

## Enterprise Linux 實戰講座

### 利用 MySQL 整合 Apache 網頁驗證

5 月號的 Enterprise Linux 實戰講座「RHEL 4 之 WWW 伺服器-Apache 2」筆者曾介紹利用 htpasswd 指令建立使用者帳號及密碼檔來限制瀏覽網頁時必須輸入帳號及密碼。此方法雖然簡易方便，但是如果用戶數眾多，利用 htpasswd 指令來管理帳號及密碼，除了效率會變差外，也很難滿足集中管理的需求。所以利用「資料庫」來存放這些帳號及密碼，整合 Apache 網頁驗證便是個很好的解決方案。本期文章筆者便是要介紹如何「利用 MySQL 整合 Apache 網頁驗證」。

#### 1.1 MySQL 簡介

MySQL 是一種跨平台，多使用者，多執行緒，支援 SQL 語言 (結構化查詢語言)，執行速度快，廣受全球網站大量採用的網路資料庫系統，最重要的是，在大部份情形下，使用 MySQL 是完全免費的！(MySQL 自 3.23.19 之後已改為 GPL 版權宣告囉)。整體而言，MySQL 免費，原始碼開放，入門容易，執行速度快，跨平台 (亦支援 Windows 2000)，支援工具眾多等特性，十分適合中小企業用來開發各種資訊系統！企業不必再花費數十萬元，去購買一些費而不惠的資料庫！

MySQL 沒有所謂連線使用者限制，其它商用資料庫系統，一定數量以上的連線使用者數，可是要付出相當可觀的費用的！像 Yahoo 等大公司，也是用 MySQL 哩！連這麼大的入口網站都安心地挑選 MySQL，一般的中小企業更無需擔心 MySQL 的穩定性。

RHEL 4 內附的 MySQL 為 4.1.7-4 版，與前版本比較具有下列重大的功能改進：

- 內建提供 InnoDB table type，支援 standard binaries, adding transactions, row-level locking, 和 foreign keys。
- 提昇 Query 速度。
- 改善 boolean mode, truncation, and phrase searching. 全文檢索效率。

- 加強 MERGE tables, 並支援 INSERT statements and AUTO\_INCREMENT.
- 在 SELECT 中支援 UNION 語法
- 支援 Multi-table DELETE statements.
- libmysqld, the embedded server library.
- 更多的 GRANT 參數來加強安全性
- 提供更多動態參數來降低停止 MySQL 的必要。
- 改善 replication code and features.
- 改善執行效率並增加可靠性.

## 1.2 安裝 MySQL 及設定 MySQL root 使用者密碼

以 root 的身份登入系統，開啟終端視窗，鍵入「system-config-packages」。利用 GUI 套件管理工具「system-config-packages」，勾選「網頁伺服器」即可，然後按「更新」，便會提示放入適當的光碟片便可順利完成安裝 MySQL 的工作。

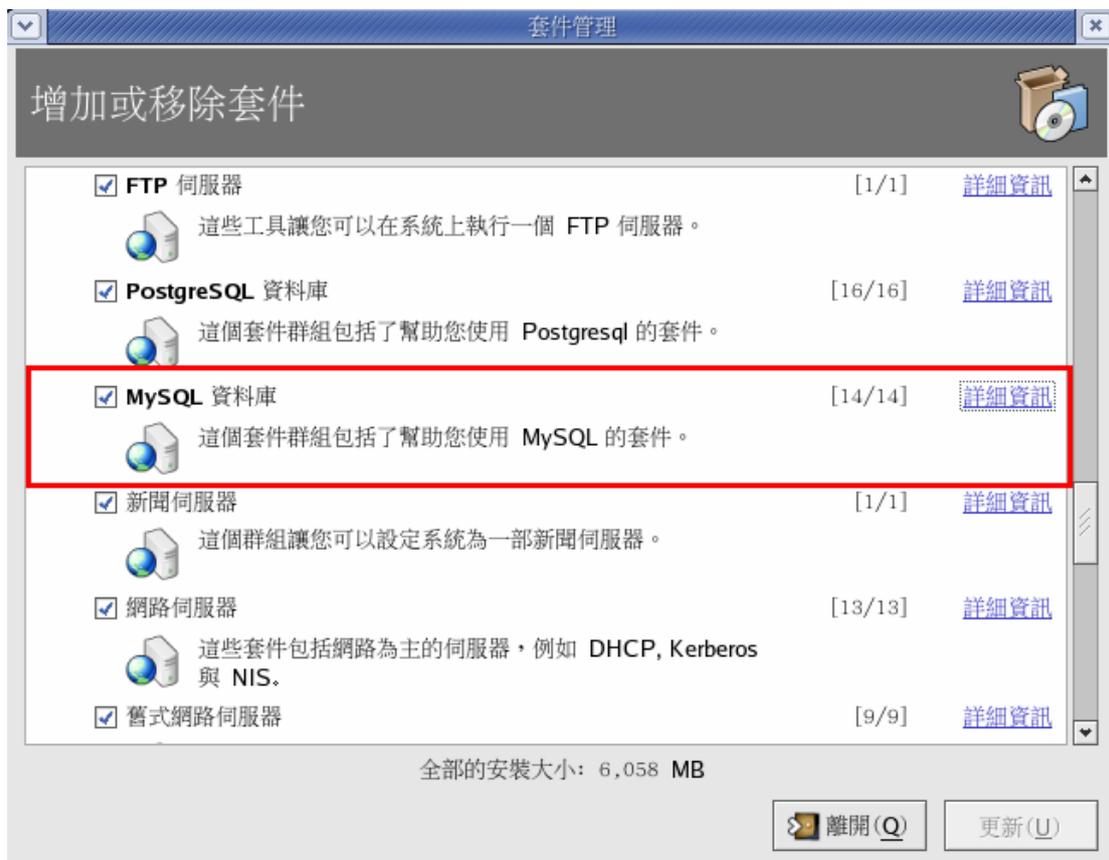


圖 1：system-config-pacakges 畫面

待安裝完成後，可利用「rpm -qa | grpe '^mysql'」檢查是否安裝成功。除了採用「system-config-packages」工具安裝外，亦可利用rpm -ivh 指令進行安裝。然後利用「service mysqld start」啟動MySQL。

```
root@server1 ~]# rpm -qa | grep ^mysql
mysql-bench-4.1.7-4.RHEL4.1
mysqlclient10-3.23.58-4.RHEL4.1
mysql-devel-4.1.7-4.RHEL4.1
mysql-server-4.1.7-4.RHEL4.1
mysqlclient10-devel-3.23.58-4.RHEL4.1
mysql-4.1.7-4.RHEL4.1
```

```
[root@server1 ~]# service mysqld start
```

```
啟動 MySQL:
```

```
[ 確定 ]
```

成功啟動 MySQL 後，因為資料庫最高權限管理者 root（此 root 使用者和作業系統的 root 帳號無關）預設沒有密碼，建議更改 MySQL 管理者密碼，請輸入下列指令進行修改密碼：

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

### 1.3 phpMyAdmin 安裝

phpMyAdmin 是免費的 PHP 原始程式碼的 Web 網站，可用來管理整個 MySQL 伺服器，這是一套 Web 介面的 MySQL 管理程式，支援中文的使用者介面。phpMyAdmin 只需透過瀏覽器就可管理 MySQL 伺服器，換句話說，不論 MySQL 是安裝在 Linux 或是 Windows 作業系統，都可以使用相同步驟在 MySQL 上建立資料庫（Database）、資料表（Table）和編輯紀錄（Record）。

**Step 1:** 載 phpMyAdmin (<http://www.phpmyadmin.net>)，選擇最新穩定的版本。

**Step 2:** phpMyAdmin-2.6.4-pl2.tar.bz2 下載至/tmp，並執行下列指令將其解壓縮至/var/www/html。

```
# tar jxvf phpMyAdmin-2.6.4-pl2.tar.bz2 -C /var/www/html/
```

**Step 3** : 為了操作方便起見，建立 Soft link 捷徑。

```
#ln -s /var/www/html/phpMyAdmin-2.6.4-pl2/ /var/www/html/phpMyAdmin
```

**Step 4** : phpMyadmin 設定檔 config.inc.php

```
# vi /var/www/html/phpMyAdmin/config.inc.php
```

第 74 行 利用何帳號存取 MySQL

第 75 行 如果 root 有設定 password，此處即填入 root 的 password

```
74 $cfg['Servers'][$i]['user']          = 'root';  
75 $cfg['Servers'][$i]['password']     = 'redhat';
```

**Step 5** : <http://主機名稱/phpMyAdmin>

啟動網頁伺服器後，打開瀏覽器後鍵入「<http://主機名稱/phpMyAdmin>」，便可看 phpMyAdmin 管理介面。

```
# service httpd start
```



圖 2：phpMyAdmin 管理介面

## 1.4 實戰演練：MySQL 整合 Apache 網頁驗證

### (mod\_auth\_mysql)

#### 1.4.1 流程分析：

此演練筆者將在建立為「auth」的資料庫，並在「auth」資料庫上新增一個「users」的Table，作為存放使用者帳號與密碼之用，其欄位名稱及資料型態如下：

表 1：users 表格的欄位名稱及資料型態

欄位名稱	資料型態	Null	Key
user_name	varchar(30)	Not Null	Primary (主鍵)
user_passwd	varchar(20)	Not Null	

為了安全起見，筆者額外建立一個資料庫使用者，名為「**authuser**」，其密碼為「**PaSsW0Rd**」，只賦予此資料庫使用者可以讀取「**users**」資料表的權限。

整個實戰演練流程分為下列部份：

- 建立「**auth**」資料庫及「**users**」Table，並新增相關紀錄。
- 建立「**authuser**」資料庫帳號並賦予適當的權限。
- 建立測試網頁/var/www/html/mysql。
- 安裝mod\_auth\_mysql模組，並修改**auth\_mysql.conf**設定檔。

### 1.4.2 建立「**auth**」資料庫及「**users**」Table，並新增相關紀錄。

要建立所需資料，可利用指令方式，亦可利用phpMyAdmin工具來新增所需的資料庫及表格資料。筆者首先示利用指令的方式，然後再介紹利用phpMyAdmin工具來完成需求。

#### 【法一：利用MySQL指令】

利用指令建立建立「**auth**」資料庫及「**users**」資料表的及新增帳號/密碼的過程如下：

```
# mysql -u root -p
Enter password:redhat ←輸入MySQL的root使用者的密碼
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3 to server version: 4.1.7

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

mysql> CREATE DATABASE auth; ←建立auth資料庫
Query OK, 1 row affected (0.03 sec)

mysql> USE auth; ←連結使用auth資料庫
Database changed

mysql> CREATE TABLE users (←連結使用users表格
```

```
-> user_name CHAR(30) NOT NULL,  
-> user_passwd CHAR(20) NOT NULL,  
-> PRIMARY KEY (user_name)  
-> );
```

Query OK, 0 rows affected (0.00 sec)

新增alex帳號，密碼為1225

```
mysql> INSERT INTO users VALUES ('alex', ENCRYPT('1225'));
```

Query OK, 1 row affected (0.00 sec)

新增neon帳號，密碼為1202

```
mysql> INSERT INTO users VALUES ('neon', ENCRYPT('1202'));
```

Query OK, 1 row affected (0.00 sec)

新增evil帳號，密碼為1234

```
mysql> INSERT INTO users VALUES ('evil', ENCRYPT('1234'));
```

Query OK, 1 row affected (0.00 sec)

## 【法二：利用phpMyAdmin工具】

### Step 1：建立「auth 資料庫」和「users 資料表」

輸入資料庫名稱 auth 後並按下「建立」（圖 3），並建立 users 資料表，因為此資料表有 2 個欄位，所以並指定欄位數量為 2（圖 4）。



圖 3：建立「auth」資料庫

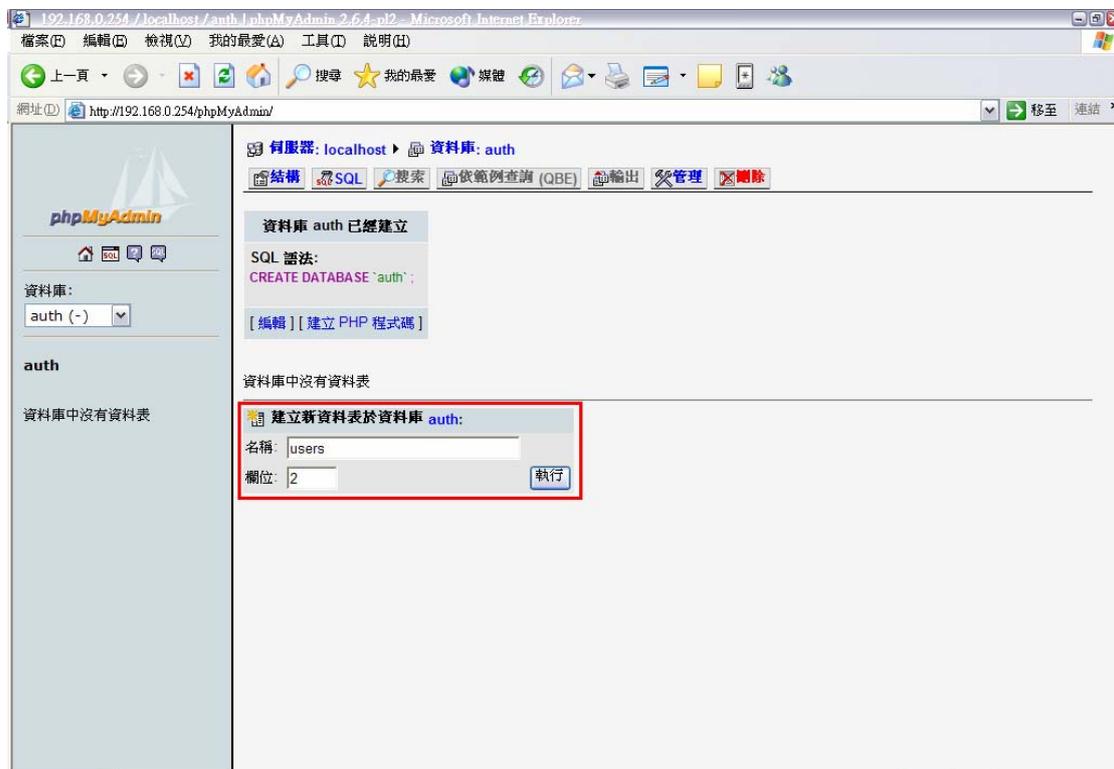


圖 4：填入資料表名稱並指定欄位數量

## Step 2：設定欄位屬性

根據表 1 建立相關的欄位及設定其資料型態與限制，首先建立「user\_name」欄位及指定其資料型態；然後再建立「user\_passwd」欄位指定其資料型態（圖 5）。記得指定「user\_name」欄位為「Primary Key（主鍵）」（圖 6），最後按下「儲存」執行相關 SQL 命令。

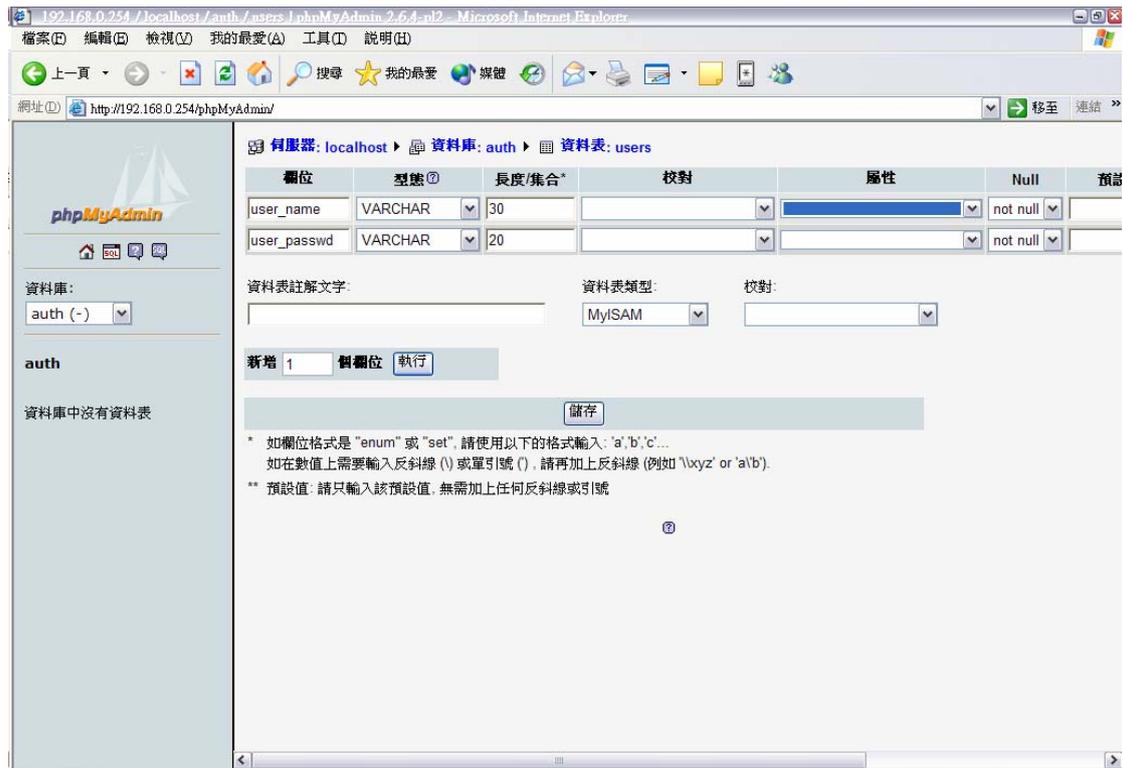


圖 5：設定欄位屬性



圖 6：設定主鍵

## Step 3：新增網頁使用者相關紀錄

新增網頁使用者紀錄，例如：user\_name=「alex」，user\_passwd=「1225」（圖 7）。設定密碼時，請特別留意必須指定使用「ENCRYPT」函數對密碼進行加密（圖 8）。

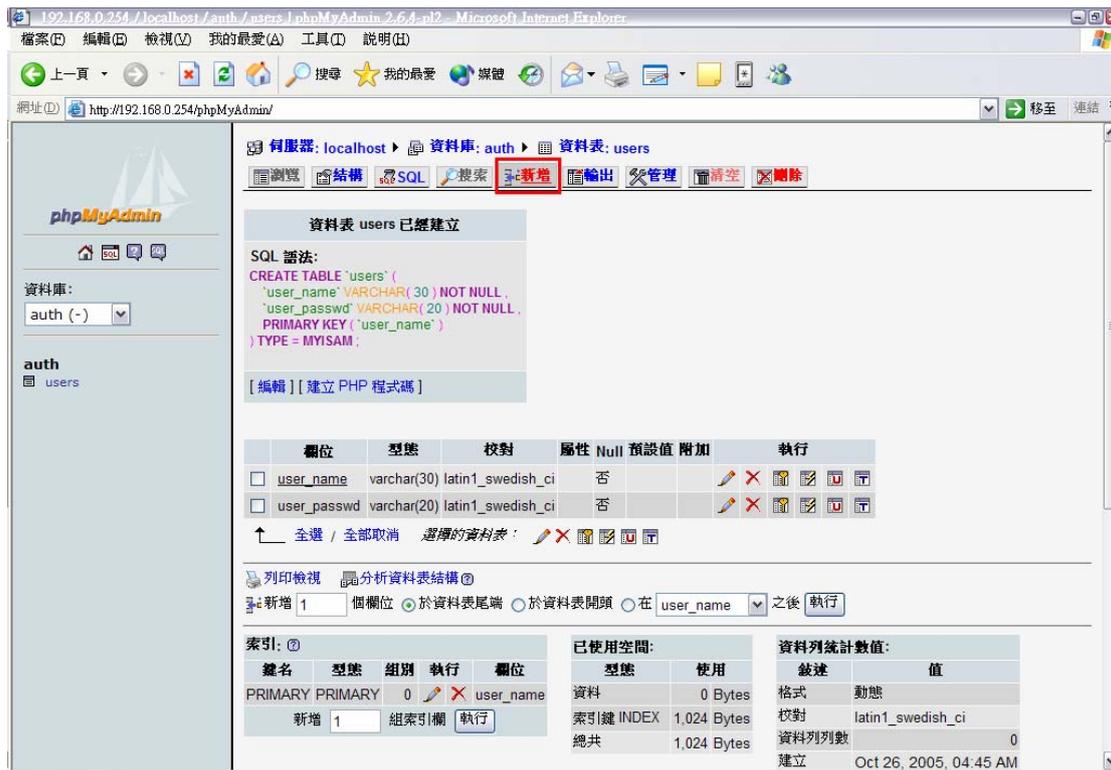


圖 7：users 資料表結構及相關 SQL 語法

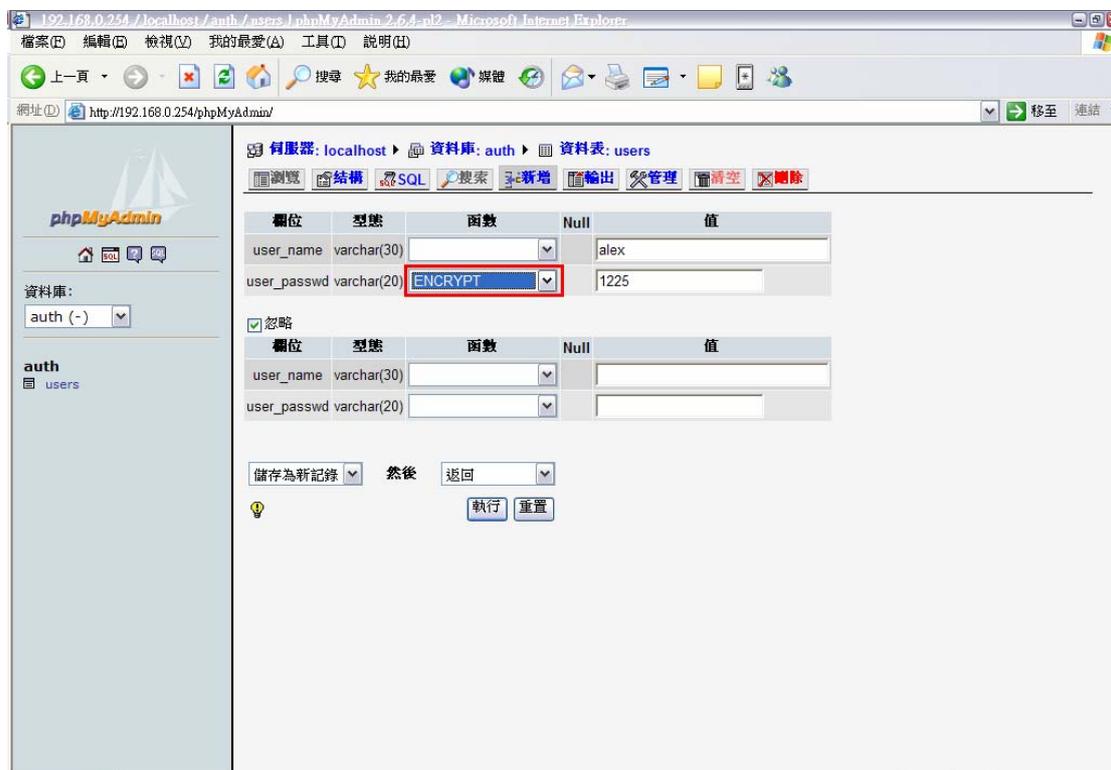


圖 8：新增網頁使用者紀錄



### 1.4.3 建立「authuser」資料庫使用者並賦予適當的權限

此部份的工作亦MySQL指令或phpMyAdmin工具來完成需求。

#### 【法一：利用MySQL指令】

利用MySQL指令建立「authuser」資料庫使用者並賦予此使用者可以select（相當於讀取）auth資料庫中users表格內容的權限的操作過程如下：

```
mysql> GRANT SELECT
-> ON auth.users
-> TO authuser@localhost
-> IDENTIFIED BY 'PaSsW0Rd';
Query OK, 0 rows affected (0.06 sec)

mysql> exit ←離開MySQL命令列
```

#### 【法二：利用phpMyAdmin】

利用phpMyAdmin工具建立「authuser」資料庫使用者並賦予此使用者可以select（相當於讀取）auth資料庫中users表格內容的權限的操作過程如下：

#### Step1：建立「authuser」使用者

點選最上方的「伺服器:localhost」回到 phpMyAdmin 管理介面首頁，點選「權限」；進入資料庫使用者管理畫面後，點選「新增使用者」，然後鍵入此使用者相關權限資料，最後按下「執行」。

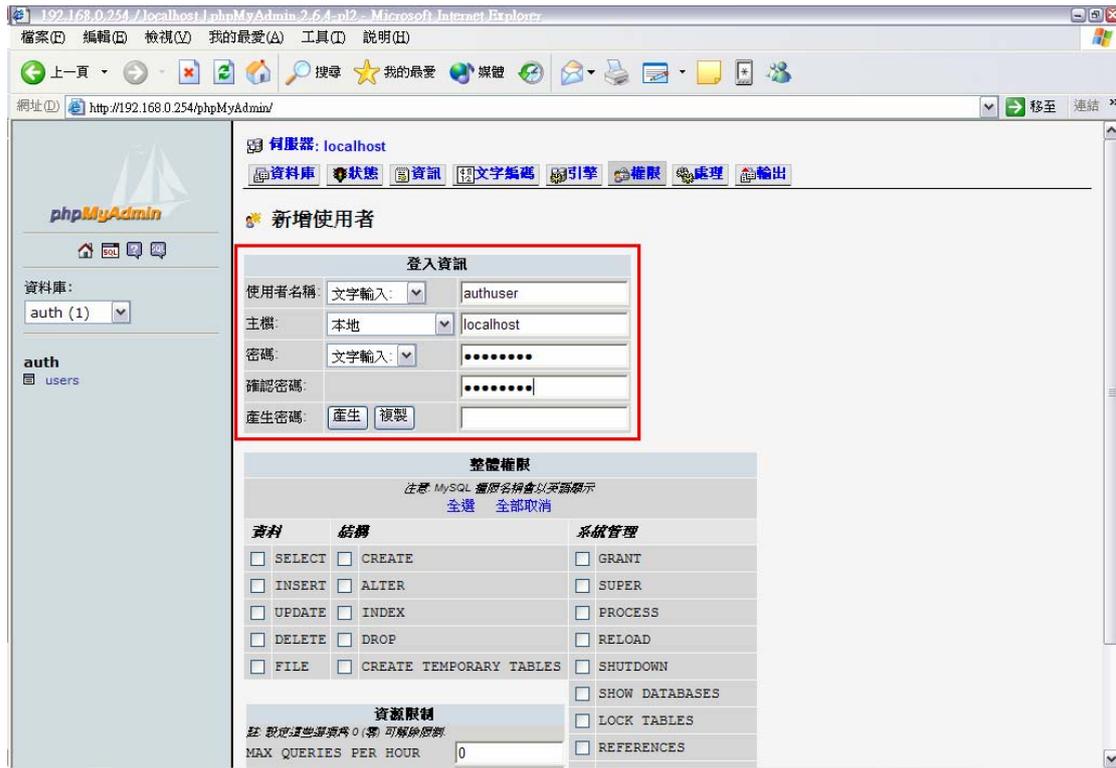


圖 9：新增資料庫使用者

**Step2：指定資料庫權限**

在圖 9 畫面中「指定資料庫權限」項目中，利用下拉式選單選取 auth 資料庫；或是直接在文字框中輸入 auth，然後按下「執行」（圖 10）。



圖 10：指定資料庫權限

**Step3：指定資料表權限**

「指定資料表權限」的功用便是設定 authuser 使用者可對資料表做那些動作 (Select、Insert、Update...)，便是在圖 10 畫面中「指定資料表權限」項目中，利用下拉式選單選取 users 資料庫；或是直接在文字框中輸入 users，然後按下「執行」(圖 11)。

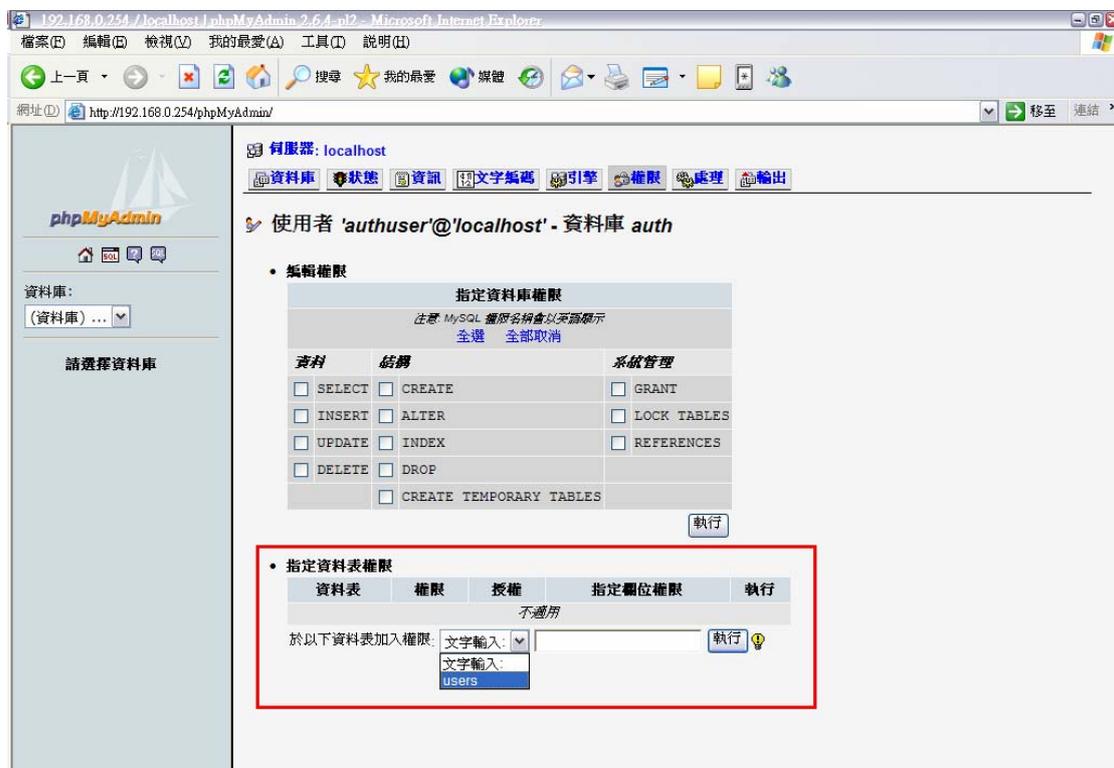


圖 11：指定資料表權限

**Step4：賦予 authuser 使用者權限**

最後賦予 authuser 使用者可以對「users」資料表中的 user\_name 及 user\_passwd 可以做「select」的權限。點選 select 框架中的 user\_name 及 user\_passwd 欄位，然後按下「執行」，賦予 authuser 使用者可以「select」「users」資料表中的 user\_name 及 user\_passwd 欄位資料。

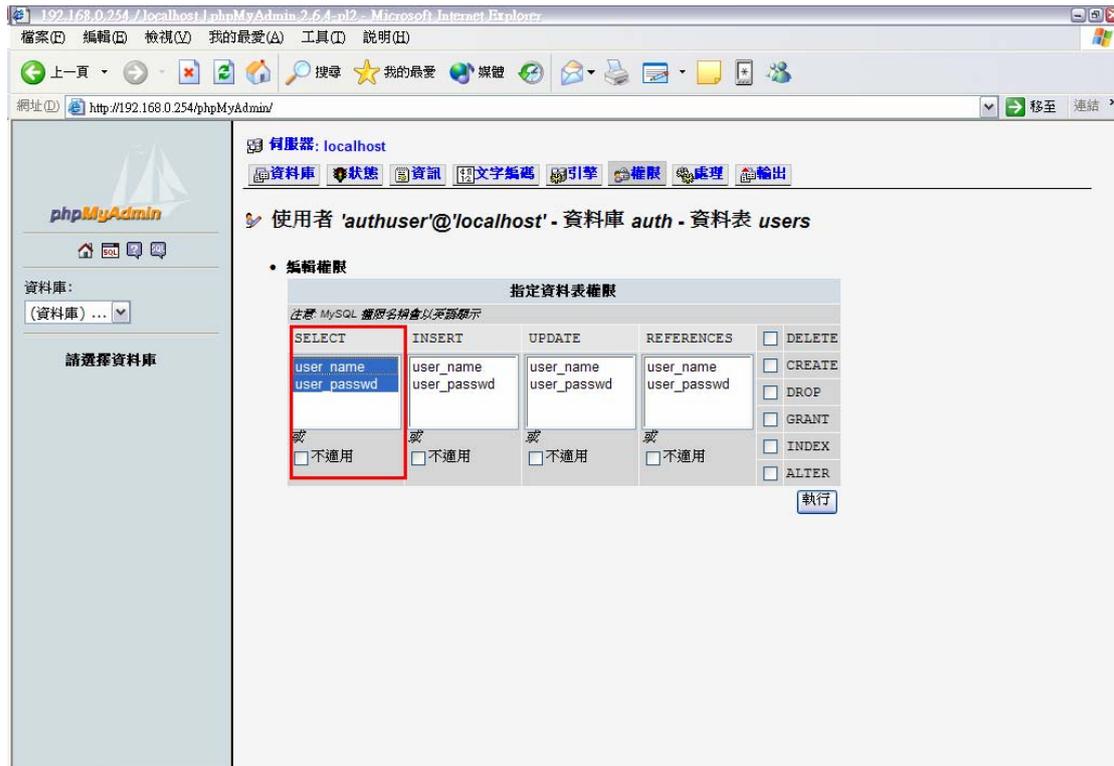


圖12：編輯資料表權限

#### 1.4.4 建立測試網頁/var/www/html/mysql

```
#mkdir /var/www/html/mysql
#echo "<h1>Test Page</h1>" > /var/www/html/mysql/index.html
```

此時若讀者打開瀏覽器，鍵入「<http://主機名稱/mysql>」應不會出現認證視窗，便可直接看到測試網頁。

#### 1.4.5 安裝 mod\_auth\_mysql 模組，並修改 auth\_mysql.conf 設定檔。

##### Step 1：安裝 mod\_auth\_mysql 模組

要讓Apache伺服器可以存取MySQL資料庫伺服器上的資料，必須利用 mod\_auth\_mysql模組作為Apache和MySQL之間認證的介面，所以必須安裝 mod\_auth\_mysql模組。

```
# rpm -ivh mod_auth_mysql-2.6.1-2.1.i386.rpm
warning: mod_auth_mysql-2.6.1-2.1.i386.rpm: V3 DSA signature: NOKEY, key ID
db42a60e
Preparing...      ##### [100%]
 1:mod_auth_mysql  #####
[100%]
```

### Step 2：修改/etc/httpd/conf.d/auth\_mysql.conf

安裝 mod\_auth\_mysql 模組後，需修改此模組的設定檔，指定認證方式，在 auth\_mysql.conf 檔案最後加上以下文字。

```
<Directory /var/www>
  AuthName "MySQL authenticated zone"
  AuthType Basic
  AuthMySQLUser authuser
  AuthMySQLPassword PaSsW0Rd
  AuthMySQLDB auth
  AuthMySQLUserTable users
  AuthMySQLNameField user_name
  AuthMySQLPasswordField user_passwd
  require valid-user
</Directory>
```

### Step 3：重新啟動 httpd

```
#service httpd restart
停止 httpd:      [ 確定 ]
啟動 httpd:      [ 確定 ]
```

**Step 4：測試「<http://主機名稱/mysql/>」**

最後打開瀏覽器，鍵入「<http://主機名稱/mysql/>」應會出現認證視窗，輸入 users 資料表中的帳號/密碼紀錄，才可直接看到測試網頁。



圖13：認證視窗畫面

**後記：**

本期文章介紹利用 MySQL 來整合網頁認證，其實除了可以利用 MySQL 外，亦可利用 LDAP 伺服器來集中管理網頁使用者的資料，下期文章筆者便會介紹如何利用 LDAP 伺服器來整合網頁認證。

]

**作者簡介**

林彥明 (Alex Lin)：現任職於 IBM Taiwan 技術支援中心，負責 Linux、AIX、WebSphere 相關技術支援工作。具有 RHCX (RedHat 認證主考官)、RHCE、NCLP (Novell Linux 認證專家)、LPIC、IBM AIX Expert、IBM MQ、SCJP、SCWCD 等國際認證，參與建置臺灣第一套商業用 IBM 1350 Linux 叢集系統及 RHEL 4、SLES 9 on zSeries 等 Linux 專案。