

Installare apache e php su Linux

Di articoli trattanti questo argomento se ne trovano a migliaia in rete; ho voluto comunque scriverne una mia perchè più se ne trovano in giro meglio è, anche perchè sapere queste nozioni è cosa essenziale per un qualsiasi sviluppatore web che adoperi php e mysql.

In questo articolo illustrerò come installare apache 2 con il supporto per php e mysql.

Il sistema

Non c'è bisogno che vi dica che potete installare apache in qualsiasi distribuzione vogliate ma ora elencherò le specifiche dell'ambiente usato in questo articolo:

- Sistema operativo: Linux Slackware 10.1
- Apache: versione 2.0.54
- Php: versione 4.3.11
- MySql: versione 4.1.11 standard
- librerie GD: versione 2.0.33
- zlib: versione 1.2.2
- jpeg-6b: versione 6b
- freetype: versione 2.1.9

Ho preso la versione 2 di apache perchè sono stati corretti parecchi bug e sfrutta meglio le risorse del sistema e il threading.

Per quanto riguarda php ho preferito scegliere la versione 4 piuttosto che la 5 in quanto la nuova versione presenta troppe features a mio avviso inutili, non mi è piaciuto il tentativo degli sviluppatori di copiare java.

Per MySQL ho installato l'ultima release stabile.

Installazione di Apache e Php

Scaricati i sorgenti di Apache e PHP dalle location situate in fondo alla guida, decompressate i file dentro la directory /usr/src (o dove volete voi, io uso sempre questa perchè è la directory predisposta dal sistema per i sorgenti) con i comandi

```
tar -zxvf php-NN.tar.gz
tar -zxvf httpd-2_0_NN.tar.gz
```

Ora entrate nella cartella dei sorgenti di apache e lanciate lo script configure con l'opzione --enable-so, che renderà possibile installare in seguito php come modulo.

```
./configure --enable-so
```

Adesso i sorgenti sono configurati a dovere e siamo pronti per la compilazione (make) e l'installazione dei file compilati (make install) quindi lanciate i comandi in sequenza:

```
make
make install
```

Ora rechiamoci nella directory di php e prepariamoci ad installarlo. Dovremo lanciare il configure anche qui; dovremo specificarli di inserire il supporto per mysql, ovvero abilitare tutte le funzioni per gestire questo database, e indicargli dove trovare il binario di apxs di apache. Apxs è il binario che permette di installare i moduli all'interno di apache, nel nostro caso sarà PHP.

Il configure sarà perciò composto così:

```
./configure --with-apxs2=/usr/local/apache2/bin/apxs --with-mysql
```

Come potete intuire l'opzione `--with-mysql` è usata per abilitare il supporto mysql; mentre l'opzione `--with-apxs2` viene usata per specificare il percorso del binario di apxs, ovviamente io ho scritto il percorso di default dove risiede apache dopo l'installazione; se volete cambiare directory ad apache basta, in fase di configure, specificare l'opzione `-PREFIX=` seguita dal percorso desiderato.

Se volete installare le librerie GD, più sotto troverete la sezione apposita.

Ora dobbiamo copiare il file di configurazione di php all'interno della directory delle librerie di linux in modo da essere rintracciato dal motore PHP al momento del bisogno. Il file si chiama `php.ini` e sta sotto la directory dei sorgenti di PHP con il nome di `php.ini-dist`; perciò copiamo questo file dentro la directory `/usr/local/lib`

```
cp php.ini-dist /usr/local/lib/php.ini
```

Ora dobbiamo andare ad agire nel file di configurazione di Apache in modo che, quando siano richieste pagine con estensione `php`, vada ad interrogare l'interprete `php` che trasformerà le pagine in `html`. Andiamo nella directory `/usr/local/apache2/conf` ed editiamo con un editor di testo (se siete da console usate `vi`) il file `httpd.conf`

```
vi httpd.conf
```

ora cercate la riga dove dichiara i `LoadModule`

```
/LoadModule
```

vi porterà alla sezione dei `LoadModule`, sotto date in invio e scrivete

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

se aveste installato PHP 5 la riga sarebbe

```
LoadModule php5_module modules/libphp5.so
```

Ora cercate la sezione `AddType` nello stesso modo e, sempre sotto, scrivete

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

Con `LoadModule` ordinate ad apache di caricare il modulo PHP all'avvio, mentre con la direttiva `AddType` associate ai file di estensione `.php` e `.phtml` l'interprete PHP che processerà le pagine e le manderà al browser del navigatore.

Nel file di configurazione di apache le righe che inserirete non devono essere precedute dal cancelletto (`#`) altrimenti risulteranno come commenti e non verranno interpretate.

Ora potete far partire apache con questo comando

```
/usr/local/apache2/bin/apachectl start
```

e cominciare a sbizzarrirvi con le vostre applicazioni.

Installare MySQL

La scelta di questo DBMS è quasi obbligata quando si parla di PHP. Le librerie di `Php` appoggiano in pieno la struttura `MySQL` mettendo a disposizione del programmatore una serie di funzioni fatte apposta per tale DBMS.

Cominciamo con lo scaricare i sorgenti di `MySQL` dal link in fondo alla guida.

Una volta installato, `MySQL` avrà bisogno di fare operazioni su disco per processare le query SQL che gli script gli ordineranno; per fare questo, in un sistema unix, esso ha

bisogno di girare come utente. Perciò ora cominceremo con il creare l'utente per mysql. Vi premetto che su parecchi di questi comandi riceverete un messaggio del tipo “user already exist” o “group already exist”; questo vorrà dire che il vostro sistema, in fase di installazione, ha creato di default l'utente per un eventuale installazione di MySQL ma voi, a scanso di equivoci, eseguite comunque i comandi che vi illustrerò. Cominciamo col creare il gruppo per l'utente MySQL

```
groupadd mysql
```

una volta creato questo gruppo creeremo l'utente mysql

```
useradd -g mysql mysql
```

con l'opzione -g associamo l'utente al gruppo mysql. Ora rechiamoci nella directory /usr/local e copiamoci dentro il file con i sorgenti di mysql. Ora, sempre rimanendo nella directory /usr/local, decomprimiamo il file

```
tar -zxvf mysql_standard_VERSIONE.tar.gz
```

creiamo un link simbolico a questa directory in quanto il nome troppo lungo potrebbe darci problemi; perciò digitiamo il comando

```
ln -s /usr/local/mysql_standard_VERSIONE mysql
```

ora è come se avessimo una cartella chiamata mysql che però è una specie di “portale dimensionale” verso la cartella mysql_standard_VERSIONE. Entriamo dentro la cartella di MySQL

```
cd mysql
```

e siamo pronti a lanciare lo script per installare il db

```
scripts/mysql_install_db --user=mysql
```

come vedete ora il DBMS è installato presso l'utente mysql. Onde evitare problemi di sicurezza diamo il permesso della directory corrente a root

```
chown -R root .
```

Ora diamo il permesso della cartella data all'utente mysql così che possa effettuare tutte le operazioni del caso

```
chown -R mysql data
```

associamo la cartella corrente ora al gruppo mysql

```
chgrp -R mysql .
```

Ora siamo pronti per avviare MySQL con questo comando; mi raccomando di usarlo ogni volta che vorrete far partire MySQL:

```
bin/mysqld_safe --user=mysql &
```

Installare le librerie GD

Le librerie GD sono librerie molto potenti e, a mio parere, parecchio divertenti. Queste

librerie permettono, tramite script PHP, di creare o editare immagini a nostro piacimento. Non starò qui a spiegarne il funzionamento perchè anche io le sto studiando, per quello vi rimando al manuale; vi dirò solamente come installarle.

Dobbiamo prima di tutto procurarci, dai link in fondo all'articolo, i seguenti software e librerie:

- Librerie GD (ovviamente)
- zlib
- jpeg-6b
- freetype

Per l'installazione dei software su citati vi consiglio di leggervi in file INSTALL al loro interno, sono semplicissimi da installare. Mi raccomando solo una cosa; non installatele con la direttiva -PREFIX ma lasciatele installare nella loro location di default così sarà più semplice configurare PHP.

Installati questi software e librerie procediamo con il ricompilare PHP. Attenzione, non ci sarà bisogno di ricompilare anche apache perchè le modifiche interessano solo l'engine PHP.

Il comando di configurazione è esattamente come prima solo che dobbiamo aggiungere delle opzioni, e perciò figurerà così:

```
./configure --with-apxs2=/usr/local/apache2/bin/apxs --with-mysql --with-gd  
--enable-gd-native-ttf --with-png --with-zlib-dir --with-ttf --with-jpeg-dir  
--with-freetype-dir --with-xpm-dir
```

Non fate caso se io sono andato a capo, voi dovete scrivere tutto su una riga. Come vedete abbiamo abilitato il supporto per le librerie gd, poi il supporto per le librerie jpeg e per le librerie TrueFont in modo che possiamo utilizzare font esterni per scrivere nelle immagini. Dopo questa riga potete dare termine all'installazione compilando ed installando la nuova configurazione di PHP.

```
make  
make install
```

Ora stoppate apache con il comando

```
/usr/local/apache2/bin/apachectl stop
```

riavviate come spiegato in precedenza e sarete pronti per usare anche le librerie GD.

I link

Apache: <http://httpd.apache.org/download.cgi>

PHP: <http://www.php.net/downloads.php>

MySQL: <http://dev.mysql.com/downloads/>

Librerie GD: <http://www.boutell.com/gd/>

zlib: <http://www.zlib.net/>

jpeg-6b: <http://quox.org/install/graphics/jpeg-6b.html>

freetype: <ftp://cis.uniroma2.it/unix/packages/FREETYPE/freetype2/>

Spero che l'articolo vi possa essere stato utile e spero di essere stato abbastanza chiaro. Per qualsiasi problema di consiglio di leggere i relativi file INSTALL e README dei software; altrimenti potrete sempre contattarmi agli indirizzi che troverete sul sito.

Daniele Argento