

LAMP (Linux Apache Mysql PHP) on Slackware Linux

Fioretti Nicola

WEB: <http://slackmania.altervista.org>

Email: nicolafioretti@yahoo.it



Ho scritto questa guida nella speranza di facilitare il processo di installazione e configurazione di un ambiente LAMP con Slackware Linux. All'interno della guida troverete tutte le informazioni necessarie a tal fine e spero che esse siano sempre corrette e puntuali. Qualora trovaste incongruenze, errori e quant'altro vi prego di contattarmi al mio indirizzo email per segnalarmi eventuali problemi, suggerimenti e critiche in modo da fornire una guida sempre più completa, corretta e di facile comprensione a tutti gli utenti di questa splendida distribuzione. Ogni segnalazione è sempre ben accetta.

Fioretti Nicola

Indice degli Argomenti

1.0 Introduzione	
1.1 Il webservice Apache	(pag. 4)
1.2 Il linguaggio PHP	(pag. 4)
1.3 Il DataBase Server MySQL	(pag. 4)
2.0 Mettiamoci al lavoro	(pag. 5)
2.1 Vantaggi e Svantaggi nell'utilizzo dei pacchetti precompilati	(pag. 5)
2.2 Reperire ed Installare i pacchetti	(pag. 5)
3.0 Funzionamento e Configurazione di Apache	(pag. 6)
3.1 Gestione del demone di Apache	(pag. 6)
3.1.1 Avvio/Arresto e Riavvio tramite apachectl	(pag. 6)
3.1.2 Avvio/Arresto e Riavvio tramite rc.httpd	(pag. 6)
3.2 Testare il funzionamento di Apache	(pag. 7)
3.3 Avvio automatico al boot di Apache	(pag. 7)
4.0 Configurazione di Apache	(pag. 8)
4.1 DirectoryIndex	(pag. 9)
4.2 DocumentRoot	(pag. 9)
4.3 Abilitare il supporto di Apache per PHP	(pag. 10)
4.4 Tool Grafici per la configurazione di Apache	(pag. 10)
5.0 Configurazione di MySQL	(pag. 11)
5.1 Avvio automatico al boot di MySQL	(pag. 12)
5.2 Tool Grafici per la gestione di MySQL	(pag. 12)
5.2.1 Installazione di phpMyAdmin	(pag. 12)

1.0 Introduzione

LAMP è l'acronimo della una piattaforma di sviluppo web che prende il nome dalle iniziali dei componenti software con cui viene realizzata, ovvero GNU/Linux, Apache, MySQL e PHP/Perl/Python. La combinazione più nota ed utilizzata risulta comunque quella che prevede GNU/Linux Apache MySQL e PHP che vedremo come realizzare in questa breve guida. Nello specifico ci occuperemo di analizzare l'installazione della nostra infrastruttura LAMP sotto Slackware Linux.

Tengo a precisare che in questa guida non affronteremo l'argomento della configurazione ottimale in termini di sicurezza dei server Apache e MySQL ma piuttosto verranno fornite all'utente sviluppatore tutte le informazioni necessarie per ottenere un ambiente correttamente configurato e pronto all'uso.

1.1 Il webserver Apache

Apache è un potente webserver per il protocollo HTTP sviluppato dall'Apache Group, nato come rimpiazzo del webserver httpd sviluppato dalla NCSA (National Center for Supercomputing Applications) del quale, la stessa NCSA aveva cessato lo sviluppo nel 1994. Apache eredita le caratteristiche dal suo predecessore httpd e introduce correzioni e migliorie rispetto a quest'ultimo.

Apache risulta un progetto davvero di ottimo livello ed offre caratteristiche davvero interessanti come ad esempio:

- Apache è un software Open Source
- Apache è multiplatforma
- Apache è sviluppato secondo le più aggiornate specifiche per i servizi HTTP

Attualmente Apache risulta il webserver più utilizzato al mondo e secondo le statistiche offerte da netcraft risulta che ad oggi (settembre 2005) più del 69% dei webserver del pianeta si affidano a questo webserver.

Per maggiori informazioni e documentazione vi invito a consultare la Home Page del progetto <http://www.apache.org/>

1.2 Il linguaggio PHP

PHP è l'acronimo ricorsivo di "PHP: Hypertext Preprocessor" ovvero "PHP: preprocessore ipertestuale". I nostri script infatti non sono elaborati dal client (browser) ma vengono eseguiti direttamente sul server il quale invia ai client pagine HTML. Questa tipologia di linguaggi comunemente viene definita server-side (lato server). Il punto di forza di questo linguaggio è rappresentato dal fatto che è un linguaggio estremamente semplice da imparare ma che offre possibilità avanzate per il programmatore esperto.

Una delle caratteristiche più importanti ed apprezzate di PHP è la possibilità di interazione tra vari tipi di database, tra qui il DBMS MySQL con il quale va davvero a braccetto. Grazie a PHP creare pagine collegate a dei DB risulta un'operazione davvero elementare.

Per maggiori informazioni e documentazione vi invito a consultare la Home Page del progetto <http://www.php.net/>

1.3 Il DataBase Server MySQL

MySQL è il DBMS (*Data base management system*) Open Source più diffuso ed utilizzato al mondo. La sigla SQL presente all'interno del nome sta a indicare che si tratta di un DBMS in grado di interpretare le istruzioni del linguaggio SQL.

MySQL può essere considerato come un "database relazionale" che consente la conservazione dei dati in tabelle separate anziché in un'unica grande entità.

Questa sua caratteristica consente di raggiungere una buona flessibilità e velocità di

accesso ai dati ed una maggior modellazione delle basi dati.

Il sistema si presenta come una struttura con architettura di tipo client/server. Il server SQL utilizza una architettura multi-thread ed offre una completa raccolta di API utilizzabili tramite i client offerti all'interno della distribuzione MySQL stessa, sia da applicazioni esterne appositamente scritte.

Per maggiori informazioni e documentazione vi invito a consultare la Home Page del progetto <http://www.mysql.com>

2.0 Mettiamoci al lavoro...

Dopo aver visto cosa andremo ad installare e aver capito che cosa significa LAMP, cominciamo ad analizzare l'installazione del nostro ambiente LAMP.

Trattandosi di software Open Source possiamo procedere in due modalità: installazione da sorgenti e installazione da pacchetti precompilati.

In questa guida vedremo come procedere con l'installazione da pacchetti precompilati che risulta più snella e semplice rispetto alla prima, ma nessuno vieta di installare tutto il software necessario direttamente da sorgenti.

2.1 Vantaggi e Svantaggi nell'utilizzo dei pacchetti precompilati

Il primo vantaggio tangibile nell'utilizzo di pacchetti precompilati è dato dal fatto che l'utente salta completamente il processo di compilazione evitandosi alcune "noie", specie se l'utente è alle prime armi. Un altro vantaggio è dato dal tempo di installazione che risulta nettamente inferiore.

Questa tipologia di installazione, comincerete a pensare, sembra che sia in grado di offrire una soluzione a tutti i nostri problemi, è semplice, veloce... Cosa volere di più? ...In effetti l'installazione da pacchetti precompilati nella stragrande maggioranza dei casi è davvero un ottimo compromesso, ma se andiamo ad indagare nel profondo scopriremo che soffre di alcune limitazioni.

L'uso di pacchetti precompilati infatti non consente di ottenere un'installazione personalizzata, ad esempio se avessimo particolari esigenze come ad esempio quella di utilizzare una particolare estensione potremmo non trovarla all'interno dei pacchetti precompilati. In secondo luogo non è possibile selezionare percorsi di installazione specifici.

Nel complesso però, possiamo affermare che per un utilizzo normale l'installazione da pacchetti compilati è la scelta migliore, in quanto contribuisce anche a mantenere un sistema pulito.

2.2 Reperire ed Installare i pacchetti

Cominciamo subito con il dire che se avete portato avanti un'installazione di tipo "FULL" di Slackware avete già installato sul vostro sistema tutto il software necessario per cominciare a lavorare con il vostro ambiente LAMP che dovrà solo essere configurato in alcuni suoi aspetti, in caso contrario dovrete procurarvi ed installare i pacchetti necessari che sono comunque presenti all'interno dei CD di Slackware.

I pacchetti necessari per installare il nostro ambiente LAMP sono i seguenti:

mysql-4.1.14-i486-1.tgz (DBMS MySQL, SQL-based relational database server)

apache-1.3.33-i486-1.tgz (The Apache HTTP Server)

php-4.4.0-i486-4.tgz (HTML-embedded scripting language)

I pacchetti ufficiali di Slackware Linux vengono di norma suddivisi in categorie, che corrispondono ciascuna ad una sottodirectory dei cdrom di installazione o dei mirror da cui scaricare i programmi. Per Apache e PHP la categoria è indicata dalla lettera "n", che sta per "net" (applicativi di rete), mentre MySQL si trova all'interno della categoria "ap" (application).

Ricercate quindi questi pacchetti all'interno dei CD di Slackware oppure sui mirror della

distruzione e procedete con la loro installazione come segue:

#installpkg nome-pacchetto.tgz

Dopo l'installazione dei tre pacchetti avrete a disposizione tutto il necessario.

In alternativa, qualora voleste essere sicuri di avere a vostra disposizione tutto il software necessario aggiornato all'ultima sua release potete scaricare i pacchetti precompilati da LinuxPackages.net (<http://linuxpackages.net>).

3.0 Funzionamento e Configurazione di Apache

Dopo aver installato l'HTTP Server Apache è bene cominciare a capire come funziona, come utilizzarlo e come configurarlo in base alle nostre esigenze.

3.1 Gestione del demone di Apache

Sotto Slackware, Apache, di default, gira come demone (httpd) in esecuzione continua in background ("Standalone"). Se nel file di configurazione la listening port è la 80 (o comunque una porta sotto la 1024), per avviare il demone è necessario avere i privilegi di root.

3.1.1 Avvio/Arresto e Riavvio tramite apachectl

Per gestire il demone httpd di apache viene spesso utilizzato lo script apachectl che tramite tre semplici parametri passati direttamente ad esso consentono l'avvio e l'arresto dell'HTTP Server in maniera molto semplice. Lo script apachectl viene utilizzato anche per verificare la correttezza della configurazione del server ed è presente in ogni distribuzione in quanto distribuito assieme ai sorgenti di Apache.

- **Avvio di Apache**
#apachectl start
- **Arresto di Apache**
#apachectl stop
- **Riavvio di Apache**
#apachectl restart

Come abbiamo detto lo script apachectl può controllare anche la configurazione di Apache. L'opzione "**configtest**" esegue un test di sintassi della configurazione. "**ConfigTest**" non controlla dal punto di vista semantico la configurazione di Apache, che deve essere fatta manualmente consultando i file di LOG che riporteranno eventuali errori.

- **Verifica della configurazione di Apache**
apachectl configtest

3.1.2 Avvio/Arresto e Riavvio tramite rc.httpd

Slackware Linux consente di gestire il demone di Apache tramite un apposito script che risiede all'interno della directory /etc/rc.d, chiamato rc.httpd. Questo script consente l'avvio, l'arresto e il riavvio di Apache in questo modo:

- **Avvio di Apache**
#/etc/rc.d/rc.httpd start
- Arresto di Apache**

```
# /etc/rc.d/rc.httpd stop
```

- **Riavvio di Apache**

```
# /etc/rc.d/rc.httpd restart
```

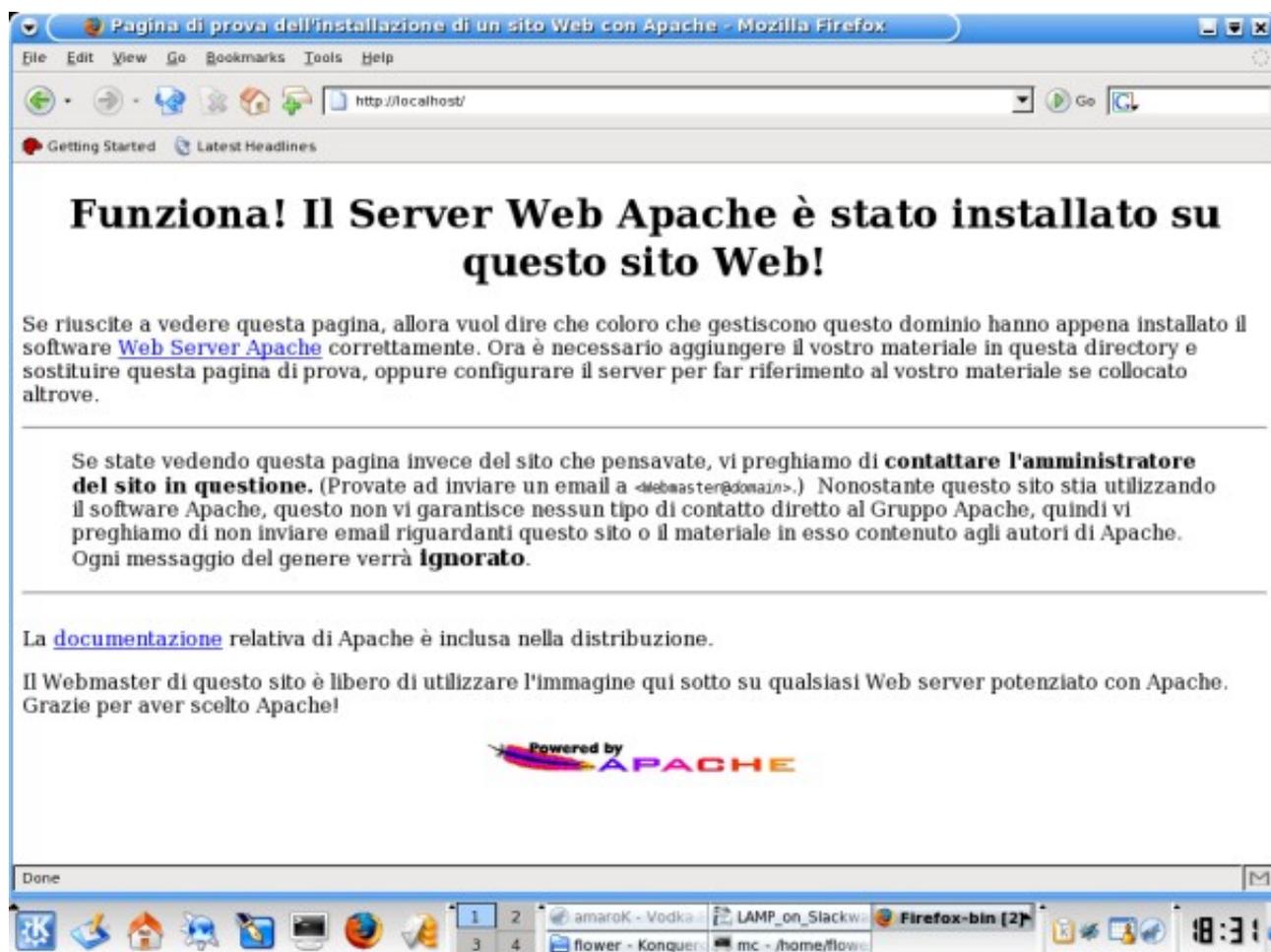
3.2 Testare il funzionamento di Apache

Dopo l'installazione e la gestione del demone di Apache (httpd) vediamo come testare il corretto funzionamento di Apache.

Innanzitutto è necessario avviare il demone httpd in uno dei modi sopra descritti, se l'avvio del demone avviene in maniera corretta apriamo il nostro browser preferito e nella barra degli indirizzi digitiamo il seguente URL:

<http://localhost>

A questo punto se tutto funziona dovrebbe essere visualizzata la schermata di benvenuto del server Apache (vedi figura sotto) che ci comunica il suo corretto funzionamento.



3.3 Avvio automatico al boot di Apache

Slackware utilizza un sistema di avvio in stile ***BSD** che risulta diverso dallo stile **SySV** utilizzato dalla stragrande maggioranza delle altre distribuzioni GNU/Linux.

Per attivare o disattivare l'avvio del demone di Apache in fase di boot è sufficiente impostare o meno il bit relativo all'esecuzione dello script **/etc/rc.d/rc.httpd**.

L'avvio automatico di apache in fase di boot avviene lanciando il comando:

```
#chmod +x /etc/rc.d/rc.httpd
```

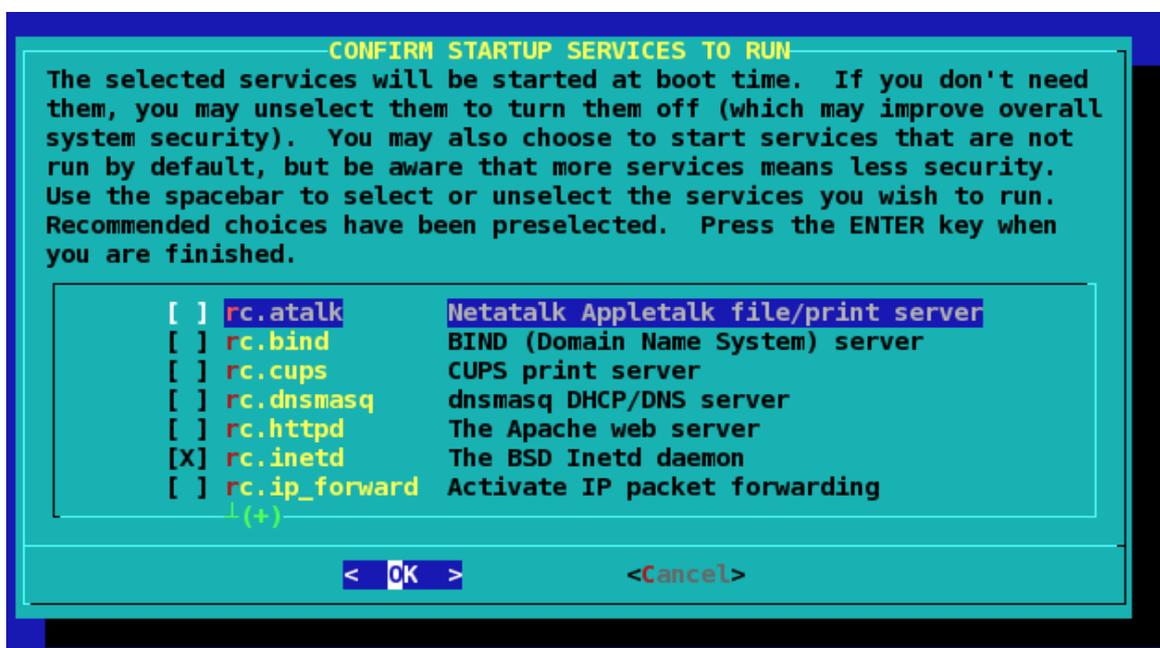
mentre se volessimo disattivare l'avvio automatico di questo demone all'avvio del sistema basta digitare:

```
#chmod -x /etc/rc.d/rc.httpd
```

Lo stesso risultato può essere ottenuto utilizzando pkgtool. Avviare pkgtool, lanciando con privilegi di root il comando

```
#pkgtool
```

Spostarsi sul menu "Setup" e premere INVIO. A questo punto si aprirà la schermata relativa a tutti i settaggi che è possibile configurare con pkgtool. A noi interessa agire sull'avvio dei servizi in fase di avvio del sistema operativo e quindi portiamoci sul menu "Services" e premiamo la barra spaziatrice seguita da INVIO. A questo punto si aprirà la seguente schermata:



Selezioniamo quindi la voce "rc.httpd" e premiamo la barra spaziatrice seguita da INVIO. Da questo momento in poi Apache verrà avviato automaticamente in fase di avvio del sistema operativo.

4.0 Configurazione di Apache

La configurazione dell'HTTP Server Apache è un compito abbastanza delicato e richiede conoscenze approfondite in ambito networking, in programmazione web e in ambito sistemistico.

Visto lo scopo della guida che è quello di introdurre l'argomento LAMP e configurare questo ambiente in ambito casalingo e comunque non professionale ci limiteremo a vedere una configurazione funzionale e a grandi linee di questo potentissimo Server. I file di configurazione di Apache si trovano all'interno della directory **/etc/apache/**.

All'interno di questa directory è presente il file **httpd.conf**, che al suo interno presenta una miriade di opzioni possibili da settare. Visto il numero davvero consistente di opzioni disponibili è consigliato fare riferimento al manuale ufficiale di Apache, che offre

una documentazione completa e puntuale su ogni parametro. Di tutte le opzioni disponibili noi andremo a modificarne ed a analizzarne solo le principali. Editiamo il file `httpd.conf`, come utente `root`, lanciando ad esempio il comando:

```
#mcedit /etc/apache/httpd.conf
```

4.1 DirectoryIndex

Questa direttiva specifica i nomi dei file che Apache considera come home page all'interno di una directory. Sostanzialmente quando viene fatto un GET che non indica il nome di un file specifico ma solo della directory Apache cerca al suo interno i file specificati con questa direttiva, nell'ordine indicato e visualizza la eventuale pagina trovata.

Ad esempio quando digitiamo <http://slackmania.altervista.org> Apache, ricerca all'interno di questa directory se sono presenti file `index.html` o `index.php` e li carica come home page del sito.

Nel nostro caso, visto che stiamo creando il nostro ambiente LAMP è di fondamentale importanza indicare tra i "**DirectoryIndex**" **index.php**, ovvero la home page del sito creato in PHP appunto.

E' possibile elencare più di un nome di file come nell'esempio qui sotto. Questi nomi hanno priorità progressiva: nel caso indicato qui sotto Apache mostra `index.html` se esiste, altrimenti `index.php`.

```
# DirectoryIndex: Name of the file or files to use as a pre-written HTML
# directory index. Separate multiple entries with spaces.
#
<IfModule mod_dir.c>
DirectoryIndex index.html index.php
</IfModule>
```

Detto questo modifichiamo il nostro file `/etc/apache/httpd.conf` come segue:

```
# DirectoryIndex: Name of the file or files to use as a pre-written HTML
# directory index. Separate multiple entries with spaces.
#
<IfModule mod_dir.c>
DirectoryIndex index.php index.php3 index.html index.htm
</IfModule>
```

In questo modo apache visualizzerà come home page la pagina chiamata "`index.php`" se esiste, altrimenti cercherà la pagina di nome "`index.php3`" poi eventualmente la pagina "`index.html`" e a seguire "`index.htm`".

4.2 DocumentRoot

Questa direttiva di `httpd.conf` specifica la directory entro la quale sono contenuti i file (pagine html, php ecc.) che contengono il nostro sito, ovvero tutti i file che saranno inviati al browser qualora li richieda. Di default questa directory è `"/var/www/htdocs"`, ma possiamo settarla a nostro piacimento ed in base alle nostre necessità.

Ad esempio possiamo settarla in questo modo:

```
DocumentRoot="/home/flower/web"
```

4.3 Abilitare il supporto di Apache per PHP

Visto che il nostro scopo è quello di creare un ambiente LAMP, si rende necessario abilitare il supporto del linguaggio PHP per il Server HTTP Apache, in modo che questo possa interpretare gli script “server-side” che andremo a creare. Per abilitare suddetto modulo portatevi alla sezione “mod_php configuration settings” presente all'interno del file httpd.conf e decommentate la riga relativa all'include. (Dal punto di vista pratico si tratta di togliere semplicemente il carattere “#” presente all'inizio della riga relativa all'include).

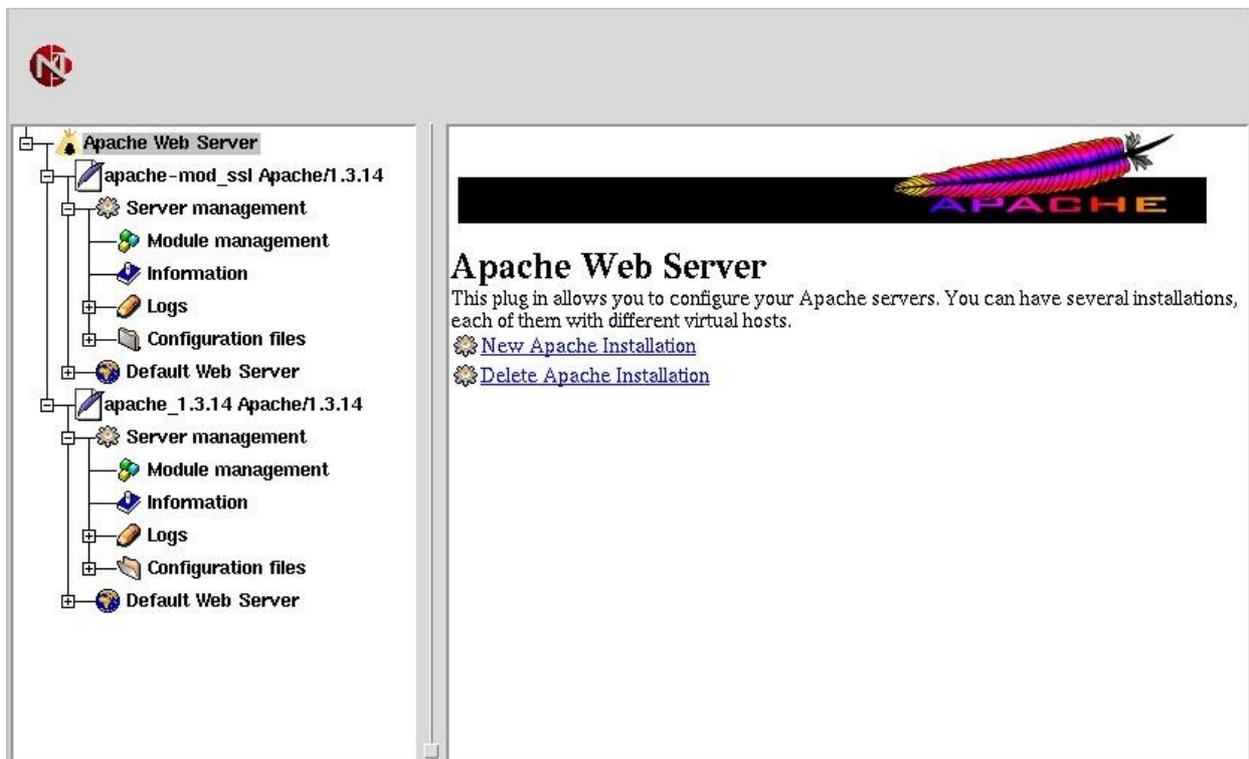
```
# ==> mod_php configuration settings <==  
#  
# PACKAGES REQUIRED: openssl-solibs (A series) and/or  
openssl (N series),  
# mysql (AP series), gmp (L series), and apache (N  
series)  
#  
#Include /etc/apache/mod_php.conf
```

4.4 Tool Grafici per la configurazione di Apache

La configurazione di Apache può avvenire nel metodo sopraesposto e quindi editando manualmente il file di configurazione di apache (/etc/apache/httpd.conf) oppure è possibile utilizzare dei tool grafici appositamente creati come “Comanche” (<http://comanche.org>)

Comanche è un tool grafico rilasciato sotto licenza GPL sviluppato “ad-hoc” per l'HTTP Server Apache. Comanche consente la configurazione di Apache tramite una comoda GUI che va direttamente a modificare il file httpd.conf.

L'installazione di questo tool è davvero semplice e consiste nello scaricare ed estrarre il pacchetto .tar.gz che contiene il binario e tutto il necessario al suo funzionamento.



5.0 Configurazione di MySQL

Dopo aver installato il DBMS MySQL è necessario procedere con la sua configurazione prima di poterlo utilizzare. Innanzitutto in base all'hardware a nostra disposizione e alle dimensioni presunte che avrà il database scegliamo in file di configurazione adatto tra quelli proposti che sono:

```
/etc/my.huge.cnf  
/etc/my.large.cnf  
/etc/my.medium.cnf  
/etc/my.small.cnf
```

Una volta individuato quello che fa al caso nostro dobbiamo rinominarlo in **/etc/my.cnf** (Per maggiori informazioni sui file di configurazione proposti potete visitare il sito ufficiale <http://www.mysql.com>). Indicativamente diciamo che il file **/etc/my.medium.cnf** è una scelta abbastanza buona nella maggior parte dei casi, quindi procediamo lanciando il comando:

```
# cp /etc/my-medium.cnf /etc/my.cnf
```

A questo punto possiamo procedere con l'installazione dei database mysql e test, lanciando il comando:

```
#mysql_install_db
```

Ultimo passo per vedere il nostro DBMS funzionante è quello di impostare correttamente i permessi sulla directory **/var/lib/mysql** lanciando il seguente comando:

```
#chown -R mysql:mysql /var/lib/mysql
```

Adesso siamo finalmente pronti ad avviare il server. Per avviarlo è sufficiente lanciare il seguente comando:

```
#mysqld_safe &
```

Impostiamo ora la password per l'utente root (ovviamente stiamo parlando dell'utente root di MySQL che nulla ha a che vedere con l'utente root del sistema) lanciando il comando:

```
# mysqladmin -u root password 'tua_password'
```

La password che intendete impostare va inserita al posto di 'tua_password' e scritta tra virgolette come nell'esempio sopra riportato. Dopo aver impostato la password per l'utente root, per avere un minimo di sicurezza in più, killiamo tutti i processi mysql. D'ora in avanti possiamo avviare manualmente MySQL con il comando:

```
#mysql_safe --user=mysql &
```

Una volta avviato il DBMS MySQL è possibile accedere alla console di MySQL digitando il comando:

```
#mysql -p
```

che avvierà la parte client del DBMS con privilegi di root. Tramite la console possiamo

creare, modificare, eliminare DB, tabelle ecc...

Per ovvi motivi di sicurezza è opportuno creare un utente con altri privilegi da utilizzare per lavorare con il nostro DBMS.

5.1 Avvio automatico al boot di MySQL

Come abbiamo già visto per Apache, Slackware utilizza un sistema di avvio in stile *BSD che consente di attivare o disattivare l'avvio di un demone in fase di boot semplicemente abilitando i permessi di esecuzione sullo script relativo al demone che ci interessa avviare.

Per quanto riguarda MySQL dobbiamo comportarci come segue:

L'avvio automatico di MySQL in fase di boot avviene lanciando il comando:

```
#chmod +x /etc/rc.d/rc.mysql
```

mentre se volessimo disattivare l'avvio automatico di questo demone all'avvio del sistema basta digitare:

```
#chmod -x /etc/rc.d/rc.mysql
```

Lo stesso risultato può essere ottenuto utilizzando pkgtool in maniera analoga a quanto visto per Apache.

5.2 Tool Grafici per la gestione di MySQL

In questo paragrafo vedremo come sia possibile interagire e gestire in maniera semplice e veloce il nostro DBS MySQL con "phpMyAdmin".

Come suggerisce il nome stesso, phpMyAdmin è completamente implementato in PHP e quindi richiede la presenza di Apache e PHP per il suo corretto funzionamento. Grazie a questo tool Open-Source è possibile gestire un intero server MySQL in tutti i suoi aspetti o un singolo DB.

Le operazioni che possiamo intraprendere con phpMyAdmin sono molteplici, tra cui ricordiamo:

- Creazione/Cancellazione di un Database
- Creazione/Modifica/Cancellazione di una tabella
- Eseguire qualsiasi istruzione SQL
- Creare/Leggere dump di tabelle
- Amministrare gli utenti
- Amministrare Server multipli

e tanto altro ancora...

5.2.1 Installazione di phpMyAdmin

La prima cosa da fare è procurarsi phpMyAdmin scaricandolo dal sito ufficiale del progetto <http://www.phpmyadmin.net> (Attualmente l'ultima release di questo tool è la 2.6.4)

A questo punto decomprimete l'archivio .tar.gz lanciando semplicemente il comando

```
#tar zxvf phpMyAdmin-2.6.4-pl1.tar.gz
```

Una volta estratto il file otterrete una directory chiamata phpMyAdmin-2.6.4 contenente tutti i file che compongono questo tool, tra cui il file di configurazione chiamato "config.inc.php" al quale apporteremo qualche modifica per settare alcune opzioni in base alle nostre esigenze.

Editiamo il file config.inc.php e ricerchiamo la sezione “Your phpMyAdmin URL” (che possiamo vedere sotto) la quale ci consentirà di settare il nome della directory all'interno della quale abbiamo installato il phpMyAdmin (ad esempio <http://slackmania.altervista.org/phpMyAdmin/>).

```
/**
 * Your phpMyAdmin URL.
 *
 * Complete the variable below with the full url ie
 * http://www.your_web.net/path_to_your_phpMyAdmin_directory/
 *
 * It must contain characters that are valid for a URL, and the path is
 * case sensitive on some Web servers, for example Unix-based servers.
 *
 * In most cases you can leave this variable empty, as the correct value
 * will be detected automatically. However, we recommend that you do
 * test to see that the auto-detection code works in your system. A good
 * test is to browse a table, then edit a row and save it. There will be
 * an error message if phpMyAdmin cannot auto-detect the correct value.
 */
$cfg['PmaAbsoluteUri'] = '';
```

Dopo aver settato questo parametro portiamoci alla sezione “Server(s) configuration” (vedi tabella sotto riportata) e procediamo inserendo le informazioni relative all'host ed all'utente per l'accesso a MySQL. In particolare andremo a modificare le seguenti impostazioni:

- `$cfg['Servers'][$i]['host']` = 'localhost';
- `$cfg['Servers'][$i]['auth_type']` = 'config';
- `$cfg['Servers'][$i]['user']` = 'root';
- `$cfg['Servers'][$i]['password']` = '';

```
/**
 * Server(s) configuration
 */
$i = 0;
//The $cfg['Servers'] array starts with $cfg['Servers'][1]. Do not use $cfg['Servers'][0].
// You can disable a server config entry by setting host to ''.
$i++;
$cfg['Servers'][$i]['host'] = 'localhost'; // MySQL hostname or IP address
....
```

Dopo aver modificato come descritto il file config.inc.php possiamo procedere a copiare la directory sul nostro server. Al termine dell'upload dei file possiamo accedere al nostro phpMyAdmin puntando al file index.php contenuto all'interno della directory appena creata sul server (ad es: <http://slackmania.altervista.org/phpMyAdmin/index.php>)
Se tutto è andato a buon fine dovremmo visualizzare la finestra principale di phpMyAdmin.



Benvenuto in phpMyAdmin 2.6.2

MySQL 5.0.10-beta-log in esecuzione su localhost come demo@localhost



MySQL	phpMyAdmin
<p> Crea un nuovo database </p> <p>test_... <input type="text"/> Collation <input type="text"/> <input type="button" value="Crea"/></p> <p> Visualizza informazioni di runtime di MySQL</p> <p> Visualizza variabili di sistema di MySQL</p> <p> Visualizza processi in esecuzione</p> <p> Set di Caratteri e Collations</p> <p> Motori di Memorizzazione</p> <p> Database</p> <p> Esporta</p> <p> Cambia password</p> <p> Disconnetti</p>	<p>Language <input type="text" value="Italian (it-utf8)"/></p> <p>Set di caratteri MySQL: UTF-8 Unicode (utf8)</p> <p>collazione della connessione di MySQL: <input type="text" value="utf8_general_ci"/></p> <p>Tema / Stile: <input type="text" value="Original"/></p> <p>Documentazione di phpMyAdmin</p> <p>Mostra le info sul PHP</p> <p>Home page ufficiale di phpMyAdmin</p> <p>[ChangeLog] [CVS] [Lists]</p>

Database:

Prego, selezionare un database



Quest'opera è stata rilasciata sotto la licenza Creative Commons Attribuzione - Condividi allo stesso modo. Per leggere una copia della licenza visita il sito web <http://creativecommons.org/licenses/by-sa/2.0/it/deed.it> o spediisci una lettera a *Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.*