

# Apache, PHP, MySQL, DRUPAL on Leopard (Mac Os X)

Di Giacomo Petronio  
<http://hijack.it>

## Apache

Mac Os X arriva con una versione di Apache preinstallata ed integrata nel sistema, attivabile dalle preferenze di sistema (Sharing, Web Sharing). Una volta avviato il servizio, Apache di default serve le pagine presenti in `/Library/WebServer/Documents/` sulla porta 80 a partire dalla root (raggiungibile quindi dall'indirizzo `http://localhost/`), inoltre serve anche le pagine personali dell'utente, presenti in `/Users/username/Sites/` e raggiungibili dall'indirizzo `http://localhost/~username/`

Qualcuno noterà, però, che mentre alla web root si accede tranquillamente, le pagine personali non vengono visualizzate per una questione di permessi (se invece siete fortunati e funziona tutto, saltate questa parte). Le cause di questo problema possono essere molteplici, adesso vediamo passo passo cosa va controllato:

- controllare che l'utente di apache sia `www` (o l'equivalente `_www`) nel file di configurazione `httpd.config`, cerca `User` e `Group` e impostarli entrambi a `www`.
- controllare che in `httpd.config` venga caricato il modulo `userdir`, scommentando la riga

```
LoadModule userdir_module libexec/apache2/mod_userdir.so
```

e che ci sia la riga

```
Include /private/etc/apache2/extra/httpd-userdir.conf
```

e che nel file `userdir.conf` ci sia

```
UserDir Sites  
Include /private/etc/apache2/users/*.conf
```

- controllare il contenuto di `/etc/apache2/users/username.conf`:

```
<Directory "/Users/username/Sites/">  
    Options Indexes MultiViews  
    AllowOverride All  
    Order allow,deny  
    Allow from all  
</Directory>
```

Prestare attenzione soprattutto ad *AllowOverride All* (per far sì che apache legga eventuali file `.htaccess` nella directory) ed *Allow from all* (per consentire l'accesso a tutti gli host)

- controllare che non ci sia un file `.htaccess` in una delle directory precedenti a `/Users/username/Sites` ed infine che tutte le directory del path `/Users/username/Sites` abbiano il flag `+X` per "others":

```
chmod o+x / /Users /Users/username /Users/username/Sites
```

questo è necessario per consentire ad Apache di listare il contenuto delle directory e sapere se ci sono file `.htaccess`

Riavviare Apache e testare puntando il browser su `http://localhost/~username/`

### I file di configurazione di apache:

```
/etc/apache2/httpd.config  
/etc/apache2/users/username.config  
eventuali .htaccess nelle directory servite da Apache
```

## PHP

L'installazione di default di Apache su Leopard ha anche il modulo per PHP5 (disattivato) che però

non contiene diverse librerie utili come le famose GD per la gestione delle immagini. Per aggiungere moduli si deve ricorrere alla compilazione dei sorgenti.

Per fortuna esiste un'altra soluzione, e cioè quella di utilizzare una versione già compilata di PHP5 per Leopard, raggiungibile dalla sezione download di [php.net](http://www.php.net), che ci reindirizza ad un server esterno (<http://www.entropy.ch/software/macosx/php/>). Da lì è possibile scaricare un comodo installer per PHP5 che si integra con il server Apache di sistema. Purtroppo però non sempre l'installazione va a buon fine (il pacchetto è per Mac OS X 10.4, ma con qualche ritocco funziona perfettamente anche su Leopard)

Come prima cosa scarichiamo il pacchetto (php5.2.4 per Apache 2) e scompattiamo da riga di comando (sul sito sconsigliano di utilizzare Stuffit!) l'archivio tar.gz:

```
tar -xvzf ./entropy-php-5.2.4-1-apache2.tar.gz
```

Assicuriamoci che in `httpd.conf` di Apache la riga:

```
#LoadModule php5_module          libexec/apache2/libphp5.so
```

sia commentata (per non creare conflitti con la versione php5 di sistema).

Creare il seguente link simbolico a cui l'installazione farà riferimento

```
sudo ln -s /etc/apache2 /etc/httpd
```

e rimuovi eventuali file e directory di entropy da `/Library/Receipts` (nel caso di tentate installazioni precedenti)

Ora possiamo fare doppioclick sul `.dmg` di entropy ed avviare l'installazione che dovrebbe andare a buon fine.

Riavviare Apache e testare, se ci sono problemi effettuare ancora queste operazioni:

```
sudo cp /usr/sbin/httpd /usr/sbin/httpd-fat
sudo lipo /usr/sbin/httpd -thin i386 -output /usr/sbin/httpd
```

e

```
sudo ln -s /usr/local/php5 /usr/local/apache2
sudo ln -s /usr/lib/libexpat.dylib /usr/local/apache2/lib/libexpat.0.dylib
```

(soluzione ricavata da <http://www.entropy.ch/phpbb2/viewtopic.php?t=3074>)

In questo modo dovremmo avere PHP5 funzionante, e i file installati in

```
/usr/local/php5/
```

Mentre il file di configurazione di php (`php.ini`) si trova in

```
/usr/local/php5/lib/php.ini
```

## MySQL

Procediamo ora con l'installazione di MySQL scaricando l'installer per il nostro sistema dalla sezione download del sito di [mysql.com](http://mysql.com).

Installare il pacchetto di MySQL, e tralasciamo il pacchetto di avvio automatico e del pannello di preferenze (a qualcuno funziona, ad altri no). L'installer ha posizionato i file del DBMS in

```
/usr/local/mysql
```

e possiamo avviare e fermare il servizio con i comandi

```
sudo /usr/local/mysql/bin/mysqld_safe
sudo /usr/local/mysql/bin/mysqladmin shutdown
```

L'utente di default è `root` con password vuota

## PhpMyAdmin

Scaricare i file dal sito di phpMyAdmin, e scompattare il tutto in una delle directory servite da Apache, ad esempio in /Users/username/Sites/phpMyAdmin. Ora dobbiamo creare un file di configurazione per consentire la connessione con MySQL. Iniziamo creando una directory config nella cartella di phpMyAdmin, e diamo i permessi in scrittura chmod o+w.

Puntare il browser allo script per la creazione di un file di configurazione:

es. `http://localhost/~jack/phpmyadmin/scripts/setup.php`

Cliccare sul pulsante per aggiungere un Server, e settare questi parametri:

```
Server hostname: localhost
Connection type: socket
Authentication type: config
User for config auth: root
Password for config auth: lasciare vuoto
```

Premere su Add, ed infine su Save (in basso, sotto “configuration”). Ora dovremmo avere un file nella cartella config, che va copiato nella directory superiore (nella directory base di phpMyAdmin).

Se da problemi, è possibile che non trovi il socket a cui connettersi al DBMS, in questo caso dobbiamo verificare il percorso: avviate mysql, e poi avviate la console di mysql:

```
sudo /usr/local/mysql/bin/mysql
```

quindi dare il comando *status* e controllare il valore di *UNIX socket*, copiatelo e incollatelo nel parametro Server socket quando create il file di configurazione di phpMyAdmin.

Ora potete puntare il browser alla directory di phpMyAdmin e cominciare a smanettare.

Nota: se dovete ripristinare un database a partire ad esempio da un dump (un file .sql), fate attenzione che esso non superi i 2 MB, altrimenti è necessario modificare il parametro `upload_max_filesize` presente nel file `php.ini` di php5 (/usr/local/php5/lib/php.ini)