

基礎 WWW 伺服器架設 on SLES 9

章節內容：

簡介

Apache2.x 核心與模組新功能

Apache 相關套件及設定檔

安裝 Apache

實例演練：公司網頁 (<http://主機名稱/>)

實例演練：個人網頁 (<http://主機名稱/~geeko>)

實例演練：Name Base 虛擬站台 (Virtual Host)

實例演練：目錄保護機制 (.htaccess)

重點整理

Apache HTTP Server 是一個免費且 Open Source 的 HTTP 伺服器，主要目的在於提供一個安全並且有效率的 HTTP Server 環境。Apache HTTP Server 目前有 Unix-Based 和 Windows 的版本。根據統計，目前 Apache HTTP Server 已成為佔有率最高的 HTTP Server 軟體，而且它的功能與執行效率並不會比一般商業的 Web Server 軟體來的遜色，甚至領先許多。Apache HTTP Server 目前的版本分為 1.3.x 版與 2.0.x 版，兩個版本基本上功能都差不多，不一樣的地方在於 2.0.x 的執行效率會比 1.3.x 來得好，因為 2.0.x 結合了執行緒〈thread〉與程序〈process〉，讓熟悉系統的管理者能夠調校各種不同的組合，以達到最佳效能。

1 簡介

在網際網路初生時期，NCSA 研發了一個網站伺服器。大約在 1995 年左右，NCSA 的網站伺服器已成為世界排名第一的佼佼者。然而，同時期 NCSA 網站伺服器的主要研發群離開了 NCSA，結果這套網站伺服器的發展就延宕下來了。正因為如此，過去採用 NCSA 網站伺服器人開始彼此交換伺服器的修正程式，沒多久之後，眾人便體會出需要一個專門的組織來管理這些補漏程式(patch)才行；該團隊採用原先 NCSA 伺服器的程式碼，並將修補後的網站伺服器正式命名為 Apache。筆者一定會好奇為什麼會命名為 Apache，Apache 不是美國戰鬥直昇機嗎？之所以會命名 Apache 原因是因為這個網站伺服器是上了很多的 patch 的「A patchy server」。所以便取其諧音「Apache」。

Apache 的核心程式是來自於 NCSA 最初的伺服器，並搭配眾家自行開發出來的修正程式綜合而成。當今的 Apache 現已是網站伺服器社區(Web server community) 的主要討論話題了。短短三年，Apache 就獲得市場認同。

Apache 0.6.2 版於 1995 年四月釋出，而 1.0 版在 1995 年 12 月 1 日發表。Apache 團隊從成軍後，便不斷地擴張組織成員，進一步整合成公益團體；整個團隊都靠網際網路運作。Apache 雖然由 Apache 團隊發表，然而，Apache 團隊並沒有對 Apache 的發展定下任何的限制，任何人只要通曉事情該怎麼作，就可以參與 Apache 的開發或者內部元件的整修，Apache 團隊的成員都樂見其成。

不過，任何的更正或修補最後都得由 Apache 團隊審核通過才能加入 Apache 網站伺服器的標準版本，簡言之，Apache 團隊仍保留了 Apache 未來發展之路的掌控權。儘管如此，開放的理念使得世界各地的開發人員得以透過網際網路齊聚一堂，討論新的功能、修正錯誤的邏輯以及移植到別種作業平台等

等。任何新生產出來的程式碼都得送交 Apache 團隊，經由成員檢視程式碼的細節、完成測試工作並執行嚴格的品管控制。如果團隊中的成員滿意新的程式碼，新的程式碼才能正大光明的登堂入室，融入 Apache 的標準範本。

Apache Server 是由 Apache Group 在開發與維護，1999.06.30 成立了 Apache Software Foundation (Apache 軟體基金會，簡稱 ASF)。從原本彙集各家的 Apache Group，到現在有幾家公司仍然持續的支持後續的發展計畫，使得 Apache Server 可以確保持續的發展。

2 Apache2.x 核心與模組新功能

自從 2000 年 3 月在 Apache Conference 2000 論壇發表第一個 Apache 2.0 的測試版後 (Alpha Version)，ASF 歷經了二年多的研究並發表數種測試版，終於在 2002 年 4 月 6 日正式推出 Apache 2.0。它較前一版的 1.3.24 最大的差異就在穩定度 (Scalability)，除此之外，尚包含以下許多的核心與模組新功能：

2-1 核心新功能

■ Unix Threading

在支援 POSIX 執行緒 (Thread) 的 UNIX 系統上，Apache 目前可以執行混合式的多程序 (Multi-process) 和多執行緒 (Multi-thread) 模式，這有助於多種組態的穩定性 (但不適用於所有的組態)。

■ 支援多種通訊協定

Apache 2.0 可支援更多的通訊協定，以增加其網路功能。

■ 新式的系統架構

新式的系統架構已被改寫為 autoconf 與 libtool 的型態，這使用 Apache 的組態設定和其他套件更為相似。

■ 對於非 Unix 平台有較佳的支援

由於 MPM 和 Apache Portable Runtime (APR) 技術的使用，使得在 BeOs、os/2 和 Windows 等非 Unix 平台上，Apache 2.0 具有較快及較穩定的表現，因為這些平台可以使用原始的 API (Application Programming Interface)，以避免模擬 POSIX 環境時引起的效能下降。

- 新型的 API
在 2.0 版的 Apache 中，各模組的 API 有較明顯的改變，同時也改進了 1.3 版時的模組順序及優先等級的問題，因為 Apache 2.0 會自動處理有關先後順序的工作。
- 篩選功能 (Filtering)
Apache 模組現在已被改寫為 Filtering 類型，當它們作用在串流的內容時，就如同由伺服器接收或傳送一般。例如在「mod_include」模組中，可允許使用「INCLUDES」篩選器中的 SSI (Server Side Includes) 指令，以剖析 CGI 指令檔的輸出。
- 多種語言的錯誤回應訊息
利用 SSI 文件，管理員可以自訂不同與演版本的錯誤訊息，以回應至用戶端瀏覽器。
- 簡化的組態設定
在 Apache 2.0 中，簡化了許多容易混淆的指令，例如「Port」和「BindAddress」指令以取消，若要鏈結 (Binding) 伺服器的 IP 位址，只需要使用「Listen」指令，而利用「ServerName」指令所設定的伺服器名稱與連接埠號碼，僅適用於 Redirect 和 Virtual Host 的識別。
- IPv6 支援
根據預設值，在支援 APR 函式庫 IPv6 系統中，Apache 2.0 都可監聽 IPv6 的通訊連接埠。此外，〈Listen〉、〈NameVirtualHost〉和〈VirtualHost〉指令 (Directive) 也可支援 IPv6 中的數字位址字串，例如「Listen [fe80::1]:8080」。
- 支援 Windows NT/2000/XP 萬國碼 (Unicode) 功能
安裝於 Windows NT 上的 Apache 2.0，使用「UTF-8」為所有檔案名稱的編碼方式，因此可以直接支援多種語系，但不支援 Windows 95/98/ME。
- 正規表示法 (Regular Expression) 函式庫更新
因 Apache 2.0 包含了「Perl Compatible Regular Expression Library」(PCRE)，所以目前在正規表示法中可使用較具效率的 Perl 5 語法。

2-2 模組新功能

- **mod_auth_digest**
使用 Share Memory 來支援程序間的交易快取，此外，使用 MD5 摘要試驗正法來驗證使用者帳號的正確性。但若網路中使用多種不同類型的瀏覽器時，不建議使用此模組的功能，因為當不支援 MD5 的瀏覽器存取時，Apache 伺服器將不會對其要求進行驗證。
- **mod_auth_ldap**
允許使用 LDAP 目錄來儲存 HTTP 基本驗證的資料庫。
- **mod_auth_dbm**
使用 DBM (DataBase Management) 檔案以驗證使用者身份，也就是利用使用者名稱與純文字密碼的對應方式，此功能與「mod_auth_file」模組相似。
- **mod_autoindex**
允許「目錄瀏覽」(Directory Indexing) 功能，所謂目錄瀏覽是指當網頁目錄沒有包含預設的網頁檔案時，例如 index.html，此功能便可以將此目錄中的所有檔案及目錄以排列的方式，就如同執行 Linux 中的「ls」或 Windows 中的「dir」指令一樣。也可以使用 HTML 標籤來設計清單的格式，並且支援排序的功能。
- **mod_charset_lite**
這是一個尚在實驗階段的模組，它可以在網頁傳送回用戶端前，指定來源物件及欲翻譯成的字元集，但它不直接翻譯 Data，而是告知 Apache 執行何種翻譯。
- **mod_dav**
此模組可以起用 WebDav (Web-base Distributed Authoring and Versioning) 的功能，它允許使用者存取管理位於遠端主機上的目錄及檔案。
- **mod_deflate**
此模組可提供「DEFLATE」輸出篩選器的功能，它可以在回應透過網路傳回用戶端前，先對檔案進行壓縮。

- **mod_file_cache**
這個模組包含 Apache 1.3 中的「mod_mmap_static」模組功能，並且將某些經常被存取的網頁快取在個別的記憶體中，以提高快取的功能，這可減少伺服器啟動程序和回應要求的時間。
- **mod_headers**
若需要自定由「mod_proxy」模組產生的 HTTP 表頭，您可以啟用 mod_headers 模組的功能，它會傳送額外的資訊給瀏覽器、在每個要求中附加時間標記，或是其他有用的資訊。
- **mod_include**
這個模組是用於伺服器剖析的 HTML 文件（Server Side Includes，SSI），它可在網頁回應用戶端之前先傳送至篩選器，而由這個篩選器設定不同的變數（由 SSI 的設定所定義）。
- **mod_negotiation**
新增的「ForceLanguagePriority」指令可用來確認用戶端接收的是單一的文件，而不是「NOT ACCEPTABLE」或「MULTIPLE CHOICES」的回應。此外，模組會先檢視用戶端要求中有關自己的資訊，然後傳送最適合或是尋找伺服器中最符合的檔案至用戶端，這種功能就稱為「內容協商」（Content Negotiation）。
- **mod_proxy**
這個模組已被完全的改寫，使得 Apache 足以擔任 Proxy 或閘道器的角色，它可以代理 FTP、SSL、HTTP/1.1 等通訊協定，同時也可以連接到其它的 Proxy 模組。
- **mod_ssl**
啟用 mod_ssl 模組可以提供 SSL（Secure Sockets Layer）v2/v3 和 TLS（Transport Layer Security）v1 等通訊協定的支援能力。

3 Apache 相關套件及設定檔

Apache 的相關套件及設定檔如下：

```
Daemon:apache2
```

```
Daemon 類別：System V standalone daemon
```

所需套件：

- apache2-2.*rpm
- apache2-prefork*.rpm
- apache2-example-pages*.rpm

啟動 Script：/etc/init.d/apache2

Port：80 (http)，443 (https)

設定檔：/etc/apache2/ *

Log 檔：/var/log/apache2

4 安裝 Apache

在 SLES 9 利用「YaST 新增或移除軟體」模組安裝 Apache 是件很容易的事，啟動方法如下：

【方法一】圖形介面：點選工具列「開始」→系統→YaST→系統→新增或移除軟體

【方法二】圖形介面：於命令列輸入「yast2」→系統→新增或移除軟體

【方法三】圖形介面：於命令列輸入「yast2 sw_single」

【方法四】文字介面：輸入「yast」→Software→Install and Remove Software

【方法五】文字介面：輸入「yast sw_single」

利用「YaST 新增或移除軟體」模組的搜尋功能，鍵入關鍵字「apache」出現如圖 1 的畫面，勾選 apache2、apache2-example-pages 及 apache2-prefork 套件。

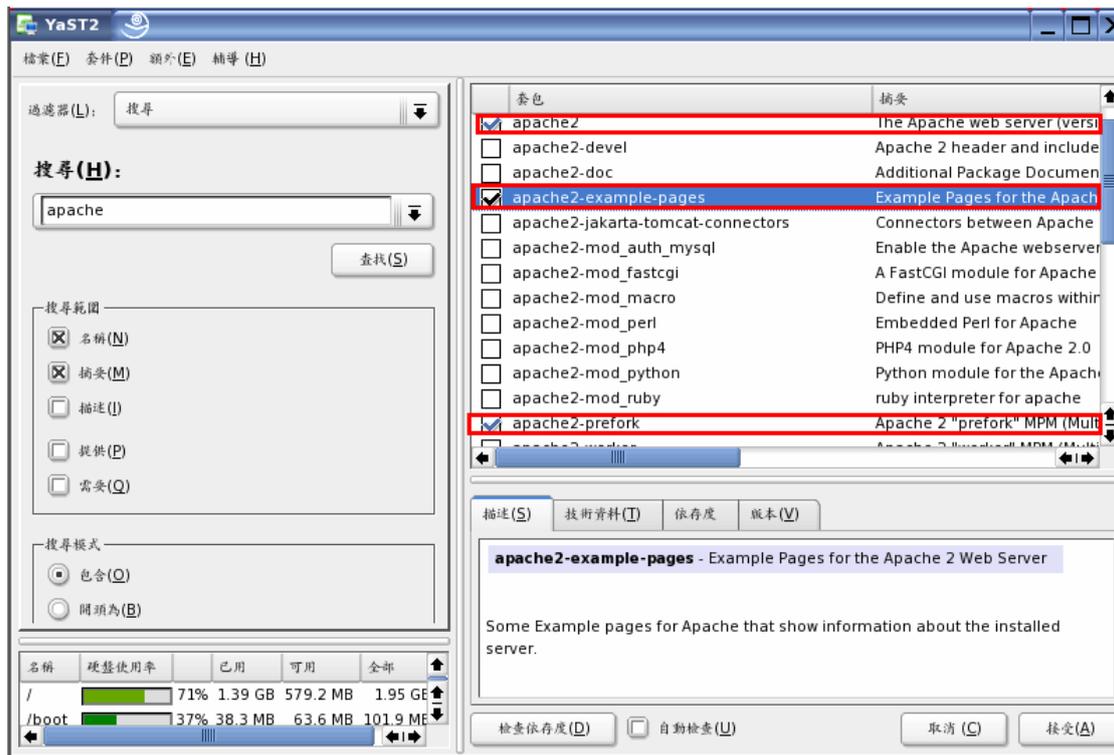


圖 1：安裝 Apache 相關套件畫面

勾選所需套件後，按下「接受 (A)」，會出現提示放入光碟的訊息，放入適當的光碟片，便可順利安裝。

要建置一個有內容的網站，除了安裝 Apache 外，html 網頁的製作撰寫也是不可或缺的一環。

HTML 就是超文件標示語言 (Hyper Text Markup Language)。主要是在瀏覽器中顯示一份文件的內容。它本身並非程式設計，只不過是標示 (Mark-up)，用來強調及組織一般性的文字。HTML 是由一些標記 (tags) 所構成，每個標記對瀏覽器下一個特定的指令，這些指令會告訴瀏覽器如何顯示文件的內容。

HTML是由很多的文件標記(Document Tags)所組成，這些標記在你建立的每一個HTML文件都會用到。它們會對文件檔中不同的部份作不同的定義。大部分的標記都是成對的(圍堵標記 Container Tags)，即包含『起始標記』和『結束標記』。如<HTML>表示網頁的開始，而</HTML>代表網頁的結束。起始標記和結束標記差別就在於前面的斜線符號而已。標記本身沒有大小寫的區分，不過建議是使用大寫字母，以便在文字編輯器中容易識別。

不過本篇文章並不是教導網頁寫作的書籍，筆者採用幾個的實用的範例來實作 Apache 的基本功能。詳細 html 的語法，讀者可以參考相關的書籍。

5 實例演練：公司網頁

一般企業建置WWW Server就是希望客戶打開瀏覽器，在網址欄內鍵入公司的主機名稱時可看到該公司所設計的網頁。本演練的目的就是告訴讀者該將公司網站的首頁命名為何（DirectoryIndex）？置於何目錄（DocumentRoot）？利用瀏覽器開啟<http://公司網頁主機>便可看到公司網站的首頁。

實作環境：

SUSE Linux Enterprise Server 9 + SP1

實作步驟：

步驟 1：啟動 apache server

讀者可利用「`rcapache2 start`」指令啟動apache server：

```
sles9:~ # rcapache2 start
Starting httpd2 (prefork) done
```

亦可直接執行apache system V script

```
sles9:~ # /etc/init.d/apache2 start
Starting httpd2 (prefork) done
```

待apache server啟動成功後，打開瀏覽器，在網址欄鍵入<http://主機名稱>便可看到如圖2的Apache預設網頁。若想重新開機時，系統預設會啟動Apache可執行下列指令：

```
sles9:~ # chkconfig apache2 on
```

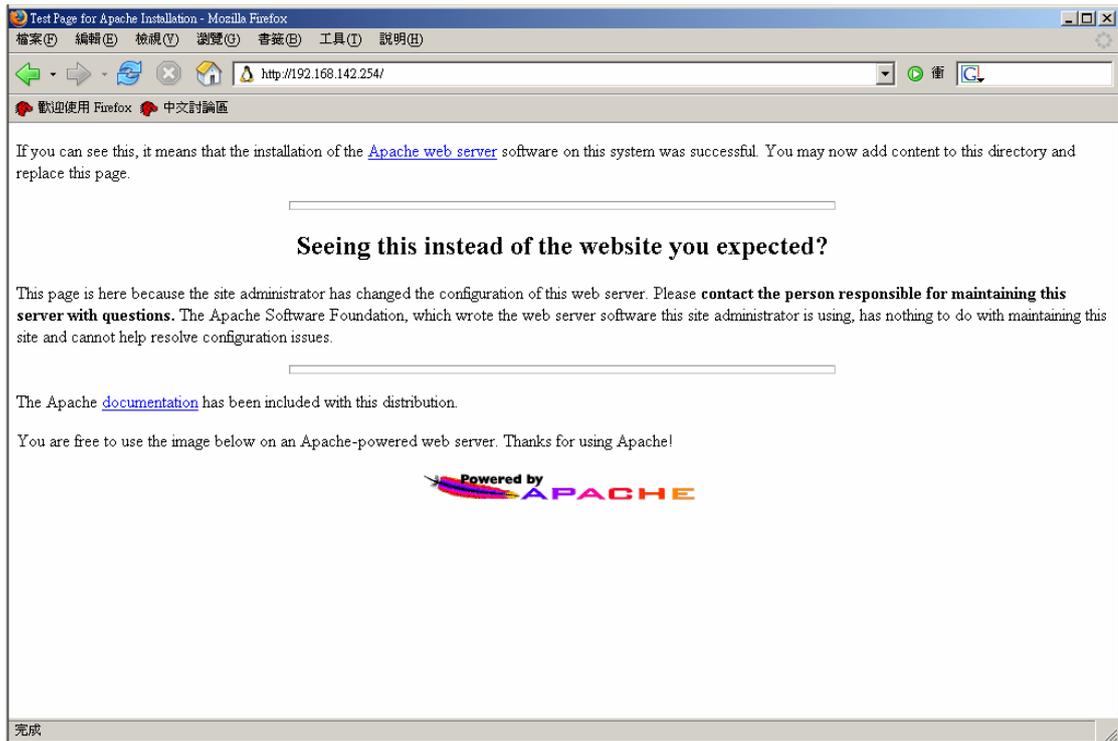


圖2：Apache預設網頁

步驟 2：修改預設網頁

讀者只需到/srv/www/htdocs（預設的DocumentRoot）下建立index.html，則<http://主機名稱>所看到的畫面就會是index.html（預設的DirectoryIndex）的內容（圖3）。

```
sles9:/srv/www/htdocs #echo "sles9.suse.com homepage" > index.html
```

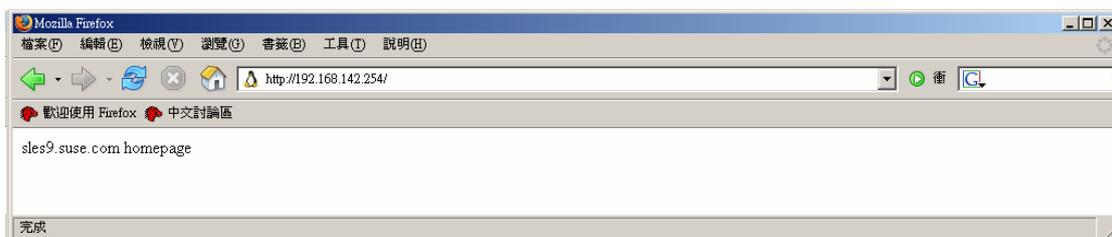


圖 3：修改 Apache 預設網頁

但若是 index.html 內有中文則會遭遇無法正常顯示的問題。例如筆者利用記事本撰寫 index.html 內容如圖 4 所示，然後利用 FTP 方式上傳至 WWW Server 上的 /srv/www/htdocs 目錄。

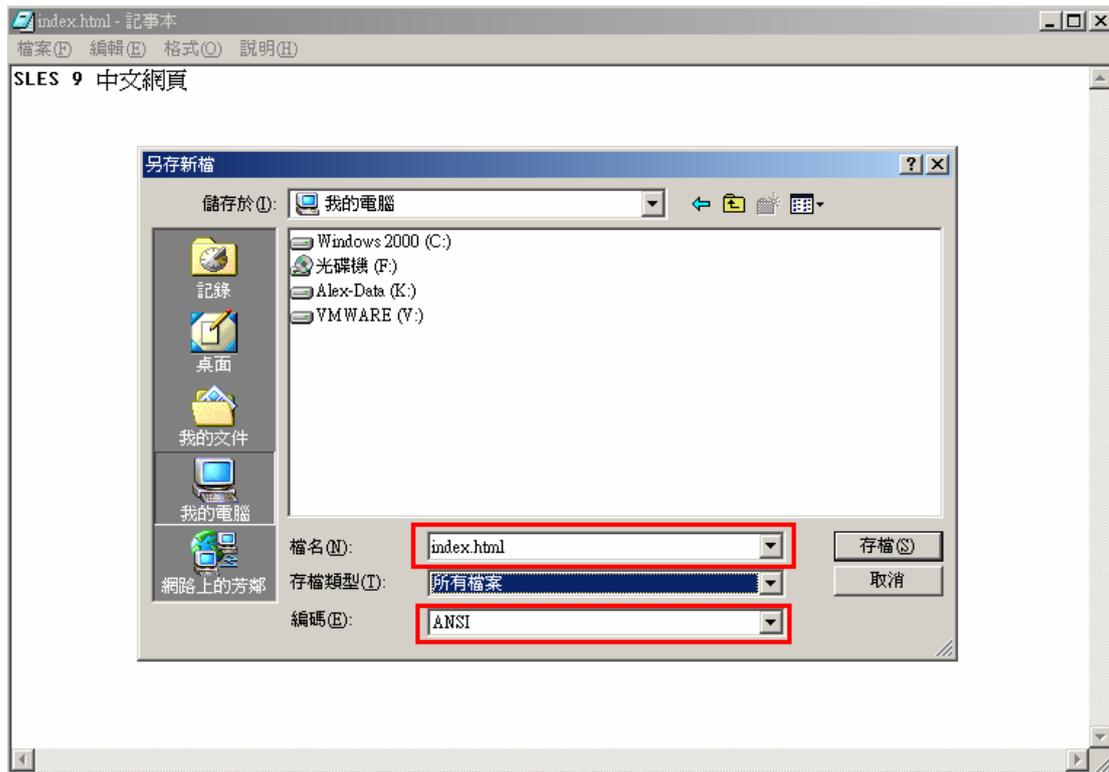


圖4：利用記事本編寫index.html

打開 Linux 上的瀏覽器或 Windows 上的瀏覽器觀看此網頁，都是出現一行亂碼（圖 5）。原因是 SLES 上 Apache 預設採用英文/西歐語系（ISO-8859-1），而使用記事本或一般的網頁編輯軟體存檔時預設是用 Big5 碼儲存文件，所以才會無法正常顯示。

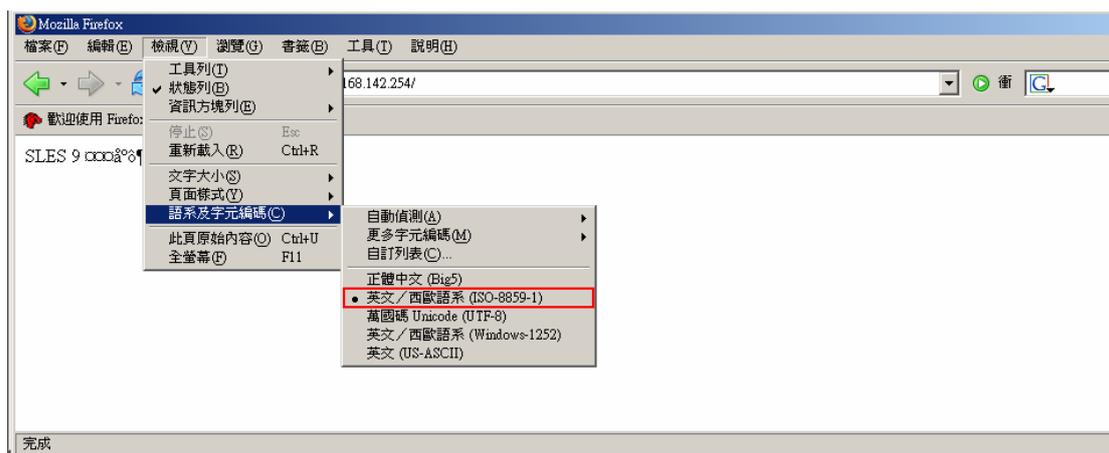


圖 5：無法正常顯示含有中文字的網頁

步驟 3：解決中文網頁亂碼問題

中文亂碼的解決方法很多，其中之一便是修改 Apache 上預設的編碼方式。讀者可修改 `/etc/apache2/mod_mime-defaults.conf` 中約 60 行的「AddDefaultCharset

ISO-8859-1」。

在此行前面加上#，註解掉此設定值，則會採用Brower上的語系設定值。
(註：IE此設定值存放於「工具」→「網際網路選項」→「一般」→「語言」)
或者是將此行改為「AddDefaultCharset Big5」，則瀏覽器將用Big5方式來解讀
網頁內容。最後執行「rcapache2 restart」重新啟動WWW伺服器。

sles9:/etc/apache2 # vi mod_mime-defaults.conf

```
59 #
60 #AddDefaultCharset ISO-8859-1
1.在AddDefaultCharset ISO-8859-1加上#
2.或改為AddDefaultCharset Big5
62
63
64 AddCharset ISO-8859-1 .iso8859-1 .latin1
65 AddCharset ISO-8859-2 .iso8859-2 .latin2 .cen
```

sles9:~ # rcapache2 restart

Syntax OK

Shutting down httpd2 (waiting for all children to terminate) done

Starting httpd2 (prefork) done

重新測試時，記得先清除瀏覽器上快取空間，然後在網址欄鍵入<http://主機名稱>便可看到如圖6正常顯示中文的網頁。

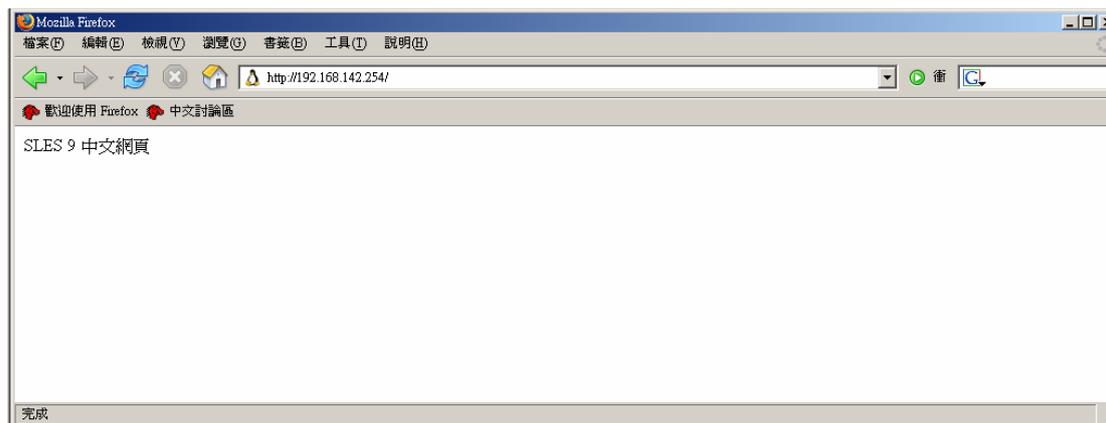


圖 6：正常顯示含有中文字的網頁

6 實例演練：個人網頁

學校或各 ISP 有時會提供空間給個人存放網頁，通常他們只是告訴使用者首頁必需命名為 index.html 或 default.htm，然後放到某目錄下。然後在瀏覽器鍵入 <http://主機名稱/~使用者名稱>（網頁主機通常是 Unix Like）或 <http://主機名稱/使用者名稱>（網頁主機可能是 Windows）便可看到使用者專屬的網頁。本演練目的便是告訴讀者要開放使用者可自製網頁，需要那些設定工作。

步驟 1：在 UserDir 內建立 index.html

SLES 9 Apache 預設開啟 UserDir 的功能，什麼是使用者 UserDir，就是 Apache 發現瀏覽器網址欄為 <http://主機名稱/~使用者名稱> 時，會對應至使用者在其家目錄下特定的子目錄，預設為 public_html 目錄，此目錄稱為 UserDir，便是使用者存放個人網頁的位置。

轉換至一般使用者身份，在其家目錄下的 public_html 子目錄（預設的 UserDir）子目錄內撰寫 index.html。

```
sles9:~ # su - geeko
geeko@sles9:~> cd public_html/
geeko@sles9:~/public_html>echo "Geeko's Homepage" > index.html
```

此時在瀏覽器網址欄中鍵入 <http://主機名稱/~geeko> 便可看到 geeko 的個人網頁（圖 7）。

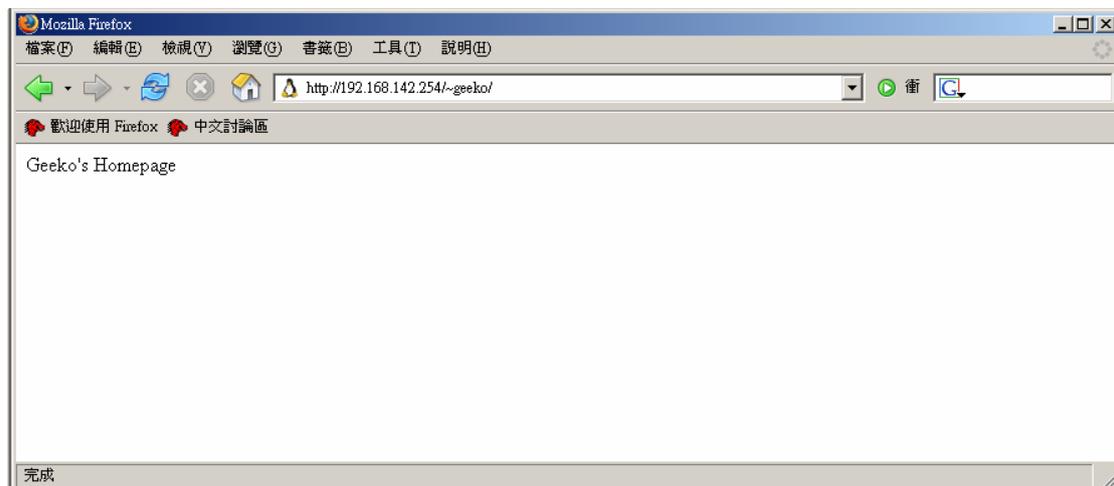


圖 7：個人網頁

步驟 3：建立/var/www/html/neo soft link

為了在瀏覽器鍵入<http://主機名稱/geeko>同樣可看到geeko的個人網頁，筆者使用root身份建立/srv/www/htdocs/geeko soft link 指向/home/geeko/public_html/

```
sles9:~ # ln -s /home/geeko/public_html/ /srv/www/htdocs/geeko
sles9:~ # ls -l /srv/www/htdocs/geeko
lrwxrwxrwx  1 root root 24 Jul 12 18:37 /srv/www/htdocs/geeko ->
/home/geeko/public_html/
```

但當你測試時，會發現如圖8的錯誤畫面，原因是SLES 9上的Apache預設不允許/srv/www/htdocs目錄的內容以soft link的方式指向他處。

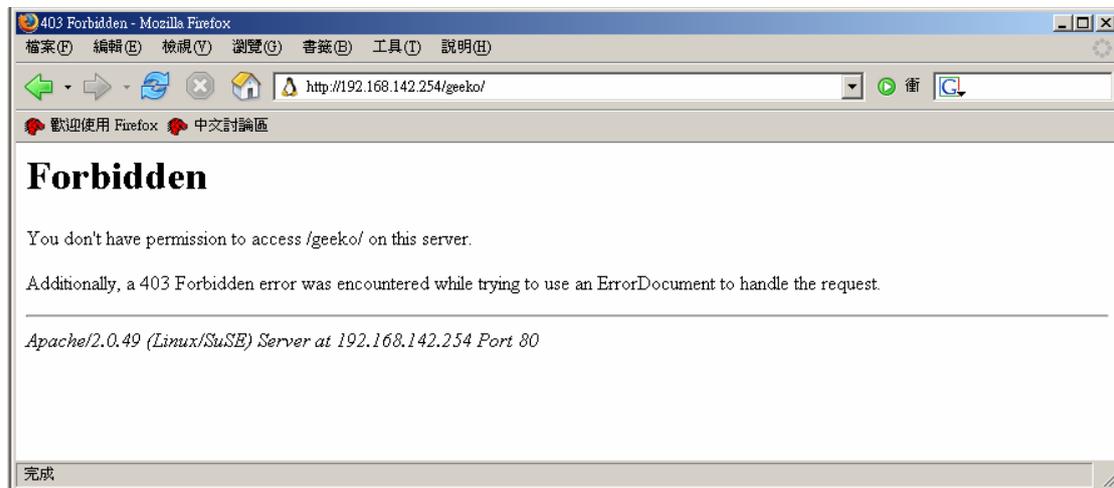


圖 8：<http://主機名稱/使用者名稱> 錯誤畫面

要解決上述問題，必須以root身份修改/etc/apache2/default-server.conf設定檔中 有關/srv/www/htdocs網頁目錄的設定，將第22行的「Options None」改為Options 「Options FollowSymLinks」，然後重新啟動apache伺服器即可。

```
9 # Configure the DocumentRoot
10 #
11 <Directory "/srv/www/htdocs">
12     # Possible values for the Options directive are "None", "All",
13     # or any combination of:
14     # Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
15     #
16     # Note that "MultiViews" must be named *explicitly* --- "Options All"
17     # doesn't give it to you.
```

```
18      #
19      # The Options directive is both complicated and important. Please see
20      # http://httpd.apache.org/docs-2.0/mod/core.html#options
21      # for more information.
22      Options None → 改為 Options FollowSymLinks
23      # AllowOverride controls what directives may be placed in .htaccess files.
24      # It can be "All", "None", or any combination of the keywords:
25      #   Options FileInfo AuthConfig Limit
26      AllowOverride None
27      # Controls who can get stuff from this server.
28      Order allow,deny
29      Allow from all
30 </Directory>
```

```
sles9:~ # rcapache2 restart
Syntax OK
Shutting down httpd2 (waiting for all children to terminate)      done
Starting httpd2 (prefork)                                         done
```

重新測試時，在瀏覽器鍵入<http://主機名稱/geeko>同樣可看到geeko的個人網頁（圖9）。



圖9：<http://主機名稱/使用者名稱>畫面

7 實例演練：Name Base 虛擬站台 (Virtual Host)

所謂虛擬站台 (Virtual Host) 就是在一台主機存放多個網站，常用的實作方式有兩種：IP Base 和 Name Base，這個實例演練將介紹 Name Base 虛擬站台 (Virtual Host) 的實作方式。

IP Base Virtual Host：利用 IP 來決定要連線到那個網站的內容，也就是說不同的 IP 對應到不同的網站。因為是根據 IP 來分辨，所以這台主機必須同時綁多個 IP 在其網卡上。

Name Base Virtual Host：利用欲連線的主機名稱 (ServerName) 來決定要連線到那個網站的內容，也就是說不同的主機名稱對應到不同的網站。而這幾個不同的主機名稱可以是同一個 IP，所以沒有 IP Base Virtual Host 得同時綁多個 IP 在其網卡上的限制。

虛擬站台的技術除了可以應用在 ISP 的主機託管業務外，也可拿來應用到行銷導向的公司。例如在前面的實例演練中，我們建置了個人網頁。在瀏覽器鍵入 <http://主機名稱/~geeko> 或 <http://主機名稱/geeko> 均可看到個人網頁。假如可簡化到在瀏覽器鍵入 <http://geeko.suse.com> 便看到個人網頁，豈不更加方便易記。

假設 suse.com 這家公司有很多的業務員，我們可以幫所有業務員建立虛擬站台，這樣印在名片上的個人網址可由 <http://sles9.suse.com/~geeko> 變為 <http://geeko.suse.com>，這樣是不是更加神氣點！

此演練欲完成的結果為：當在瀏覽器鍵入 <http://sles9.suse.com> 時可看到公司網頁；鍵入 <http://geeko.suse.com> 時可看到 geeko 的個人網頁。

假設網頁主機的 IP 為 192.168.142.254，在實作之前，我們先來思考原理，既然 <http://sles9.suse.com> 和 <http://geeko.suse.com> 均指向同一台主機，所以得麻煩 DNS 伺服器設定相對應的 Resource Record 將這兩個主機名稱均解析成同一個 IP 192.168.142.254。

接下來，當 Apache 收到來自 <http://sles9.suse.com> 和 <http://geeko.suse.com> 瀏覽器的需求時，該如何分辨呢？別擔心，當 <http://sles9.suse.com> 瀏覽器送出需求時，會註明欲連線網路主機的 IP 外，亦會將網址欄內的所輸入的字串

「sles9.suse.com」一併送出。同理，<http://geeko.suse.com> 瀏覽器除欲連線網路主機的 IP 外，亦會將網址欄內的所輸入的字串「geeko.suse.com」一併送出。

那 Apache 該做什麼事呢？Apache 最主要的工作有二：

- 註明那個 IP 為 Name Virtual Host 的 IP
- 根據送來的字串 sles9.suse.com 或 geeko.suse.com，此字串稱為 ServerName。不同的 ServerName 對應到不同的目錄，即對應到不同的 DocumentRoot。

步驟 1：修改 suse.com 的 DNS 的 Zone File

為了 sles9.suse.com 和 geeko.suse.com 均指向同一 IP 192.168.142.254，所以在 suse.com 的 DNS 伺服器設定一筆 CNAME Resource Record，設定 geeko.suse.com 為 sles9.suse.com 的 CNAME。

```
sles9:~ # vi /var/lib/named/master/suse.com.zone
```

```
...
sles9.suse.com.      IN      A       192.168.142.254
geeko.suse.com.     IN      CNAME   sles9 ←加入 CNAME R.R
...
```

```
sles9:~ # rncamed restart ←重新啟動 named
```

```
Shutting down name server BIND           done
Starting name server BIND                done
```

```
sles9:~ # host geeko.suse.com ←查詢 geeko.suse.com 的 IP
```

```
geeko.suse.com is an alias for sles9.suse.com.
sles9.suse.com has address 192.168.142.254
```

步驟 2：修改 apache 設定檔

這個部份設定較為繁瑣，不過只要清楚原理就不易出錯，Apache 最主要得設定兩件事：

- 註明那個 IP 為 Name Virtual Host 的 IP
- 不同的 ServerName 對應到不同的 DocumentRoot。

```
sles9:~ # vi /etc/apache2/default-server.conf
```

```
...在這檔案最後加上下列設定
```

```
NameVirtualHost 192.168.142.254 ←註明那個 IP 為 Name Virtual Host 的 IP
<VirtualHost 192.168.142.254>
  ServerName sles9.suse.com
  DocumentRoot /srv/www/htdocs
</VirtualHost>
```

```
<VirtualHost 192.168.142.254>
  ServerName geeko.suse.com
  DocumentRoot /home/geeko/public_html
</VirtualHost>
```

步驟 3：重新啟動 apache 設定檔

```
sles9:~ # rcapache2 restart
Syntax OK
Shutting down httpd2 (waiting for all children to terminate)      done
Starting httpd2 (prefork)                                          done
```

步驟 4：測試

在瀏覽器鍵入 <http://sles9.suse.com>，如圖 10。

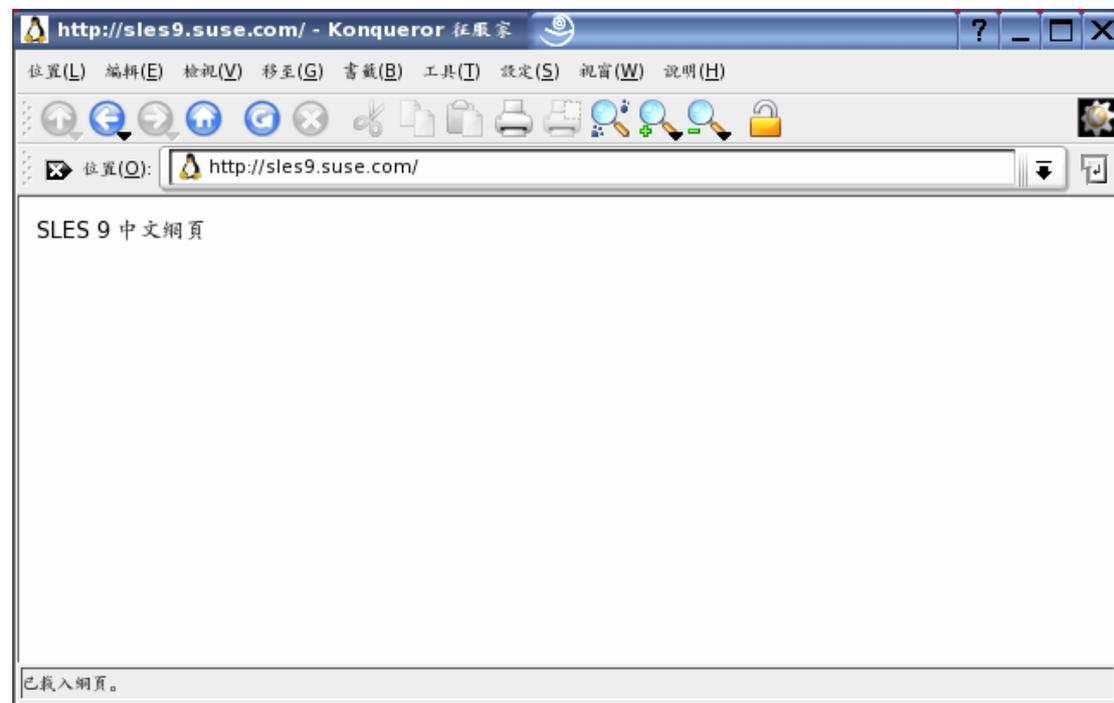


圖 10：<http://sles9.suse.com> 測試結果

在瀏覽器鍵入 <http://geeko.suse.com>，如圖 11。

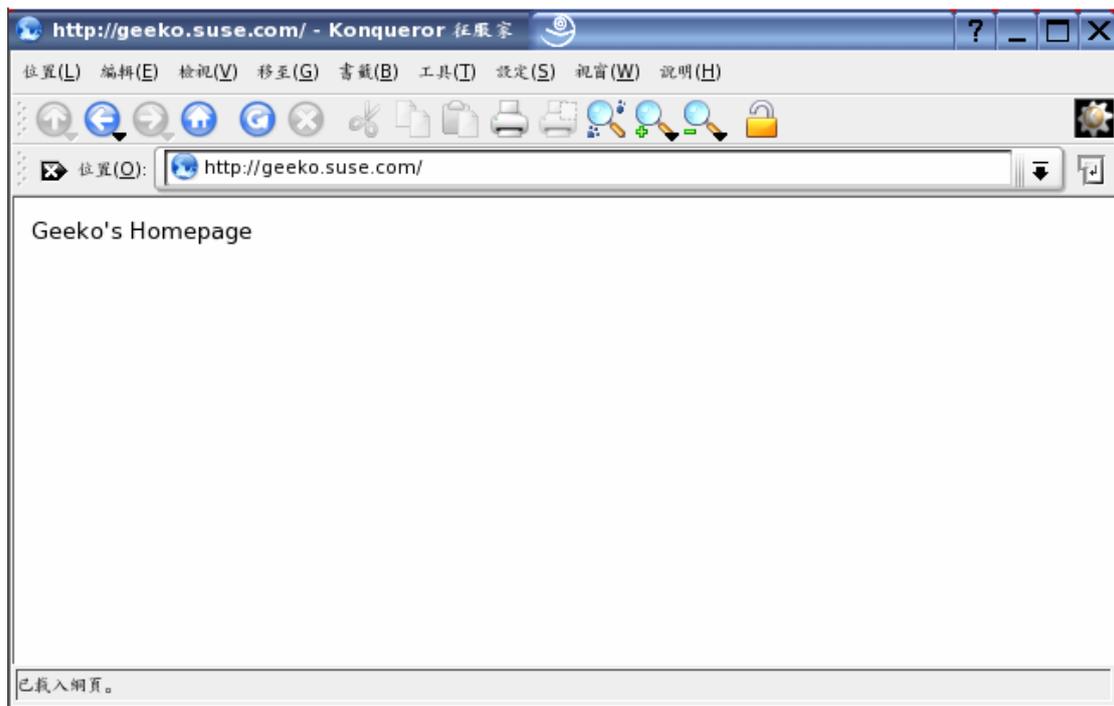


圖 11：<http://geeko.suse.com> 測試結果

8 實例演練：目錄保護機制（.htaccess）

讀者一定有過進入某個網站之後，按下某些連結，竟然出現一個對話框，告訴你要登入該目錄，需要輸入帳號與密碼才能登入的經驗。利用Apache所提供的『.htaccess目錄認證保護』機制便可實作此功能。這種認證的模式彈性很大，不用寫程式控制就可限制某些帳號及密碼才可進入此目錄，使網頁資料只有註冊使用者才可存取。此實例演練目的便是當你在瀏覽器輸入<http://主機名稱/nclp>時會出現輸入帳號與密碼認證對話框。

步驟 1：建立/srv/www/htdocs/nclp 目錄

既然是『進入某個目錄之後，才會出現認證對話視窗』，那麼首先當然就是要設定認證網頁的『目錄』！而<http://主機名稱/nclp>實際上對應系統的/srv/www/htdocs/nclp。所以首先建立/srv/www/htdocs/nclp目錄，並在此目錄下編輯index.html。

```
# mkdir /srv/www/htdocs/nclp
# echo "NCLP Homepage" > /srv/www/htdocs/nclp/index.html
```

步驟 2：指定/srv/www/htdocs/nclp 採用.htaccess 認證

接下來，當然就是到 default-server.conf 檔案中去指定保護

/srv/www/htdocs/nclp 這個目錄。在檔案最後加入下列文字，然後重新啟動 apache 伺服器。

sles9:~ # vi /etc/apache2/default-server.conf

...在檔案最後加上下列設定

```
<Directory "/srv/www/htdocs/nclp">  
    AllowOverride authconfig  
</Directory>
```

步驟 2：重新啟動 apache 設定檔

```
sles9:~ # rcapache2 restart
```

Syntax OK

Shutting down httpd2 (waiting for all children to terminate) done

Starting httpd2 (prefork) done

步驟 3：編寫.htaccess

在/srv/www/htdocs/nclp 目錄下產生一個**.htaccess** 檔，內容如下：

```
sles9:/srv/www/htdocs/nclp # cat .htaccess
```

AuthName "NCLP Page" ← 這個是顯示在視窗上面的提示字元

AuthType basic ← 認證的類型

AuthUserFile /srv/www/htdocs/nclp/wwwpasswd ← 帳號密碼檔放置的地方

require valid-user ← wwwpasswd檔內的帳號皆可登入

步驟 4：建立密碼檔及設定可進入此目錄的帳號及密碼

```
sles9:/srv/www/htdocs/nclp#htpasswd2 -c wwwpasswd alex
```

第一次執行 htpasswd2 指令時，因為還未有 wwwpasswd 檔案，所以必須指定 -c (create) 參數以建立 wwwpasswd 檔案

New password:

Re-type new password:

Adding password for user alex

```
sles9:/srv/www/htdocs/nclp # htpasswd2 wwwpasswd eric
```

New password:

Re-type new password:

Adding password for user eric

```
sles9:/srv/www/htdocs/nclp # cat wwwpasswd
```

alex:/WNJ8b0zn.U/k

eric:3IBMDfEob9aRA

步驟5：測試

打開瀏覽器在位址欄輸入http://sles9.suse.com/nclp時就會出現目錄認證視窗（圖8-12），必須輸入步驟4所建立的帳號及對應的密碼才可進入此網頁目錄。

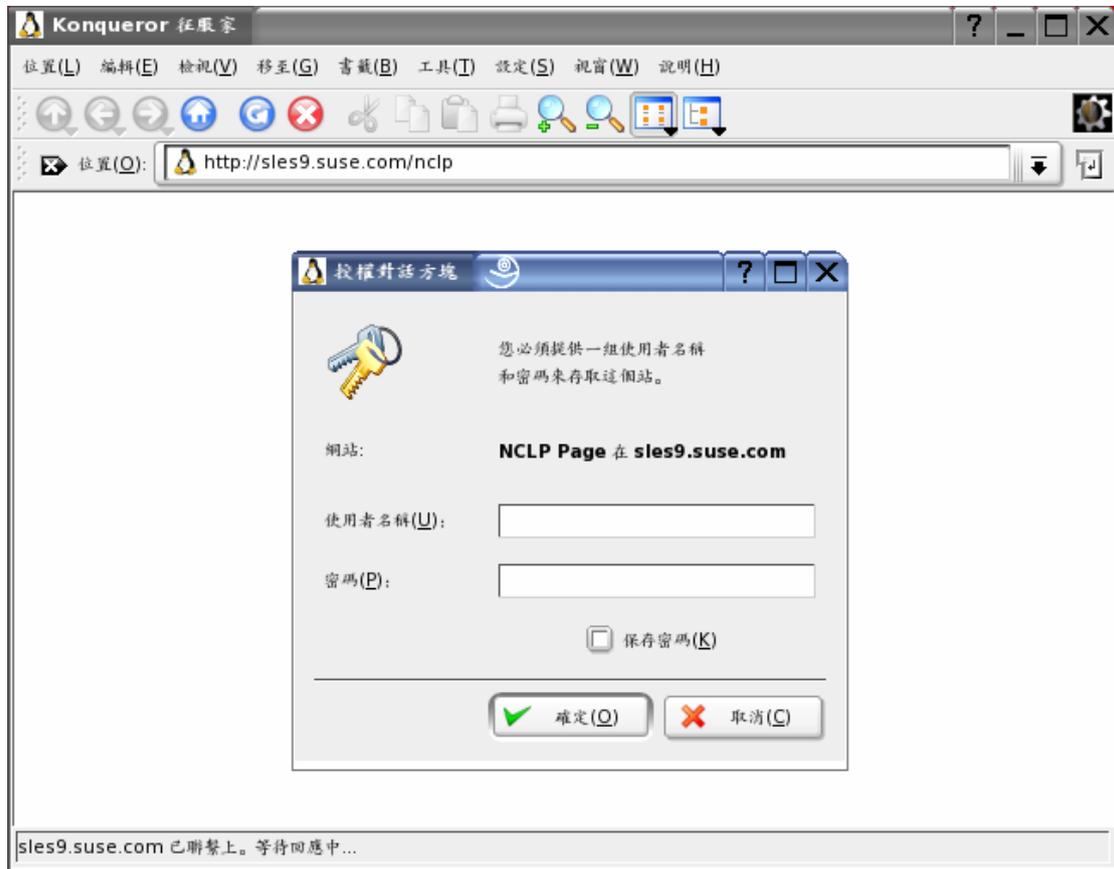


圖8-12：.htaccess認證視窗

是不是蠻方便的，讀者看到這裏滿可能會想到如果一般使用者也希望利用.htaccess的機制來保護他們的個人的網頁目錄。難道開放讓每個使用者皆可修改default-server.conf加入他們想保護的網頁目錄嗎？

其實只要root先幫所有使用者設定他們的UserDir要啟用.htaccess保護機制。設定方法為在default-server.conf最後加上：

```
<Directory "/home/*/public_html">  
    AllowOverride authconfig  
</Directory>
```

然後重新啟動apache伺服器，由每個使用者自行決定是否要撰寫.htaccess檔，若沒有.htaccess檔案，則無需輸入帳號密碼便可存取網頁。

重點整理

■ 何謂 DocumentRoot，其預設值為何？

當我們在瀏覽器在鍵入 <http://主機名稱/>時，代表我們想看這個網站的最上層的網頁資料。網頁資料可想成 Document，最上層的意思就如同根目錄 root。所以 Apache 把 <http://主機名稱/>所對應的網頁資料的稱為 DocumentRoot，SLES 9 上的 Apache 預設的 DocumentRoot 的位置在 /srv/www/htdocs 目錄，此設定存於 /etc/apache2/default-server.conf 檔案中，檔案內容如下：

```
...  
DocumentRoot "/srv/www/htdocs"  
...
```

■ 何謂 DirectoryIndex，其預設值為何？

在瀏覽器在鍵入 <http://主機名稱/some-directory/>時，並未指定要閱讀此目錄下何網頁時，apache 預設會去搜尋 /some-directory/ 目錄下的 index.html。index.html 即所謂 "DirectoryIndex"。

有關 DirectoryIndex 的設定存放於 /etc/apache2/httpd.conf，檔案內容如下：

```
...  
# List of resources to look for when the client requests a directory  
DirectoryIndex index.html index.html.var  
...
```

■ 試述 VirtualHost 的用途、技術

所謂虛擬站台 (Virtual Host) 就是在一台主機存放多個網站，常用的實作方式有兩種：IP Base 和 Name Base，這個實例演練將介紹 Name Base 虛擬站台 (Virtual Host) 的實作方式。

IP Base Virtual Host：利用 IP 來決定要連線到那個網站，也就是說不同的 IP 對應到不同的網站。因為是根據 IP 來分辨，所以這台主機得同時綁多個 IP 在其網卡上。

Name Base Virtual Host：利用欲連線的主機名稱 (ServerName) 來決定要連線到那個網站，也就是說不同的主機名稱對應到不同的網站。而這幾個不同的主機名稱可以是同一個 IP，所以沒有 IP Base Virtual Host 得同時綁多個 IP 在其網卡上的限制。

彥明有感：

廣告詞：「小時候想趕快長大，想談戀愛，想獨立。當我長大了，卻想回家！以家的價值，啟動前進的動力。」。最近可能是因為工作量太大，右手竟不時會有發麻的情形發生。去年筆者的父親生了一場重病，今年母親身體也出了些狀況，深深感受到身體健康的重要，也讓筆者驚覺父母的年事已高，需人照料。以家的價值，啟動前進的動力，也期許自己能早點有能力及時間好好照顧家人，和大家互勉之！