Enterprise Linux 實戰講座

利用 LDAP 整合 Apache 網頁驗證

上期的 Enterprise Linux 實戰講座「**利用 MySQL 整合 Apache 網頁驗證**」，筆者介紹利用 MySQL 資料庫伺服器來存放帳號及密碼，藉以整合 Apache 網頁驗證，以達集中管理之目的。此法雖好，但唯一美中不足的是很多的應用程式並無法與 MySQL 整合，所以此篇文章，筆者介紹利用 LDAP 整合 Apache 網頁驗證，因為很多的應用程式，都可以與 LDAP 整合，所以用 LDAP 來整合 Apache 網頁驗證才是王道。

1 LDAP 簡介

LDAP（Lightweight Directory Access Protocol；輕量型目錄存取協定）是一個利用目錄將資訊以層級架構的方法組織起來。可以把它想像成一個簡單的資料庫系統。在完成建立後，可透過網路來存取它們。LDAP 是根據 X.500 目錄分享的標準，但並不如 X.500 複雜。實際上，LDAP 有時候被稱為 X.500 的輕量簡化版。LDAP 目錄伺服器可以儲存很多不同的資訊，並允許使用者從支援 LDAP 協定的應用程式來存取他們的帳戶資料。

LDAP 是一個客戶端／伺服器系統。伺服器可使用不同的資料庫來儲存目錄，每一個皆最佳化以提供快速及大量的存取要求。當 LDAP 客戶端應用程式連線至 LDAP 伺服器時，它可以查詢目錄資訊或上載資料。當進行查詢時，伺服器可回答查詢，或者如果該伺服器無法提供答案的話，則會將查詢轉至能夠答覆的較高一層的 LDAP 伺服器。如果客戶端企圖將資料上載至 LDAP 目錄，伺服器會先驗證使用者擁有更改的權限，然後才允許新增或更新資料。

RHEL 4 內附的 LDAP 伺服器為 OpenLDAP 2.2.13-2 版，OpenLDAP 2.x 包括數個重要功能：

- 支援 LDAPv3 - OpenLDAP 2.0 除了其他改善外還支援 SASL（Simple Authentication and Security Layer）、TLS（Transport Layer Security）以及 SSL（Secure Sockets Layer）。LDAPv2 之後通訊協定很多的改變都是為了加強 LDAP 的安全性。

- 支援 IPv6 - OpenLDAP 支援新一代的網際網路通訊協定第 6 版。

- LDAP Over IPC - OpenLDAP 能夠使用 IPC 在系統內進行通訊。這可藉由避免使用網路通訊以增加安全性。

- 新版 C 應用程式界面 - 改善程式設計人員連線及使用程式的方法。

- 支援 LDIFv1 - 完全合乎 LDIF（LDAP Data Interchange Format）第一版的標準。

- 增強獨立 LDAP 伺服器 - 包括新版的存取控制系統及較佳的工具。

- 內建提供 InnoDB 表格型別，支援標準二進位格式資料及資料庫交易異動（Transactions）機制及 row-level locking 和 foreign keys。

2 實戰原理及流程

LDAP 的原理及相關名詞、設計、協定...等相關文件常動輒百頁甚至上千頁，常令人望之卻步。其實筆者覺得一個技術的提出，目的也不外乎是要解決許多人會遇到的共同問題。如果太難不免就失去實用的價值。若讀者從未接觸過 LDAP，也不用擔心此篇文章看不懂，就把 LDAP 伺服器想成一本公司通訊錄，這個通訊錄記錄各個部門員工的相關資料，例如：姓名、性別、密碼、電話、住址、e-mail…等資訊。而利用 LDAP 完成 Apache 網頁認證工作的原理其實很簡單，當 Apache 收到使用者在對話框輸入的帳號／密碼後，將其送給 LDAP 伺服器，LDAP 將帳號／密碼和其資料庫中的資料比對，查看輸入的帳號／密碼是否有誤。

所以此實戰演練必須先架設 LDAP 伺服器，由於要在 LDAP 伺服器新增人員紀錄，通常要編寫複雜的 LDIF 檔，通常不是初接觸 LDAP 的人可輕易完成的。所以筆者的想法為：先在利用 Linux 上用傳統方法建立帳號並為其設定密碼；然後將此台伺服器轉為 LDAP 伺服器，並將原 Linux 帳號轉為 LDAP 伺服器內的人員紀錄。由於 RHEL 4 預設並未提供一個圖形化的 LDAP 管理介面，筆者將會安裝 phpldapadmin，讀者可利用 phpldapadmin 這個 Web 介面來管理 LDAP 的資料，最後修改 Apache 設定檔，讓網頁認證機制與 LDAP 整合。

綜合上述，整個實作流程如下：

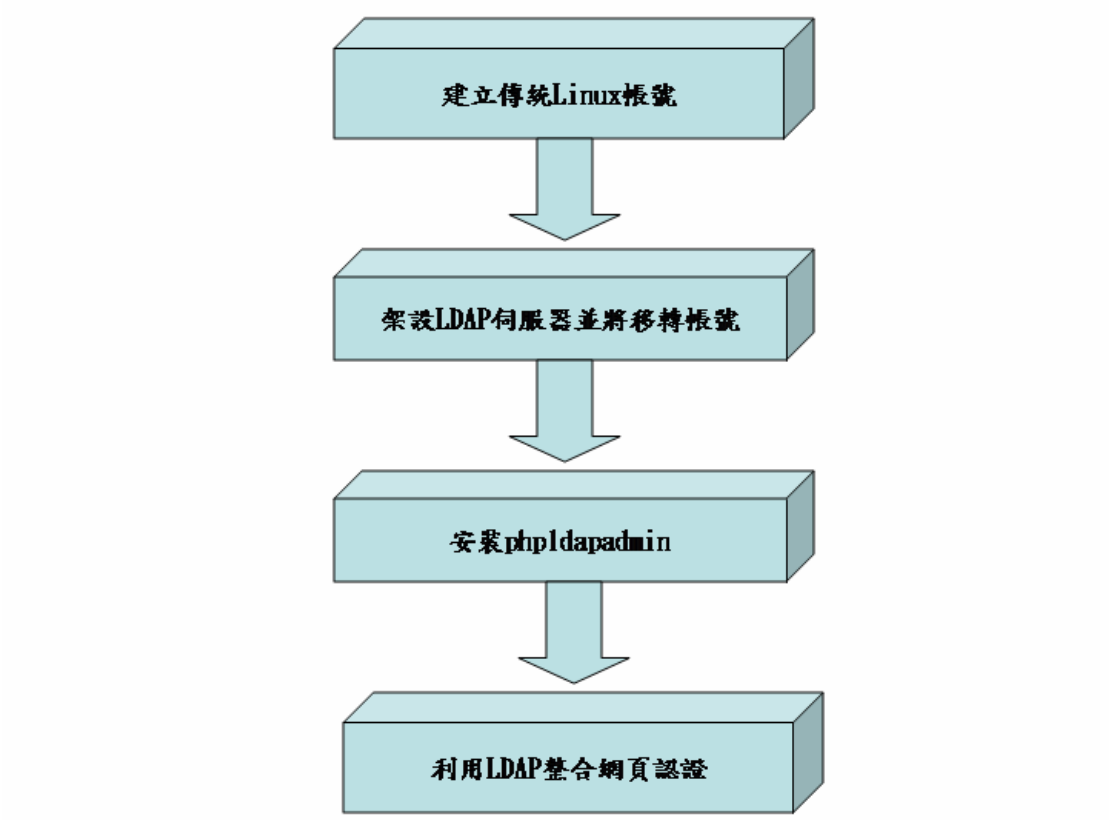
建立傳統Linux帳號
架設LDAP伺服器並將移轉帳號
安裝phpldapadmin
利用LDAP整合網頁認證

圖 1：實作流程

3 實戰演練：LDAP 整合 Apache 網頁驗證

3.1 建立傳統 Linux 帳號

筆者利用 Shell Script 來建立多個使用者帳號，步驟如下：

步驟 1：建立使用者清單 users.lst

筆者所撰寫的 shell script 需要新增使用者的清單，其中只需包含兩個欄位，第一個欄位為使用者名稱；第二個欄位為預設密碼，中間必須用空格隔開。

```
# cat users.list
www1 1234
www2 1234
www3 1234
www4 1234
www5 1234
www6 1234
www7 1234
www8 1234
www9 1234
www10 1234
```

步驟 2：撰寫大量建立帳號的 shell script：batch-add-users.sh

```
# vi batch-add-users.sh
#!/bin/bash
#awk 兩旁的是反單引號，就是鍵盤數字鍵 1 左邊的符號鍵
for i in `awk '{print $1}' users.list `
do
   useradd $i
   grep "\<$i\>" users.list | awk '{print $2}' | passwd --stdin $i
   ←設定使用者的密碼
```

done

步驟 3：執行 batch-add-users.sh

修改 batch-add-users.sh 權限，使其可以被執行。執行完 batch-add-users.sh 後，會發現/etc/passwd、/etc/group 和/etc/shadow 多了這些使用者的相關記錄。

步驟 4：測試，利用新建的帳號登入

可在 Virtual Console（Ctrl+Alt+F1~F6）或圖形登入畫面（Ctrl+Alt+F7）利用這些帳號登入，測試帳號／密碼可否順利運作。

3.2 架設 LDAP 伺服器並移轉帳號

步驟 1：安裝 LDAP 伺服器

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」→「網路伺服器」（圖 2），點選「詳細資訊」，然後勾選「openldap-server」（圖 3），便會提示放入適當的光碟片，順利完成安裝 LDAP 伺服器的工作。


套件管理
增加或移除套件
新聞伺服器 [0/1]
這個群組讓您可以設定系統為一部新聞伺服器。
網路伺服器 [6/13] 詳細資訊
這些套件包括網路為主的伺服器，例如 DHCP, Kerberos 與 NIS。
舊式網路伺服器 [2/9] 詳細資訊
這些套件包括舊式網路協定的伺服器，例如 rsh 與 telnet。
程式開發
開發工具 [0/73]
這些工具包括主要的開發工具，例如 automake, gcc, perl, python 以及 debuggers。
X 軟體開發 [0/18]
這些套件讓您可以為 X 視窗系統開發應用程式。
全部的安裝大小: 1,704 MB
離開(Q)
更新(U)

圖 2：增加或移除套件

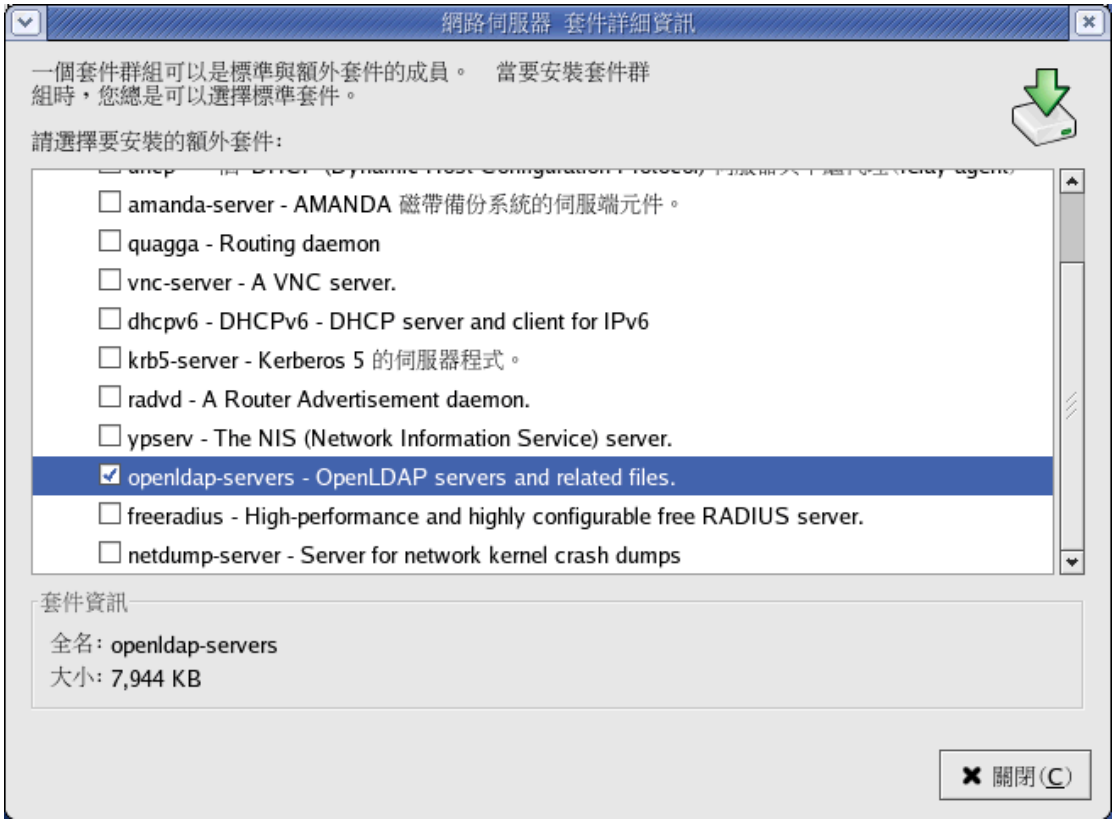

圖 3：選取 openldap-servers 套件

待安裝完成後，可利用「rpm -qa | grpe '^openldap'」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。

```
# rpm -qa | grep '^openldap'
openldap-devel-2.2.13-2
openldap-servers-2.2.13-2
openldap-2.2.13-2
openldap-clients-2.2.13-2
openldap-servers-sql-2.2.13-2
```

步驟 2：修改/etc/openldap/slapd.conf

#vi /etc/openldap/slapd.conf

```
68 database      bdb
69 suffix        "dc=example,dc=com"
```

```
70 rootdn          "cn=Manager,dc=example,dc=com"
74 rootpw          redhat
```

步驟 3：將原有 Linux 帳號轉為 LDIF 檔

原有 Linux 伺服器上有 www1~www10 這些使用者帳號，密碼均為 1234；筆者欲將這些帳號／密碼轉換至 LDAP 伺服器上，也就是在 LDAP 伺服器上新增 10 個員工紀錄。先撇開複雜的 LDAP 理論，前面提到 LDAP 伺服器就想成公司通訊錄，那這本通訊錄就得註明是那家公司的通訊錄；對應原有 Linux 伺服器的觀念，讀者可以這樣想，原本每台 Linux 主機都會有主機名稱，例如 server1.example.com，其中 example.com 為網域名稱，網域名稱不就是代表那家公司。

原有 Linux 帳號管理的觀念為使用者（/etc/passwd）及群組（/etc/group），所以筆者為了對應原有的 Linux 帳號管理結構，便將這個公司（example.com）的通訊錄分為兩大組織：1.people，2.group，整個公司通訊錄結構應如圖 4。

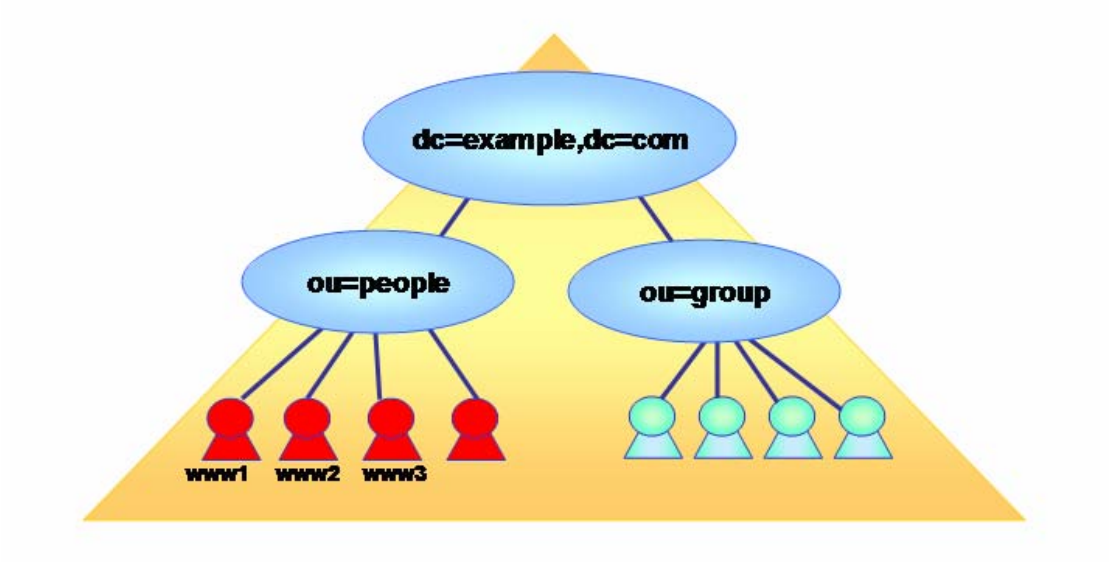

圖 4：LDAP 通訊錄架構

接著利用 Linux 的工具將原有 Linux 帳號轉換至 LDAP 通信錄，這些轉換工具置於/usr/share/openldap/migration 目錄，下面便是轉換的步驟。

```
# cd /usr/share/openldap/migration
```

 # vi migrate_common.ph

```
71 $DEFAULT_MAIL_DOMAIN = "example.com";
72
73 # Default base
74 $DEFAULT_BASE = "dc=example,dc=com";
```

```
# ./migrate_passwd.pl /etc/passwd > /worktmp/user.ldif
# ./migrate_group.pl /etc/group > /worktmp/group.ldif
```

步驟 4：建立 example.ldif,ou_people.ldif, ou_group.ldif

example.ldif 這個檔案是為了建立 example.com 這個最上層的物件；ou_people.ldif 是為了建立 people 這個組織物件；ou_group.ldif 是為了建立 group 這個組織物件。

#cat example.ldif

```
dn: dc=example,dc=com
dc: example
objectClass: dcObject
objectClass: organizationalUnit
ou: example.com
```

#cat ou_people.ldif

```
dn: ou=people, dc=example, dc=com
objectclass: organizationalunit
ou: people
```

#cat ou_group.ldif

```
dn: ou=group, dc=example, dc=com
objectclass: organizationalunit
ou: group
```

步驟 5：轉換原有 Linux 帳號至 LDAP Server 上

```
#slapadd -vl example.ldif
added: "dc=example,dc=com" (00000001)
```

```
#slapadd -vl ou_people.ldif
added: "ou=people,dc=example,dc=com" (00000002)
```

```
#slapadd -vl ou_group.ldif
added: "ou=group,dc=example,dc=com" (00000043)
```

```
#slapadd -vl user.ldif
```

```
#slapadd -vl group.ldif
```

步驟 6：啟動 LDAP Server 並檢查其資料庫

記得把/var/lib/ldap/目錄內的檔案變更擁有者及群組為 ldap。

```
#chown ldap.ldap /var/lib/ldap/*
#service ldap start
```

利用 ldapsearch 指令可搜尋 LDAP 伺服器的資料，若是可看到以下的資料，代表整個設定正確無誤。

```
#ldapsearch -x -b "dc=example,dc=com"
```

```
# extended LDIF
#
# LDAPv3
# base <dc=example,dc=com> with scope sub
```


```
# filter: (objectclass=*)
# requesting: ALL
#

# example.com
dn: dc=example,dc=com
dc: example
objectClass: dcObject
objectClass: organizationalUnit
ou: example.com

..
# www9, Group, example.com
dn: cn=www9,ou=Group,dc=example,dc=com
objectClass: posixGroup
objectClass: top
cn: www9
userPassword:: e2NyeXB0fXg=
gidNumber: 508

# www10, Group, example.com
dn: cn=www10,ou=Group,dc=example,dc=com
objectClass: posixGroup
objectClass: top
cn: www10
userPassword:: e2NyeXB0fXg=
gidNumber: 509

# search result
search: 2
```


```
result: 0 Success

# numResponses: 141
# numEntries: 140
```

3.3 安裝 phpldapadmin

phpldapAdmin 是免費的工具，可以管理 LDAP 伺服器，使用 phpldapAdmin 只需透過瀏覽器就可管理 LDAP 伺服器。筆者所用的版本的為 0.9.4b 版本，讀者可至 http://phpldapadmin.sourceforge.net/download.php 下載。

步驟 1：下載 phpldapadmin-0.9.4b.tar.gz

將 phpldapadmin-0.9.4b.tar.gz 下載至/tmp，並執行下列指令將其解壓縮至 /var/www/html。

```
# tar zxvf phpldapadmin-0.9.4b.tar.gz -C /var/www/html/
```

步驟 2：為了操作方便起見，建立 Soft link。

```
#cd /var/www/html
# ln -s phpldapadmin-0.9.4b phpldapadmin
```

步驟 3：修改 phpMyadmin 設定檔。

```
#cd /var/www/html/phpldapadmin
#cp config.php.example config.php
```

 # vi config.php

```
20 $servers[$i]['host'] = 'ldap.localhost';
27 $servers[$i]['base'] = 'dc=example,dc=com'; ←不用改
51 $servers[$i]['login_pass'] = 'secretredhat';
```

步驟 4：http://主機/phpldapadmin。

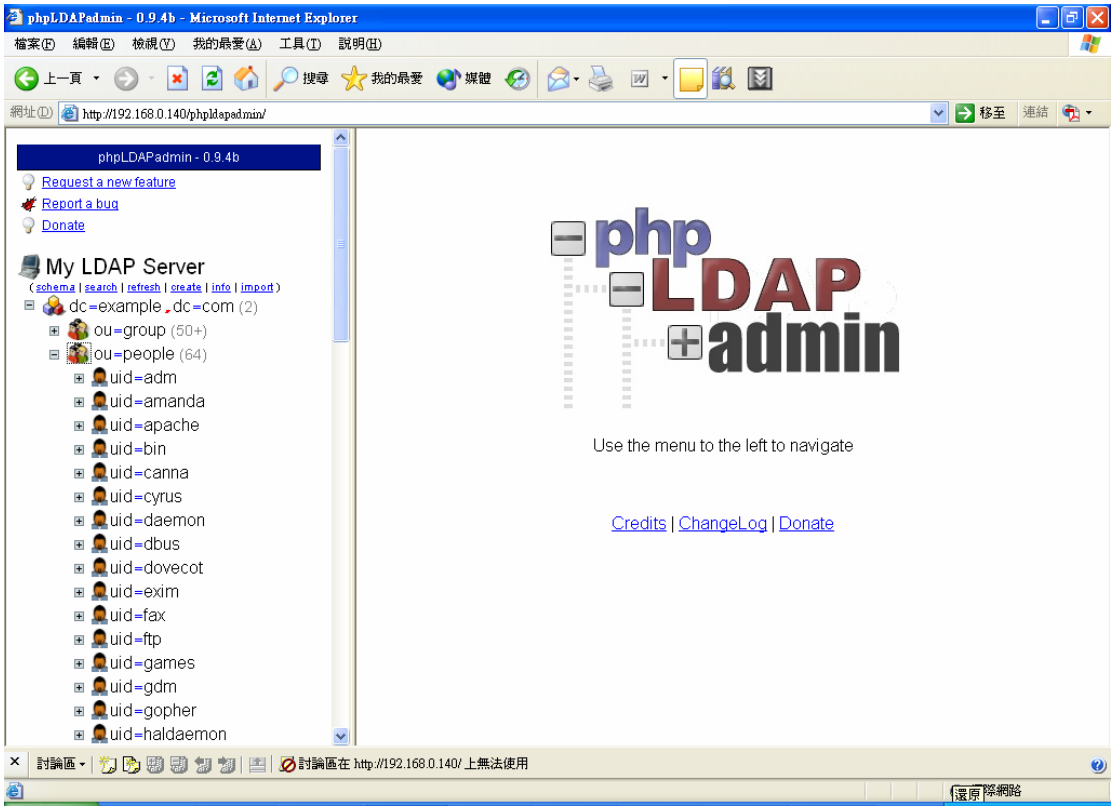

圖 5：phpldapadmin 畫面

3.4 利用 LDAP 整合網頁認證

步驟 1：建立測試網頁。

```
#mkdir /var/www/html/ldap
#echo “LDAP Auth Test Page” > /var/www/html/ldap/index.html
```

步驟 2：安裝 mod_authz_ldap 模組

要讓 Apache 伺服器可以存取 LDAP 伺服器上的資料，必須利用 mod_authz_ldap 模組作為 Apache 和 LDAP 伺服器之間認證的介面，所以必須安裝 mod_authz_ldap 模組。

```
# rpm -ivh mod_authz_ldap＊.rpm
```

步驟 3：修改/etc/httpd/conf.d/auth_mysql.conf。

若之前曾安裝 mod_auth_mysql 模組，則必須將 mod_auth_mysql 模組功能關閉。

```
6 #LoadModule mysql_auth_module modules/mod_auth_mysql.so
```

全部加以註解

步驟 4：修改/etc/httpd/conf.d/authz_ldap.conf。

```
<IfModule mod_authz_ldap.c>

#    <Location /private>
#        AuthzLDAPEngine on
#
#        AuthzLDAPServer localhost
#        AuthzLDAPUserBase ou=People,dc=example,dc=com
#        AuthzLDAPUserKey uid
#        AuthzLDAPUserScope base
#
#        AuthType basic
#        AuthName "ldap@example.com"
#        require valid-user
#
#    </Location>

</IfModule>
```

修改為以下文字

```
LoadModule authz_ldap_module modules/mod_authz_ldap.so

<IfModule mod_authz_ldap.c>
```


```
    <Directory /var/www/html/ldap>
        AuthzLDAPMethod ldap ←RHEL 4 bug
        見 https://bugzilla.redhat.com/bugzilla/show_bug.cgi?id=164620
        AuthzLDAPServer localhost
        AuthzLDAPUserBase ou=People,dc=example,dc=com
        AuthzLDAPUserKey uid
        AuthzLDAPUserScope base 或用 subtree

        AuthType basic
        AuthName "ldap@example.com"
        require valid-user

    </Directory>

</IfModule>
```

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

步驟 5：用帳號 www1~www10 測試「http://localhost/ldap/」。

後記：

本期文章介紹利用 LDAP 來整合網頁認證，除了 Apache 外，現今很多的應用程式均可以與 LDAP 整合，運用 LDAP 來扮演帳號管理的角色是現今企業運用的主流。

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。