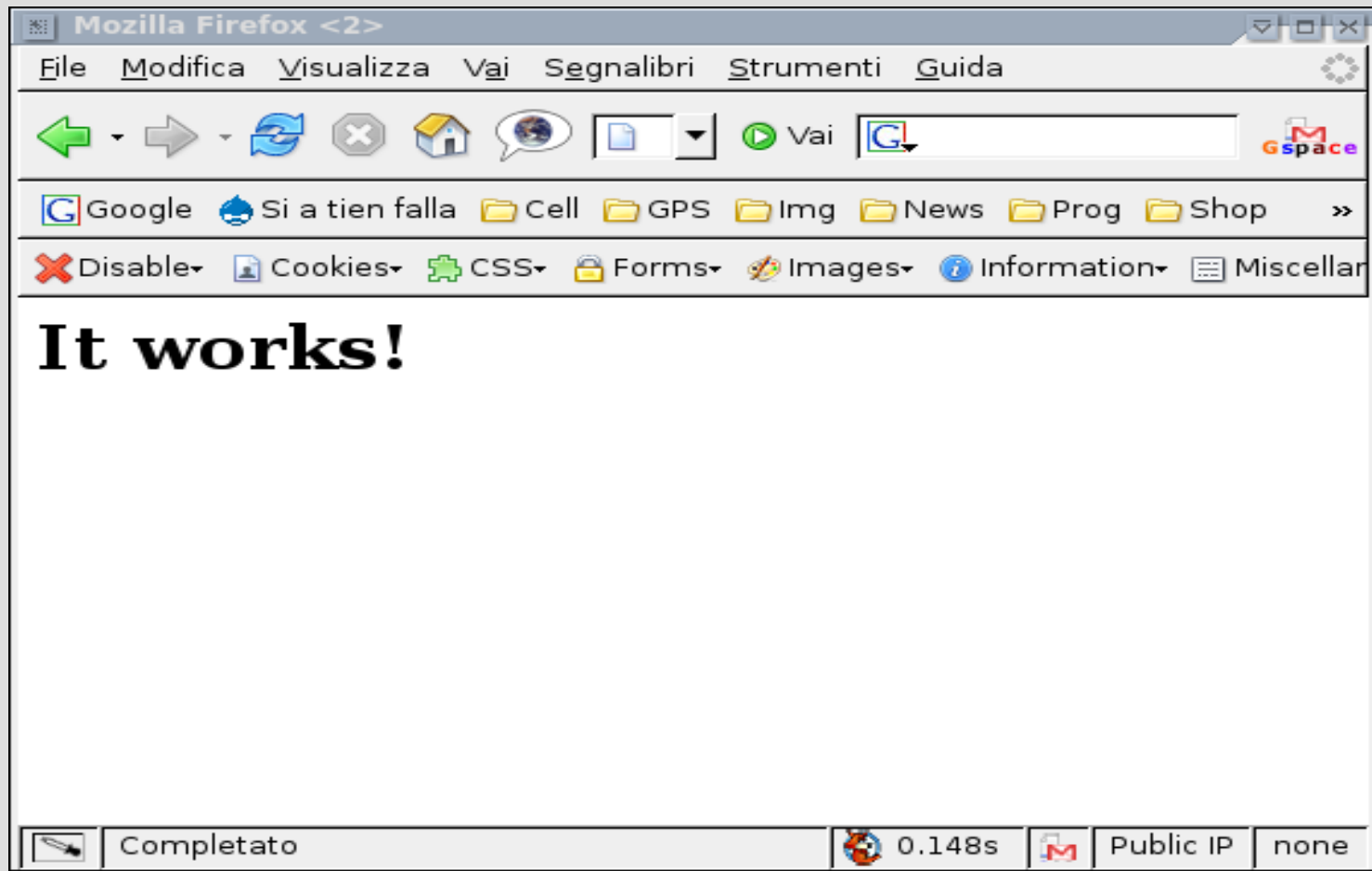
```
# /usr/local/apache2/bin/apache2ctl start
```

oppure

```
# apache2ctl start
```

# Verifica Funzionamento

- Aprire il browser
- Puntare all'indirizzo `http://localhost`



# Configurare PHP

- La configurazione del PHP si basa su un solo file **php.ini**
- E' un file di testo
- Si trova in **/etc/php5/apache2** o in **/usr/local/php5**

# php.ini

```
.....  
max_execution_time = 30;  
max_input_time = 60;  
memory_limit = 8M  
  
.....  
; UNIX: "/path1:/path2"  
;include_path = "./usr/share/php"  
; Windows: "\\path1;\path2"  
;include_path = ".;c:\php\includes"  
  
.....  
extension=gd.so  
extension=mysql.so  
extension=xmlrpc.so  
extension=mysqli.so
```

# Attivare il PHP

- Apache metta a disposizione delle utility per attivare e disattivare i moduli **a2enmod** e **a2dismod**:

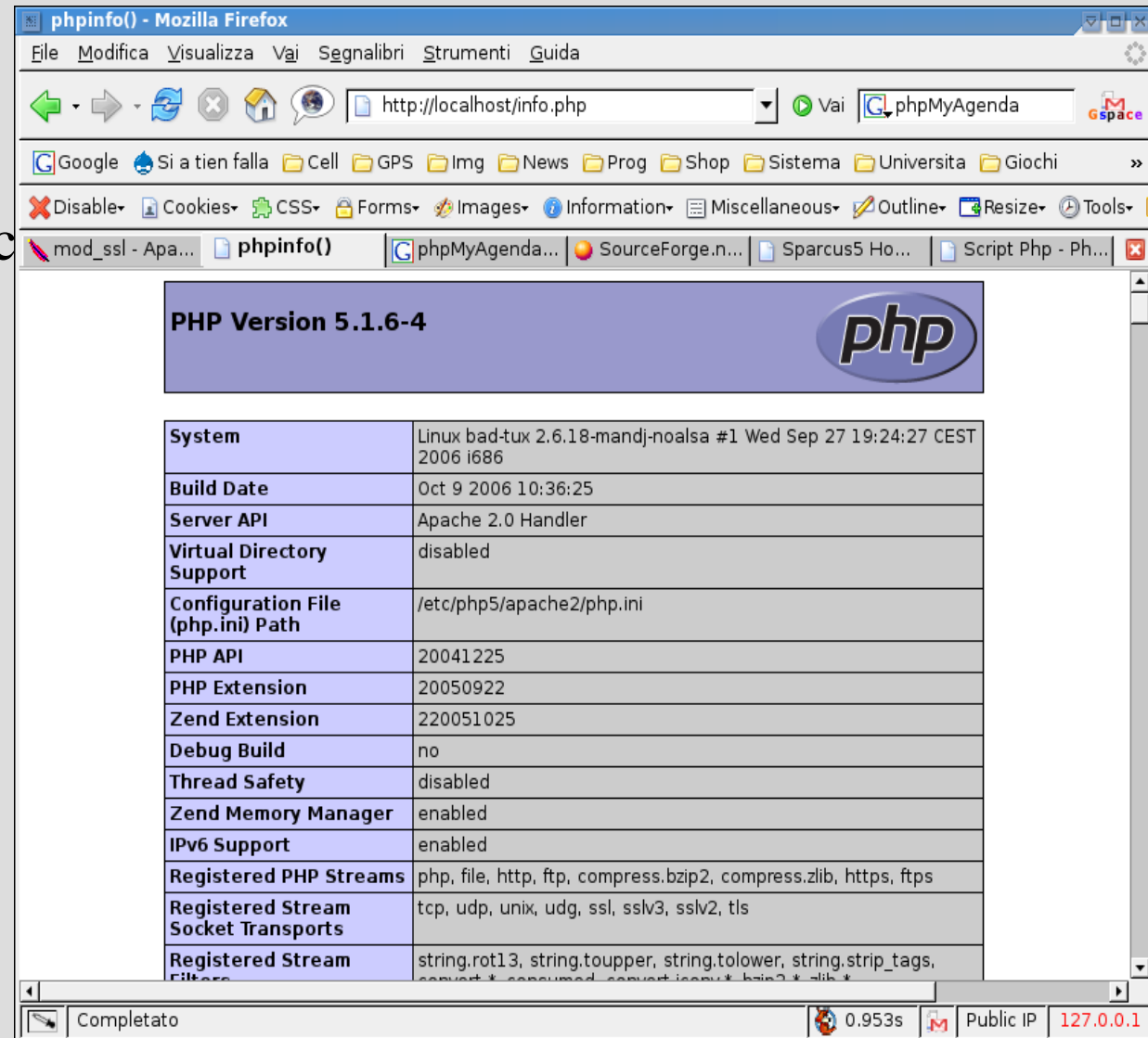
```
# a2enmod php5
```

Questa utility non fa altro che creare un collegamento ai file `/etc/apache2/mods-available/php5.conf` e `/etc/apache2/mods-available/php5.load` nella directory `/etc/apache2/mods-enabled`

- Riavviare apache

# Verifica Funzionamento

- Creiamo un file nella DocumentRoot: **info.php**
- Copiamo il seguente codice e salviamo
- `<?php phpinfo(); ?>`



System	Linux bad-tux 2.6.18-mandj-noalsa #1 Wed Sep 27 19:24:27 CEST 2006 i686
Build Date	Oct 9 2006 10:36:25
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2/php.ini
PHP API	20041225
PHP Extension	20050922
Zend Extension	220051025
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	php, file, http, ftp, compress.bzip2, compress.zlib, https, ftps
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, sslv2, tls
Registered Stream Filters	string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*-converted, convert.iconv.*-bz1c2.*-zlib.*

Completo 0.953s Public IP 127.0.0.1

# Configurare mysql

- La configurazione di MySQL si basa su un solo file **my.cnf**
- E' un file di testo
- Si trova in **/etc/mysql**
- Esempi di configurazione si trovano nella directory **/usr/share/doc/mysql-server-5.0/examples/:** my-small.cnf, my-medium.cnf, my-large.cnf.

# mysql.cnf

[client]

port = 3306

socket = /var/run/mysql/mysql.sock

.....

[mysqld]

user = mysql

pid-file = /var/run/mysql/mysql.pid

socket = /var/run/mysql/mysql.sock

port = 3306

basedir = /usr

datadir = /var/lib/mysql

tmpdir = /tmp

language = /usr/share/mysql/english

.....

# Avviare mysql

```
# /etc/init.d/mysql start
```

oppure

```
# /usr/local/mysql/bin/mysqld &
```

oppure

```
# mysqld &
```

Verifica

```
# mysql -u root mysql
```

```
mysql> show tables;
```

# Un passo importante

- L'utente **root** è l'amministratore del DB
- La password di default di **root** è **nulla**
- E' molto importante impostare una password per l'utente root.

```
mysql> SET PASSWORD
```

```
--> FOR root@localhost = PASSWORD('pluto');
```

```
Query OK, 0 rows affected (0.04 sec)
```

```
# mysql -u root mysql -p
```

# Esempio

```
# mysql -u root mysql -p
```

```
mysql> GRANT USAGE ON *.* TO 'utente'@'localhost'  
IDENTIFIED BY 'password';
```

```
mysql> FLUSH PRIVILEGES;
```

```
mysql> CREATE DATABASE prova;
```

```
mysql> show databases;
```

```
mysql> GRANT ALL PRIVILEGES ON prova.* TO  
'utente'@'localhost';
```

## Esempio (2)

```
# mysql -u utente mysql -p
```

```
mysql> CREATE TABLE nomi (cognome  
VARCHAR(20), nome VARCHAR(20));
```

```
mysql> INSERT INTO nomi VALUES('Pippo', 'Pluto');
```

```
mysql> INSERT INTO nomi VALUES('Rossi', 'Mario');
```

```
mysql> INSERT INTO nomi VALUES('Sasso', 'Luca');
```

# Esempio (3)

```
<?php
```

```
/* Connessione e selezione del database */
```

```
$connessione = mysql_connect("localhost", "demo", "demo") or  
die("Connessione non riuscita: " . mysql_error());
```

```
print "Connesso con successo";
```

```
mysql_select_db("prova") or die("Selezione del database non  
riuscita");
```

```
/* Esecuzione di una query SQL */
```

```
$query = "SELECT * FROM nomi";
```

```
$risultato = mysql_query($query) or die("Query fallita: " .  
mysql_error() );
```

# Esempio (4)

```
echo "<table border=\"1\">";
while ($res = mysql_fetch_row($risultato)) {
    list($cognome,$nome)=$res;
    echo "<tr><td>$cognome</td><td>$nome</td></tr>";
}
print "</table>";
mysql_free_result($risultato); /*Liberazione delle risorse del risultato
*/
mysql_close($connessione); /*Chiusura della connessione*/
```

?>

# Esempio (5)

The screenshot shows the Mozilla Firefox browser window. The title bar reads "Mozilla Firefox". The menu bar includes "File", "Modifica", "Visualizza", "Vai", "Segnalibri", "Strumenti", and "Guida". The address bar contains the URL "http://localhost/esempio2.php". The search bar is empty. The toolbar shows various icons for navigation and search. The page content displays the text "Connesso con successo" followed by a table of names:

Pippo	Pluto
Rossi	Mario
Sasso	Luca

The status bar at the bottom shows "Completato" on the left, and on the right, it displays "1.967s", "Public IP", and "127.0.0.1".

# Lista Funzioni PHP-MySQL

- `mysql_connect`
- `mysql_select_db`
- `mysql_query`
- `mysql_error`
- `mysql_fetch_row`
- `mysql_fetch_array`
- `mysql_affected_rows`
- `mysql_num_rows`
- `mysql_num_fields`
- `mysql_insert_id`
- `mysql_escape_string`
- `mysql_free_result`
- `mysql_close`

# Attivare SSL

Se vogliamo attivare SSL per la cifratura delle connessioni dobbiamo creare un file di configurazione per un nuovo VirtualHost con le seguenti direttive:

```
NameVirtualHost *:443
```

```
<VirtualHost *:443>
```

```
.....
```

```
SSLEngine on
```

```
SSLProtocol all
```

```
SSLCertificateFile /etc/apache2/ssl/server.crt
```

```
SSLCertificateKeyFile /etc/apache2/ssl/server.key
```

```
SSLCipherSuite ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:  
+MEDIUM:+LOW:+SSLv2:+EXP
```

```
.....
```

```
</VirtualHost>
```

# Creare Certificato

```
# openssl req -x509 -newkey rsa:2048 -keyout server.key \  
-out server.crt -days 365 -nodes
```

Copiamo server.key e server.crt nella cartella /etc/apache2/ssl

Abilitiamo il nuovo VirtualHost con **a2ensite**:

```
# a2ensite default-ssl
```

Facciamo ripartire apache:

```
# /etc/init.d/apache2 restart
```

https://localhost

# Riferimenti

- <http://httpd.apache.org/docs/2.2/>
- <http://www.php.net/>
- <http://www.mysql.com/>
- <http://www.openssl.org/>