

教育部顧問室「製商整合科技教育改進」計畫
九十二年度執行成果---教材與教案

計畫名稱：淡江製商整合科技產學合作教育改進計畫

教材名稱：

電子商業伺服器管理

指導老師：魏世杰 博士

教材編撰：陳世偉、林世唐、林志龍、謝逢鳴

編撰日期：中華民國九十二年十二月

Ch1. 電子商務伺服器管理.....	6
1.1 電子商務發展背景：.....	6
1.2 本書架構：.....	7
Ch2 Linux 系列之安裝設定.....	8
2.1 DNS 伺服器.....	8
2.1.1 前言.....	8
2.1.2 安裝.....	8
2.1.2.1 手動安裝.....	8
2.1.2.2 由安裝程式勾選.....	9
2.1.3 啟動 DNS 伺服器.....	10
2.1.4 DNS 伺服器設定檔.....	12
2.1.5 結語.....	17
2.2 DHCP 伺服器.....	18
2.2.1 前言.....	18
2.2.2 安裝.....	18
2.2.2.1 手動安裝.....	18
2.2.2.2 由安裝程式勾選.....	19
2.2.3 啟動 DHCP 伺服器.....	20
2.2.4 DHCP 伺服器設定檔.....	20
2.2.5 結語.....	22
2.2.6 DHCP 伺服器設定檔範例：(取自 dhcp-3.0p12/system/dhcpd.conf)	22
.....	22
2.3 SAMBA 伺服器.....	26
2.3.1 前言.....	26
2.3.2 安裝.....	26
2.3.2.1 手動安裝.....	26
2.3.2.2 由安裝程式勾選.....	27
2.3.3 啟動 SAMBA 伺服器.....	28
2.3.4 開機時啟動.....	28
2.3.5 SAMBA 伺服器設定檔.....	29
2.3.6 結語.....	32
2.3.7 SAMBA 伺服器設定檔範例：(取自 smba-3.0.1/packaging/RedHat/smb.conf).....	32
2.4 Proxy 伺服器.....	40
2.4.1 前言.....	40
2.4.2 安裝.....	40
2.4.2.1 手動安裝.....	40
2.4.2.2 由安裝程式勾選.....	41

2.4.3	啟動 Proxy 伺服器	42
2.4.4	Proxy 伺服器設定檔	42
2.4.5	結語	45
2.5	Apache	46
2.5.1	Apache Server	46
2.5.2	Apache 編譯和安裝	46
2.5.3	編輯設定檔	48
2.6	Mysql	55
2.6.1	Mysql 簡介	55
2.6.2	Mysql 安裝	57
2.6.3	Mysql 簡單操作	58
2.6.4	Mysql 權限控制	64
2.7	Ftp	68
2.7.1	Proftpd 簡介	68
2.7.2	Proftpd 下載和安裝	69
2.7.3	Proftpd 的設定檔	70
2.7.4	編輯歡迎檔	74
2.8	Sendmail	77
2.8.1	郵件伺服器	77
2.8.2	安裝 sendmail	77
2.8.3	sendmail 的設定檔	78
2.8.4	安裝 POP3 和 IMPA	79
2.8.5	建立虛擬郵件地址	81
Ch3	Windows 系列之安裝設定	85
3.1	IIS (Internet Information Server)	85
3.1.1	簡介	85
3.1.2	安裝	86
3.2	DNS (Domain Name System)	89
3.2.1	簡介	89
3.2.2	原理	89
3.2.3	安裝	91
3.3	Web site	94
3.3.1	簡介:	94
3.3.2	安裝	94
3.4	DHCP (Dynamic Host Confirguation Protocol)	98
3.4.1	簡介	98
3.4.2	安裝	99
3.5	SMTP(Simple Mail Transfer Protocol)	102

3.5.1 簡介	102
3.5.2 安裝	103
3.6.FTP (File Transfer Protocol).....	106
3.6.1 簡介	106
3.6.2 安裝	106
3.7 NNTP (Network News Transport Protocol).....	109
3.7.1 簡介	109
3.7.2 安裝	110
3.8 SQL Server 2000.....	114
3.8.1 簡介:	114
3.8.1.1 資料庫 (Database) :	114
3.8.1.2 SQL 語言介紹 :	116
3.8.1.3 SQL Server 2000 簡介 :	119
3.8.2 安裝:	121
3.9 憑證伺服器與 SSL	128
3.9.1 名詞簡介 :	128
3.9.1.1 憑證:.....	128
3.9.1.2 憑證授權單位:	128
3.9.1.3 伺服器憑證 :	128
3.9.1.4 SSL :	128
3.9.2 憑證伺服器的安裝 :	129
3.9.3 取得伺服器憑證的兩種方法 :	132
3.9.4 建立 Web 伺服器的 SSL 安全機制步驟 :	132
3.10 索引伺服器.....	133
3.10.1 索引服務的基本概念 :	133
3.10.2 索引伺服器簡介 :	134
3.10.3 索引伺服器安裝 :	134
3.11 ASP.....	135
3.11.1 ASP 介紹 :	135
3.11.2 ASP 用法簡介	135
附錄 A IIS 安全設計概要.....	141
附錄 B : 保護你的 Windows NT/2000 機器避免網路攻擊.....	148
附錄 C ASP 常用函式.....	153

課程名稱：電子商業伺服器管理

主題：電子商業伺服器管理

編撰者：陳世偉、林世唐、林志龍、謝逢鳴

參考文獻：

1. 陳雅秀/精通 Linux 網路伺服器架設實務/松崗研發中心
2. 廖文良/易學易用 RedHat Linux 8/碁峰資訊
3. 施威銘研究室/Linux Mandrake 9.1 實務應用/旗標出版
4. 陳庚暉/輕鬆掌握 Linux8.2 網路架設與管理/文魁資訊
5. 施威銘研究室/RedHat Linux 8 實務應用/旗標出版
6. 鳥哥的 Linux 與 Adsl 文件 ,網址 : <http://linux.vbird.org/>
7. 永遠的 UNIX ,網址 : http://www.fanqiang.com/index_b.shtml
8. Apache 2.0 文件,網址:<http://httpd.apache.org/docs-2.0/install.html>
9. Mysql 技術手冊,
網址 : <http://www.mysql.com/documentation/mysql/bychapter/index.html>
10. sendmail 線上 FAQ ,網址 : <http://www.sendmail.org/faq/>
11. Proftpd 文件,網址 : <http://proftpd.linux.co.uk/docs/>
12. 阿棟 BIND DNS 安裝及設定(一),網址 : <http://www.linuxcenter.com.tw/>
13. Nicolai Langfeldt HOWTO : DNS
網址 :
<http://www.linux.org.tw/CLDP/HOWTO/network/DNS-HOWTO/index.html>
14. 施威銘 研究室/RedHat Linux 7 架站實務/旗標出版
15. LINUXER 零客情報/天充
16. [Chinese Linux Documentation Project](http://www.linux.org.tw/CLDP/)
網址 : <http://www.linux.org.tw/CLDP/>
17. study-area,網址 : <http://www.study-area.org/menu1.htm>
18. Linux 技術網站,網址 : <http://www.linuxuser.com.tw/>
19. Harry M. Server 原著,陳世訓、陳淳哲 譯 / Windows 2000 Server 建構徹底研究/旗標出版.
20. 莊彥澤/Windows 2000 Server 網域建置與安全維護/學貫.
21. 顏逸品 編著/Windows 2000 Server 企業網路建構實務-群組原則與安全規劃篇/博碩.
22. 陳峰棋策畫、資訊教育小組編著 / IIS 5 網站規劃與架設實務 / 知城數位科技公司
23. 劉裕真/IIS 5 網站資料庫/知城數位科技公司
24. 陳會安/ASP 3.0 與 IIS 4/5 網站架設徹底研究 / 旗標出版
25. 莫瑟/ASP 3.0 程式設計入門手冊/美商麥格羅·希爾國際公司
26. 章立民/MS SQL Server 7.0 資料庫技術資源工具手冊/碁峰資訊公司

教材編撰年度：92 年度。

Ch1. 電子商務伺服器管理

1.1 電子商務發展背景：

「網路」這個名詞其實很早就有了，不過真正開始快速發展是在 90 年代，90 年代由於電腦體積越來越小、價格越來越便宜，所以使得個人電腦越來越普及；1993 年美國宣佈建立國家資訊基礎設施 NII 後，全世界許多國家紛紛制定和建立本國的 NII，從而極大地推動了計算機網路技術的發展，使計算機網路進入了一個嶄新的階段。

電子商務其實也並非 90 年代的產物，追溯自民國 70 年起，台灣政府最初推動“產業自動化服務團十年計畫”及“產業自動化十年計畫”。及至民國 80 年以發展 EDI 為主，這可謂台灣早期的電子商務雛型。當時產業自動化策略是引進國外 EDI 觀念與技術，並開始推動商業、海空運通關、製造業、金融及環保等 EDI 計畫，致有所謂的通關自動化的關貿網路及金融銀行的 FEDI 網路，然而隨著 90 年代網際網路的快速成長，電子商務這個名詞也開始跟著快速興起。

隨著網際網路科技越來越進步，許多方便的網路管理工具和網頁製作軟體普及，目前許多企業都有建置屬於自己公司的網頁，許多人都已瞭解到電子商務將可帶給全企業更多的獲益，所以紛紛開始積極經營這塊領域，甚至將整個企業上下游的供應鍊整個連結起來，再配合如：企業資源規劃（ERP；Enterprise Resource Planning）、顧客關係管理（CRM；Customer Relationship Management）等技術，以便能迅速反應市場的變化，降低更多成本、獲得更大利益、增加更多的客戶。

就電子商務從交易主體來區分主要有 B2C 與 B2B 二類。台灣針對國內電子商務發展以此二類為主軸。在 NII 推動計畫中的網際網路商業應用計畫，基本上，涉及 B2C 部份，就是指消費性的電子商務，目前正隨著網際網路的發展，快速普及中，以協助日用品業、3C 產業、成衣、圖書、旅遊與金融證券等行業建立各式商業中心、虛擬商店或虛擬企業等為重點。B2C 電子商務主要作業模式包括提供售前與售後服務、銷售與付款等，其中線上信用卡、電子支票、電子現金與電子發票等金融支付系統將是未來發展的關鍵因素。

至於 B2B 電子商務是指企業間有關採購、生產、配銷等相關商務透過網路作業，這方面即是行政院於民國 88 年 6 月 3 日通過的“產業自動化及電子化推動方案”，以推動生產、倉儲、運送及銷管之自動化外，並帶動各產業導入供應鍊及需求鍊的電子化作業能力，以提升整體競爭力。另外還有一種 C2C 模式也有相當程度的市場，此種模式最常見的即是拍賣網站，如最近廣告相當多的 eBay 及雅虎奇摩的拍賣網站，都有相當多的人參與，也有相當大的交易量產生。在上面所講的三種電子商務模式中，交易金額最大的要算是 B2B 模式的電子商務，而未來在 B2B 電子商務方面有可能會發展成像「電子市集」模式（e-Marketplace），根據研究機構 IDC 的預估，全球 2000 年電子市集的市場總規模僅達 158 億美元，但到 2004 年，透過電子市集的交易金額將超過 1.2 兆美

元,幾乎是 2000 年時 76 倍的規模。電子交易市集佔總體 B2B 市場比例的預估值,也非常驚人。雖然也有幾份調查報告謹慎地將數字定在 30% 出頭,但也有許多知名研究機構如 IDC、佛瑞斯特 (Forrester) 研究中心、Keenan Vision 等,樂觀地認為電子市集交易量,將佔整個 B2B 電子商務的 50% 以上。顯然,電子市集已被視為能夠主導 B2B 未來的主要潮流,不過雖然電子市集有相當的遠景,不過期中陷阱也頗多,也已經有許多失敗例子出現,所以未來電子市集要如何發展,還需要小心的做觀察及應變。

1.2 本書架構：

隨著電子商務日益興盛,越來越多企業開始建構自己的專屬伺服器系統,以進入電子商務的領域,在伺服器系統的管理上效率、安全、功能健全、可靠性等都是需考慮的,而目前也有相關產業可以協助一企業作規劃及管理,但因為需考量整體安全性,故仍有許多企業是由自己 MIS 部門來負責,而目前市面上伺服器的系統有許多種,而整體來說以 Linux 及 MS Windows 2000 最為普遍,故本教案將針對這兩種市佔率最多的產品做介紹。

在本教案中有關 Linux 方面將對 DNS、DHCP、Proxy、Mail、Samba、Apache、MySQL、FTP 八項做介紹,而在 Windows 2000 方面則將對 IIS、DNS、Web Site、DHCP、SMTP、FTP、NNTP、MSSQL、憑證伺服器與 SSL、索引伺服器、ASP 等各項作介紹,最後由於 Windows 2000 本身設計上有一些安全上的漏洞,故將於附錄中介紹如何做修正,以增進在使用時的安全性,另外附錄 C 則是 ASP 的一些常用函式介紹。

Ch2 Linux 系列之安裝設定

2.1 DNS 伺服器

2.1.1 前言

DNS 就是領域名稱系統 (Domain Name System) , 主要有三種類型 , 分別為 : Master、Slave 及 Cache-Only。下面將對這三種類型稍作介紹 :

Master

本身含有領域名稱的設定檔 , 該設定檔存有網域中所有正解 / 反解的正確資訊 , 每一個網域只能擁有一台 Master 伺服器。

Slave

即 Master 的備用伺服器 , 相關資訊都是由 Master 伺服器傳送過來 , 因此 Master 與 Slave 擁有的資訊是相同的。雖然說是備用伺服器 , 不過 Master 與 Slave 伺服器通常都是並存的 , 同時服務所有的名稱服務需求。

Cache-Only

顧名思義 , Cache-Only 類型的伺服器主要是用來幫助向外查詢 , 並將查詢得來的結果 , 以快取的方式存於伺服器中 , 伺服器中的內容全部為向外查詢的結果。此類型伺服器通常是架設在防火牆上。

本章將介紹 DNS 伺服器的安裝與設定 , 至於工作原理已於「網路作業系統」一書中詳細介紹 , 在此不再多做詳述。

2.1.2 安裝

目前 Linux 已朝向圖形化介面的方向發展 , 縮短與一般使用者的距離。在安裝方面 , 也愈來愈圖形化、人性化 , 只需在系統安裝時勾選想要安裝的各伺服器選項 , 安裝程式便會自動地將相關的檔案安裝完成 , 不需再像以往一般費時的一個個下載、安裝。不過在這裡還是將安裝方式分成「手動安裝」和「由安裝程式勾選」來介紹。

2.1.2.1 手動安裝

Linux 上的 DNS 伺服器大部分都是使用 BIND (Berkeley Internet Name Domain) , 這個套件很容易從全名看出是由柏克萊大學所發展。目前最新的版本為 9.2.3 版 (October 23, 2003) , 我們使用的 RedHat 9 內建的是 9.2.1 的版本。

官方網頁：<http://www.isc.org/products/BIND/>

下載處：<ftp://ftp.isc.org/isc/bind9/9.2.3/bind-9.2.3.tar.gz>

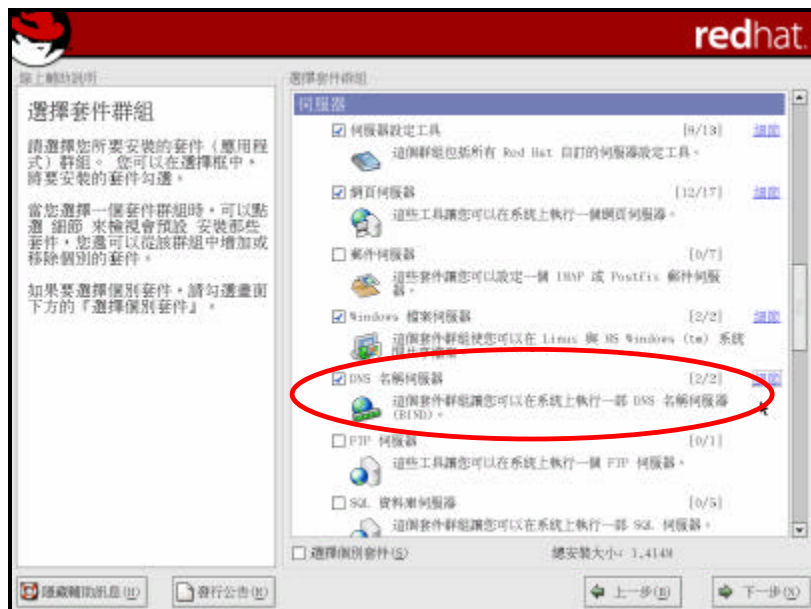
在 BIND 的官方網站僅提供 tar.gz 檔案格式的版本，請以下列步驟安裝：

```
[root@localhost root]# tar zxvf bind-9.2.3.tar.gz
[root@localhost root]# cd bind-9.2.3
[root@localhost bind-9.2.3]# ./configure
[root@localhost bind-9.2.3]# make
[root@localhost bind-9.2.3]# make install
```

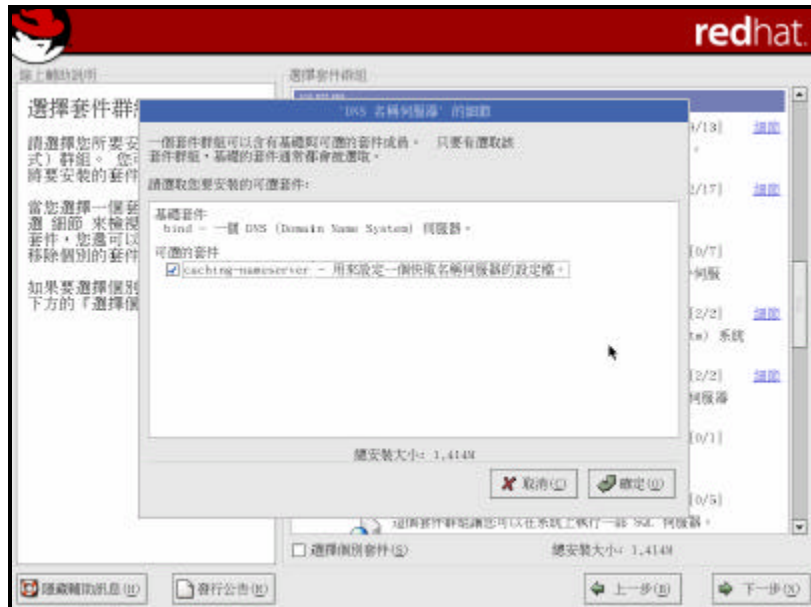
第一行指令會將 bind-9.2.3.tar.gz 這個檔案解壓縮到 bind-9.2.3 這個目錄(資料夾)中，第二行指令是切換到該目錄，第三行是在執行組態設定檔，第四行是編譯相關檔案，最後一行則將套件安裝到系統中。

2.1.2.2 由安裝程式勾選

在安裝 RedHat 9 (shrike) 的過程中，可以選擇想要安裝的套件，這種安裝方式比手動安裝簡單許多，不需要繁複的指令，只需要勾選想要安裝套件的選取方塊，安裝程式便會自動的將該套件加入安裝清單中，如果有其他相依的套件需要一併安裝，安裝程式也會詢問你是否要安裝(筆者建議安裝)。



勾選 DNS 名稱伺服器



DNS 名稱伺服器細節

2.1.3 啟動 DNS 伺服器

請先開啟並編輯一個新檔 `/etc/rc.d/init.d/named`，內容如下：

```

#!/bin/sh
#
# named This shell script takes care of starting and stopping
# named (BIND DNS server).
#
# Source function library.
. /etc/rc.d/init.d/functions
# Source networking configuration.
. /etc/sysconfig/network
# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0
[ -f /usr/local/sbin/named ] || exit 0
[ -f /etc/named.conf ] || exit 0
RETVAL=0
# See how we were called.
case "$1" in
start)
# Start daemons.
echo -n "Starting named: "
start-stop-daemon -S -n named -x /usr/local/sbin/named -- -u bind
RETVAL=$?
[ $RETVAL -eq 0 ] && touch /var/lock/subsys/named
echo
;;
stop)
# Stop daemons.
echo -n "Shutting down named: "
killproc named
RETVAL=$?
[ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/named
echo
;;
status)
/usr/local/sbin/rndc status
exit $?
;;
restart)
$0 stop
$0 start
;;
reload)
/usr/local/sbin/rndc reload
exit $?
;;
probe)
# named knows how to reload intelligently; we don't want linuxconf
# to offer to restart every time
/usr/local/sbin/rndc reload >/dev/null 2>&1 || echo start
exit 0
;;
*)
echo "Usage: named {start|stop|status|restart}"
exit 1
esac
exit $RETVAL

```

編輯好後，接下來在 /etc/rc.d/rc3.d 及 rc5.d 中新增啟動 named 的符號連結：

```
[root@localhost root]# cd /etc/rc.d/rc3.d
[root@localhost rc3.d]# ln -s /etc/rc.d/init.d/named S10named
[root@localhost rc3.d]# cd /etc/rc.d/rc5.d
[root@localhost rc5.d]# ln -s /etc/rc.d/init.d/named S10named
```

再來，在 /etc/rc.d/rc0.d 及 rc6.d 中新增結束 named 的符號連結：

```
[root@localhost rc5.d]# cd /etc/rc.d/rc0.d
[root@localhost rc0.d]# ln -s /etc/rc.d/init.d/named K90named
[root@localhost rc0.d]# cd /etc/rc.d/rc6.d
[root@localhost rc6.d]# ln -s /etc/rc.d/init.d/named K90named
```

上面得步驟都完成了後，便可以下述方式啟動 DNS 伺服器，當然，設定檔事先要先編輯好：

```
[root@localhost root]# /etc/rc.d/init.d/named start
```

重新啟動方式：

```
[root@localhost root]# /etc/rc.d/init.d/named restart
```

停止伺服器運作：

```
[root@localhost root]# /etc/rc.d/init.d/named stop
```

2.1.4 DNS 伺服器設定檔

DNS 伺服器有幾個重要的設定檔，分別為 named.conf、named.root 以及記錄檔(名稱可自訂)。

named.conf

指出該伺服器管轄的網域名稱及轄區檔檔案存放的位置及名稱。BIND 4 的版本使用 named.boot，可利用 named-bootconf.pl 這支 Script 將舊版的 named.boot 轉換成 named.conf，轉換方式為：

```
[root@localhost root]# named-bootconf.pl < named.boot > named.conf
```

以 Linux 的 HOWTO-DNS 作者 Nicolai Langfeldt 在其文章中舉的例子來看，原本 named.boot 內容如下：

```
directory /var/named
cache . root.hint
primary 0.0.127.IN-ADDR.ARPA
127.0.0.zone
primary localhost
localhost.zone
```

轉換後成為 named.conf 格式如下：

```
options {
    directory "/var/named" ;
};

zone "." {
    type hint;
    file "root.hint" ;
};

zone "0.0.127.IN-ADDR.APRA" {
    type master;
    fiel "127.0.0.zone" ;           //反解
};

zone "localhost" {
    type master;
    file "localhost.zone" ;       //正解
};
```

上面 named.conf 的內容相當簡單，option 裡描述的是 DNS 伺服器的記錄檔存放目錄為 /var/named/，因此，所有的 DNS zone file 都會儲存在此目錄下，後面在指定檔案時就不需給絕對路徑了。接下來都是 zone 的部份，第一段 zone 的描述是指定此 DNS 伺服器的根目錄檔案為 (/var/named/下的)root.hint 這個檔案(即為下段描述的 named.root)；第二個 zone 描述為定義 127.0.0 這段 IP zone 的反查，DNS 型態為 master，記錄檔名稱為 127.0.0.zone；第三個 zone 的描述是 localhost 正解的定義，DNS 型態為 master，記錄檔名稱為 localhost.zone。

named.root

列出各式根伺服器的所在位置。可以在
<ftp://rs.internic.net/domain/named.root> 下載到最新資訊的
named.root 檔，其內容如下：

```
; This file holds the information on root name servers needed to
; initialize cache of Internet domain name servers
; (e.g. reference this file in the "cache . <file>"
; configuration file of BIND domain name servers).
;
; This file is made available by InterNIC
; under anonymous FTP as
; file /domain/named.root
; on server FTP.INTERNIC.NET
;
; last update: Nov 5, 2002
; related version of root zone: 2002110501
;
;
; formerly NS.INTERNIC.NET
;
. 3600000 IN NS A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 3600000 A 198.41.0.4
;
; formerly NS1.ISI.EDU
;
. 3600000 NS B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET. 3600000 A 128.9.0.107
;
; formerly C.PSI.NET
;
. 3600000 NS C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET. 3600000 A 192.33.4.12
;
; formerly TERP.UMD.EDU
;
以下省略
```

記錄檔

紀錄檔又分兩種，一種是將主機名稱轉換成當地的位置，一般我們俗稱為 DNS 正解。另外一種是作反向網域查詢，該檔是將 IP 位址與主機名稱轉換的資料庫檔案，一般我們將這個動作稱為 DNS 反解。這兩種檔案的名稱都是在 named.conf 檔案中定義，當然，你可以自訂檔案名稱。

紀錄檔的部份，我想直接用例子來看是最快了，以下我們先介紹幾個紀錄檔中重要的項目：

TTL

Time to live，存放於 cache 中的資料之有效期限。

ORIGIN

即為 @ 符號，會自動的轉換成 named.conf 中指定的區域。

SOA

Start Of Authority，表示這個檔案是該區域的授權記錄檔。緊接在 SOA 後是指定這個區域的授權主機以及管理者信箱(由於@在這裡是個保留符號，所以改由 . 代替)。後面接著有五個數字，為 Serial、Refresh、Retry、Expire 以及 Minimun，他們分別代表此次更新資料的序號(可自訂形式)、進行更新的時間間隔、若更新失敗進行重試的時間間隔、紀錄的逾期時間以及預設的 TTL 值。

NS

用來指定特定領域所使用的名稱伺服器(Name Server)。

A

Address，定義 IP 位址。

MX

Mail Exchange，指定負責該領域的郵件伺服器名稱，數值愈低愈先做出回應。

CNAME

別名，同一個完整領域名稱可以同時擁有多個別名。

PTR

Pointer 的簡寫，在反解檔案中將 IP Address 指向所對應的領域名稱。

```

$TTL 14400
@ IN SOA
my.test.com.tw.
administrator.test.com.tw. (
1997022700 ; Serial
28800 ; Refresh
14400 ; Retry
3600000 ; Expire
86400 ) ; Minimum           //首先假設named.conf 中
                             //指定的IP 範圍為163.0.0
IN NS my.test.com.tw.
IN MX 10 my.test.com.tw.    // 此網域第一部郵件伺服器名稱
IN MX 20 m2.test.com.tw.   // 此網域第二部郵件伺服器名稱
my IN A 163.0.0.22         //my.test.com.tw 對應到IP 位址163.0.0.22
jc IN A 163.0.0.19        //jc.test.com.tw 對應到IP 位址163.0.0.19
www IN CNAME my.test.com.tw. // my.test.com.tw 有三個別名,
pop3 IN CNAME my.test.com.tw. // 分別為www、pop3 與smtp
smtp IN CNAME my.test.com.tw. //

```

正解檔範例

```

$TTL 14400
@ IN SOA
my.test.com.tw.
administrator.test.com.tw. (
1997022700 ; Serial
28800 ; Refresh
14400 ; Retry
3600000 ; Expire
86400 ) ; Minimum
IN NS my.test.com.tw.
22 IN PTR my.test.com.tw.    //IP 位址163.0.0.22 指向my.test.com.tw
19 IN PTR jc.test.com.tw.   //IP 位址163.0.0.19 指向jc.test.com.tw

```

反解檔範例

2.1.5 結語

DNS 伺服器不算容易架設，但對於使用 IP 位址的網路作業環境來說，是一個必備的伺服器，目前都透過 cache 的方式，加快網域名稱的查詢速度。本章所介紹的只是屬於簡單的網路環境下運作，若需要管理更大型的網路環境，則還需要考慮到備用伺服器等架構的規劃，可參考相關資料作進一步研究。

2.2 DHCP 伺服器

2.2.1 前言

DHCP (Dynamic Host Configuration Protocol)主要是動態的分配 IP Address 給用戶端，本章將介紹該伺服器的安裝與設定，至於工作原理已於「網路作業系統」一書中詳細介紹，在此不再多做詳述。

2.2.2 安裝

目前 Linux 已朝向圖形化介面的方向發展，縮短與一般使用者的距離。在安裝方面，也愈來愈圖形化、人性化，只需在系統安裝時勾選想要安裝的各伺服器選項，安裝程式便會自動地將相關的檔案安裝完成，不需再像以往一般費時的一個個下載、安裝。不過在這裡還是將安裝方式分成「手動安裝」和「由安裝程式勾選」來介紹。

2.2.2.1 手動安裝

我們使用的 RedHat 9 內建的是目前最新的 3.0 版本，在其官方網頁上可以下載到最新的版本。

官方網頁：<http://www.isc.org/products/DHCP/>

下載處：<ftp://ftp.isc.org/isc/dhcp/dhcp-latest.tar.gz>

下載完畢後，請依下列步驟安裝：

```
[root@localhost root]# tar zxvf dhcp-latest.tar.gz
[root@localhost root]# cd dhcp-3.0pl2
[root@localhost dhcp-3.0pl2]# ./configure
[root@localhost dhcp-3.0pl2]# make
[root@localhost dhcp-3.0pl2]# make install
```

第一行指令會將 dhcp-latest.tar.gz 這個檔案解壓縮到 dhcp-3.0pl2 這個目錄(資料夾)中，第二行指令是切換到該目錄，第三行是在執行組態設定檔，第四行是編譯相關檔案，最後一行則將套件安裝到系統中。

如果下載的是 rpm 的版本，則依下列步驟安裝：

```
[root@localhost root]# rpm -Uvh dhcp-latest.rpm
```

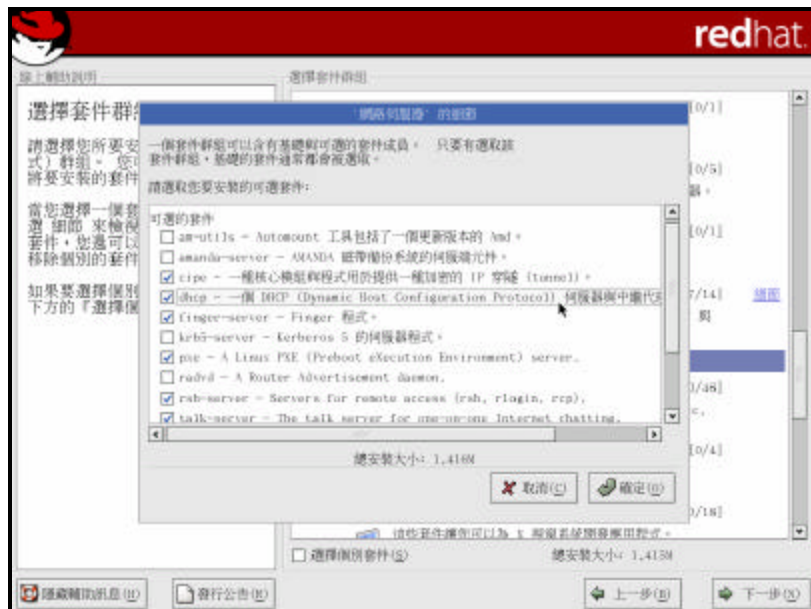
使用-Uvh 這樣的參數，如果你原先有安裝過 dhcp 伺服器，系統會幫你升級成新的版本。

2.2.2 由安裝程式勾選

在安裝 RedHat 9 (shrike) 的過程中，可以選擇想要安裝的套件，這種安裝方式比手動安裝簡單許多，不需要繁複的指令，只需要勾選想要安裝套件的選取方塊，安裝程式便會自動的將該套件加入安裝清單中，如果有其他相依的套件需要一併安裝，安裝程式也會詢問你是否要安裝(筆者建議安裝)。



點選網路伺服器的細節



勾選 dhcp 選項

2.2.3 啟動 DHCP 伺服器

啟動 DHCP 伺服器需要先設定 DHCP 的設定檔，如果沒有範本可參考，可以從解開的檔案夾 `dhcp-3.0p12/system` 中找到 `dhcpd.conf` 檔案，並將它 copy 到 `/etc` 目錄裡。接下來還需要開一個名為 `dhcpd.leases` 的新檔案 (`touch dhcpd.lease`)，這個檔案主要是用來記錄 IP 位址的租用情況 (類似 log file)，再將他 copy 到 `/var/lib/dhcp` 這個目錄下，就可以啟動 DHCP 伺服器了。由於 DHCP 伺服器一般都是在開機時便需要啟動，所以直接介紹如何使用開機時啟動的方式。

Red Hat Linux 本身有開發兩支程式：`ntsysv` 和 `tksysv`。前者是 shell 下的互動式選單模式，後者則有在 X Windows System 環境下的圖形介面。只需任執行其中一支程式，並勾選起 `dhcpd` 這個選項，下次系統啟動時便會自動的啟動 DHCP 伺服器。

2.2.4 DHCP 伺服器設定檔

DHCP 伺服器的設定檔為 `dhcpd.conf`，這邊列出幾個重要的參數說明如下，再配合設定檔範例說明各參數用法：

`default-lease-time`

設定 IP 出租的時間，單位為秒。

`domain-name`

設定網域名稱。

`domain-name-servers`

設定要提供服務 DNS Servers，多台時以空白隔開。

範例：

```
default-lease-time 259200;  
max-lease-time 777600;  
option domain-name "im.tku.edu.tw";
```

//預設 IP 租約時間 259200 秒

//最長 IP 租約時間 777600 秒

//網域名稱為 im.tku.edu.tw

`fixed-address`

設定固定的 IP 位址

hardware

指定某一台機器所使用的界面種類和硬體位址

host

主機區段，以{}為一區塊，{}中為該主機的細項設定。

範例：

```
host pc44{
    hardware ethernet 00:C1:79:E5:CD:A0;
    fixed-address 163.13.200.44;
    option broadcast-address 163.13.200.255;
    option routers 163.13.200.21;
    option domain-name-servers 163.13.1.180;
}
```

//主機 pc44 相關細項

//使用乙太網路，網路卡的 MAC address 為 00:C1:79:E5:CD:A0

//發配固定 IP 位址 163.13.200.44

//路由器的 IP 位址為 163.13.200.21

//DNS 伺服器 IP 位址為 163.13.1.180

max-lease-time

設定 IP 最長的出租時間，單位為秒。若 client 未要求，則採用 default-lease-time 中設定的時間，若有，則以不超過 max-lease-time 設定的時間為原則。

range

設定動態分配的 IP 範圍

routers

設定路由器的 IP

subnet

子網路區段，以{}為一區塊，{}中為該子網路的細項設定。

subnet-mask

設定子網路遮罩

範例：

```
subnet 163.13.200.0 netmask 255.255.255.0 {  
    range 163.13.200.11 163.13.200.40;  
    range 163.13.200.111 163.13.200.140;  
    option broadcast-address 163.13.200.255;  
    option routers 163.13.200.21;  
    option domain-name-servers 163.13.200.180;  
}
```

//子網路 163.13.200. 相關細項

//網路遮罩為 255.255.255.0

//發配動態 IP 位址的範圍有兩段，一個是 163.13.200.11~40，另一個是 163.13.200.111~140。

//路由器的 IP 位址為 163.13.200.21

//DNS 伺服器 IP 位址為 163.13.1.180

2.2.5 結語

DHCP 伺服器應該算是最容易架設的伺服器了，對於真實 IP 位址不足的網路作業環境來說，是一個相當需要的伺服器，管理上也不會有太大的問題，相當容易上手。

2.2.6 DHCP 伺服器設定檔範例：(取自 dhcp-3.0p12/system/dhcpd.conf)

```
# dhcpd.conf  
# Sample configuration file for ISC dhcpd  
# option definitions common to all supported networks...  
option domain-name "example.org";  
option domain-name-servers ns1.example.org, ns2.example.org;  
  
default-lease-time 600;  
max-lease-time 7200;  
  
# If this DHCP server is the official DHCP server for the local  
# network, the authoritative directive should be uncommented.  
#authoritative;  
  
# Use this to send dhcp log messages to a different log file (you also  
# have to hack syslog.conf to complete the redirection).
```

```

log-facility local7;

# No service will be given on this subnet, but declaring it helps the
# DHCP server to understand the network topology.

subnet 10.152.187.0 netmask 255.255.255.0 {
}

# This is a very basic subnet declaration.

subnet 10.254.239.0 netmask 255.255.255.224 {
    range 10.254.239.10 10.254.239.20;
    option routers rtr-239-0-1.example.org, rtr-239-0-2.example.org;
}

# This declaration allows BOOTP clients to get dynamic addresses,
# which we don't really recommend.

subnet 10.254.239.32 netmask 255.255.255.224 {
    range dynamic-bootp 10.254.239.40 10.254.239.60;
    option broadcast-address 10.254.239.31;
    option routers rtr-239-32-1.example.org;
}

# A slightly different configuration for an internal subnet.
subnet 10.5.5.0 netmask 255.255.255.224 {
    range 10.5.5.26 10.5.5.30;
    option domain-name-servers ns1.internal.example.org;
    option domain-name "internal.example.org";
    option routers 10.5.5.1;
    option broadcast-address 10.5.5.31;
    default-lease-time 600;
    max-lease-time 7200;
}

# Hosts which require special configuration options can be listed in
# host statements.  If no address is specified, the address will be
# allocated dynamically (if possible), but the host-specific information

```

```
# will still come from the host declaration.
```

```
host passacaglia {  
    hardware ethernet 0:0:c0:5d:bd:95;  
    filename "vmunix.passacaglia";  
    server-name "toccata.fugue.com";  
}
```

```
# Fixed IP addresses can also be specified for hosts.  These addresses  
# should not also be listed as being available for dynamic assignment.  
# Hosts for which fixed IP addresses have been specified can boot using  
# BOOTP or DHCP.  Hosts for which no fixed address is specified can only  
# be booted with DHCP, unless there is an address range on the subnet  
# to which a BOOTP client is connected which has the dynamic-bootp flag  
# set.
```

```
host fantasia {  
    hardware ethernet 08:00:07:26:c0:a5;  
    fixed-address fantasia.fugue.com;  
}
```

```
# You can declare a class of clients and then do address allocation  
# based on that.  The example below shows a case where all clients  
# in a certain class get addresses on the 10.17.224/24 subnet, and all  
# other clients get addresses on the 10.0.29/24 subnet.
```

```
class "foo" {  
    match if substring (option vendor-class-identifier, 0, 4) = "SUNW";  
}
```

```
shared-network 224-29 {  
    subnet 10.17.224.0 netmask 255.255.255.0 {  
        option routers rtr-224.example.org;  
    }  
    subnet 10.0.29.0 netmask 255.255.255.0 {  
        option routers rtr-29.example.org;  
    }  
    pool {  
        allow members of "foo";  
    }  
}
```



```
    range 10.17.224.10 10.17.224.250;
  }
  pool {
    deny members of "foo";
    range 10.0.29.10 10.0.29.230;
  }
}
```

2.3 SAMBA 伺服器

2.3.1 前言

SAMBA 是讓 Linux 與 Windows 間能共享資源的軟體。在 80 年代，微軟發展了 SMB (Server Message Block) 通訊協定，藉這 SMB 通訊協定，讓網路芳鄰的電腦主機間能夠分享其檔案、印表機等資源。後來便有熱心人士開發在 Linux 上的 SMB 通訊協定，演變成目前的 SAMBA。事實上，由於 SMB 通訊協定採用了 client/server 的架構，因此 SAMBA 也就分為 client (用戶)端與 server (伺服器)端兩部份，雖然本章標題僅為 SAMBA 伺服器，不過對於 SAMBA client 也會加以介紹。剛剛說到 SAMBA 分為 client (用戶)端與 server (伺服器)端兩部份，這兩個部份分別負責不同的工作。執行 SAMBA client，Linux 便可以透過 Windows 的網路芳鄰存取由執行 Windows 作業系統的電腦所共享出來的資源。執行 SAMBA server，則是讓執行 Windows 作業系統的電腦能夠使用由執行 Linux 的 SAMBA server 所分享的資源。

2.3.2 安裝

目前 Linux 已朝向圖形化介面的方向發展，縮短與一般使用者的距離。在安裝方面，也愈來愈圖形化、人性化，只需在系統安裝時勾選想要安裝的各伺服器選項，安裝程式便會自動地將相關的檔案安裝完成，不需再像以往一般費時的一個個下載、安裝。不過在這裡還是將安裝方式分成「手動安裝」和「由安裝程式勾選」來介紹。

2.3.2.1 手動安裝

我們使用的 RedHat 9 內建的是 2.2.7 的版本，而目前最新的版本為 3.0.1 rc1 版(December 4, 2003)，在其官方網頁上可以下載到最新的版本。

官方網頁：<http://www.samba.org/>

下載完畢後，請依下列步驟安裝：

```
[root@localhost root]# tar zxvf samba-latest.tar.gz
[root@localhost root]# cd samba-3.0.1
[root@localhost samba-3.0.1]# ./configure
[root@localhost samba-3.0.1]# make
[root@localhost samba-3.0.1]# make install
```

第一行指令會將 samba-latest.tar.gz 這個檔案解壓縮到 samba -3.0.1 這個目錄(資料夾)中，第二行指令是切換到該目錄，第三行是在執行組態設定檔，

第四行是編譯相關檔案，最後一行則將套件安裝到系統中。預設是安裝到 /usr/local/SAMBA 這個目錄。

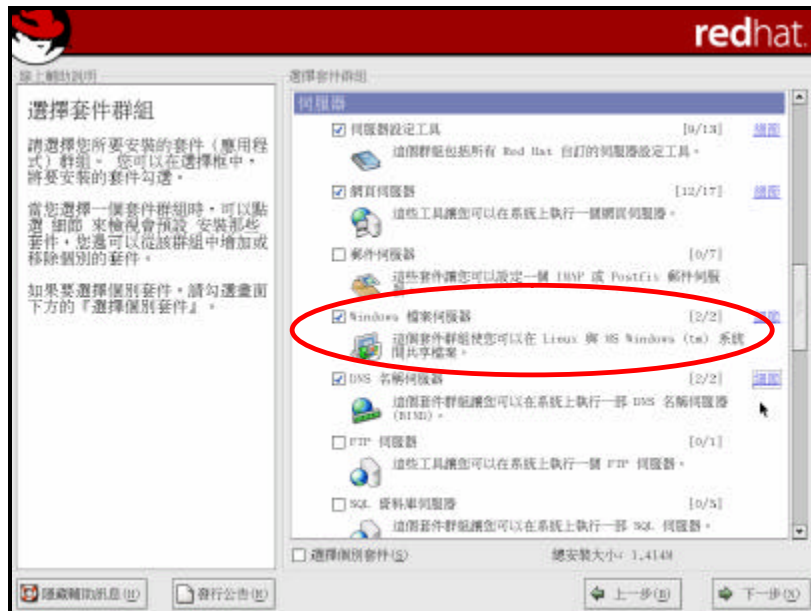
如果下載的是 rpm 的版本，則依下列步驟安裝：

```
[root@localhost root]# rpm -Uvh dhcp-latest.rpm
```

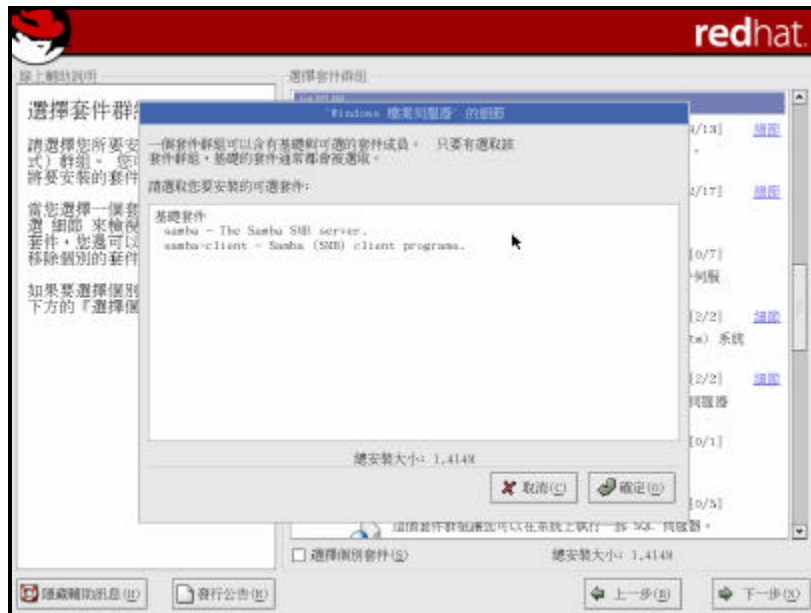
使用-Uvh 這樣的參數，如果你原先有安裝過 dhcp 伺服器，系統會幫你升級成新的版本。

2.3.2.2 由安裝程式勾選

在安裝 RedHat 9 (shrike) 的過程中，可以選擇想要安裝的套件，這種安裝方式比手動安裝簡單許多，不需要繁複的指令，只需要勾選想要安裝套件的選取方塊，安裝程式便會自動的將該套件加入安裝清單中，如果有其他相依的套件需要一併安裝，安裝程式也會詢問你是否要安裝(筆者建議安裝)。



勾選 Windows 檔案伺服器



Windows 檔案伺服器 (SAMBA) 細節

2.3.3 啟動 SAMBA 伺服器

啟動 SAMBA 伺服器需要先設定 SAMBA 的設定檔，如果是在安裝時勾選或安裝 RPM 的版本，那設定檔的位置預設是在 `/etc/samba/smb.conf`。如果是 `tar.gz` 的原始檔版本，並不會自動產生 `smb.conf` 這個設定檔，可以從解開的檔案夾 `smaba-3.0.1/packaging/RedHat` 中找到 `smb.conf` 檔案，並將它 copy 到 `/usr/local/samba/lib` 裡。接下來就是要啟動 SAMBA 伺服器了，如果是在安裝時勾選或安裝 RPM 的版本，執行步驟一，若是 `tar.gz` 的原始檔版本，則執行步驟二。

步驟一：

```
[root@localhost root]# /etc/rc.d/init.d/smb start
```

步驟二：

開啟並編輯 `/etc/rc.d/rc.local` 檔案，在檔案中加入下面兩行

```
/usr/local/samba/bin/smbd -D -d1
/usr/local/samba/bin/nmbd -D -d1 -n SAMBA server 的 netbiosname
```

2.3.4 開機時啟動

Red Hat Linux 本身有開發兩支程式：`ntsysv` 和 `tksysv`。前者是 shell 下的互動式選單模式，後者則有在 X Windows System 環境下的圖形介面。只需任

執行其中一支程式，並勾選起 smb 這個選項，下次系統啟動時便會自動的啟動 SAMBA 伺服器。

2.3.5 SAMBA 伺服器設定檔

SAMBA 伺服器的設定檔為 smb.conf，其中有相當多的參數。這邊僅介紹常用的幾個參數如下：

global 區段

netbios name

設定 SAMBA 伺服器的 NetBIOS 名稱。

netbios aliases

設定 SAMBA 伺服器的 NetBIOS 別名，如果想要同時使用多個別名，只需要將各別名以空白分開即可。

server string

設定 SAMBA 伺服器的說明文字。

workgroup

設定 SAMBA 伺服器的工作群組名稱。

admin users

設定 SAMBA 伺服器的管理者帳號名稱，管理者將擁有所有權利，預設為空。

encrypt passwords

設定 SAMBA 伺服器的使用者登入時，是否將登入密碼加密，預設值為 no，不過由於後期的 Windows 作業系統版本都採取以加密的方式傳送密碼，因此這邊建議改為 yes。

hosts allow

設定允許連線使用 SAMBA 伺服器的主機位址，預設為空(允許所有)。

範例：

```
hosts allow = 163.13.200. 163.13.201. EXCEPT 163.13.200.44
```

```
//除了 163.13.200.44 外，允許 163.13.200.* 和 163.13.201.* 的所有連線
```

```
hosts allow = im.tku.edu.tw
```

```
//允許來自 im.tku.edu.tw 這個 domain 的所有連線
```

hosts deny

與 hosts allow 相反，設定不允許連線使用 SAMBA 伺服器的主機位址，預設為空 (允許所有)。

範例：

hosts deny = 163.13.200.

//不允許來自 163.13.200.* 的所有連線

hosts deny = im.tku.edu.tw

//不允許來自 im.tku.edu.tw 這個 domain 的所有連線

invalid users

設定不允許登入 SAMBA 伺服器的使用者。

範例：

invalid users = vic, tau, @job

//不允許 vic、tau 及屬於 job 群組的使用者登入

root directory

設定 SAMBA 伺服器的根目錄，只要不在此目錄下的資源，即使有權限，任何人都無法使用。

security

設定 SAMBA 伺服器的安全等級，一共有 share、user、server 和 domain 四種安全等級，預設值為 user。

安全等級	
share	使用者不需帳號密碼即可登入。
user	由 SAMBA 伺服器檢查使用者帳號及密碼。
server	可將檢查使用者帳號及密碼的工作指定給另一台 SAMBA 或 Windows 伺服器負責。
domain	需指定 Windows 伺服器來進行使用者帳號及密碼驗證的工作。

vaild users

與 invalid users 相反，設定允許登入 SAMBA 伺服器的使用者。

log file

設定 SAMBA 伺服器的記錄檔位置。

max log size

設定 SAMBA 伺服器記錄檔的最大容量，預設為 5000，0 則表示無限制。

分享資源區段

comment

設定該項分享資源的說明。

path

分享出來的目錄之所在位置。若分享資源為印表機，則指定印表機佇列的位置

範例：

path = /home/vic

//分享/home/vic 目錄的資源時

path = /var/spool/

//分享印表機資源時

guest ok

設定 SAMBA 伺服器是否允許使用者不需帳號及密碼便能使用該項資源，預設值為 no。

read only

設定該項資源是否唯讀，預設值為 yes。

writable

與 read only 相反設定該項資源是否可寫入，預設值為 no。

print ok

是否允許使用者更改印表機佇列中的檔案，預設為 yes。

printer name

設定分享的印表機名稱。

available

設定該項資源是否啟用，預設為 yes。

browseable

設定使用者是否可看到該項資源，預設為 yes。若設定為 no，使用者將無法看到該項資源，不過可以直接輸入該資源的路徑，以存取該項資源。

2.3.6 結語

對於 SAMBA 伺服器的管理，其實有相當方便及人性化的工具—SWAT。SWAT 是一個 web-based 的 SAMBA 管理工具，可以在 SAMBA 的官方網站中找到，這個工具簡化了繁雜的使用者設定 SAMBA 設定檔手續，以親切的人機介面讓整個伺服器的設定更加容易，不彷試試。

2.3.7 SAMBA 伺服器設定檔範例：(取自 smaba-3.0.1/packaging/RedHat/smb.conf)

```
# This is the main Samba configuration file. You should read the
# smb.conf(5) manual page in order to understand the options listed
# here. Samba has a huge number of configurable options (perhaps too
# many!) most of which are not shown in this example
#
# Any line which starts with a ; (semi-colon) or a # (hash)
# is a comment and is ignored. In this example we will use a #
# for commentry and a ; for parts of the config file that you
# may wish to enable
#
# NOTE: Whenever you modify this file you should run the command "testparm"
# to check that you have not made any basic syntactic errors.
#
#===== Global Settings =====
[global]

# workgroup = NT-Domain-Name or Workgroup-Name
    workgroup = MYGROUP

# server string is the equivalent of the NT Description field
    server string = Samba Server

# This option is important for security. It allows you to restrict
# connections to machines which are on your local network. The
# following example restricts access to two C class networks and
# the "loopback" interface. For more examples of the syntax see
# the smb.conf man page
; hosts allow = 192.168.1. 192.168.2. 127.
```



```

# if you want to automatically load your printer list rather
# than setting them up individually then you'll need this
    printcap name = /etc/printcap
    load printers = yes

# It should not be necessary to spell out the print system type unless
# yours is non-standard. Currently supported print systems include:
# bsd, sysv, plp, lprng, aix, hpux, qnx
;    printing = bsd

# Uncomment this if you want a guest account, you must add this to /etc/passwd
# otherwise the user "nobody" is used
;    guest account = pcguest

# this tells Samba to use a separate log file for each machine
# that connects
    log file = /var/log/samba/log.%m

# Put a capping on the size of the log files (in Kb).
    max log size = 50

# Security mode. Most people will want user level security. See
# security_level.txt for details.
    security = user
# Use password server option only with security = server
;    password server = <NT-Server-Name>

# Password Level allows matching of _n_ characters of the password for
# all combinations of upper and lower case.
;    password level = 8
;    username level = 8

# You may wish to use password encryption. Please read
# ENCRYPTION.txt, Win95.txt and WinNT.txt in the Samba documentation.
# Do not enable this option unless you have read those documents
;    encrypt passwords = yes
;    smb passwd file = /etc/samba/smbpasswd

```

```

# The following are needed to allow password changing from Windows to
# update the Linux system password also.
# NOTE: Use these with 'encrypt passwords' and 'smb passwd file' above.
# NOTE2: You do NOT need these to allow workstations to change only
#         the encrypted SMB passwords. They allow the Unix password
#         to be kept in sync with the SMB password.
; unix password sync = Yes
; passwd program = /usr/bin/passwd %u
; passwd chat = *New*UNIX*password* %n\n *ReType*new*UNIX*password* %n\n
*passwd:*all*authentication*tokens*updated*successfully*

# Unix users can map to different SMB User names
; username map = /etc/samba/smbusers

# Using the following line enables you to customise your configuration
# on a per machine basis. The %m gets replaced with the netbios name
# of the machine that is connecting
; include = /etc/samba/smb.conf.%m

# Most people will find that this option gives better performance.
# See speed.txt and the manual pages for details
    socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192

# Configure Samba to use multiple interfaces
# If you have multiple network interfaces then you must list them
# here. See the man page for details.
; interfaces = 192.168.12.2/24 192.168.13.2/24

# Configure remote browse list synchronisation here
# request announcement to, or browse list sync from:
#   a specific host or from / to a whole subnet (see below)
; remote browse sync = 192.168.3.25 192.168.5.255
# Cause this host to announce itself to local subnets here
; remote announce = 192.168.1.255 192.168.2.44

# Browser Control Options:
# set local master to no if you don't want Samba to become a master
# browser on your network. Otherwise the normal election rules apply

```

```

; local master = no

# OS Level determines the precedence of this server in master browser
# elections. The default value should be reasonable
; os level = 33

# Domain Master specifies Samba to be the Domain Master Browser. This
# allows Samba to collate browse lists between subnets. Don't use this
# if you already have a Windows NT domain controller doing this job
; domain master = yes

# Preferred Master causes Samba to force a local browser election on startup
# and gives it a slightly higher chance of winning the election
; preferred master = yes

# Enable this if you want Samba to be a domain logon server for
# Windows95 workstations.
; domain logons = yes

# if you enable domain logons then you may want a per-machine or
# per user logon script
# run a specific logon batch file per workstation (machine)
; logon script = %m.bat
# run a specific logon batch file per username
; logon script = %U.bat

# Where to store roving profiles (only for Win95 and WinNT)
# %L substitutes for this servers netbios name, %U is username
# You must uncomment the [Profiles] share below
; logon path = \\%L\Profiles\%U

# All NetBIOS names must be resolved to IP Addresses
# 'Name Resolve Order' allows the named resolution mechanism to be specified
# the default order is "host lmhosts wins bcast". "host" means use the unix
# system gethostbyname() function call that will use either /etc/hosts OR
# DNS or NIS depending on the settings of /etc/host.config, /etc/nsswitch.conf
# and the /etc/resolv.conf file. "host" therefore is system configuration
# dependant. This parameter is most often of use to prevent DNS lookups

```

```

# in order to resolve NetBIOS names to IP Addresses. Use with care!
# The example below excludes use of name resolution for machines that are NOT
# on the local network segment
# - OR - are not deliberately to be known via lmhosts or via WINS.
; name resolve order = wins lmhosts bcast

# Windows Internet Name Serving Support Section:
# WINS Support - Tells the NMBD component of Samba to enable it's WINS Server
; wins support = yes

# WINS Server - Tells the NMBD components of Samba to be a WINS Client
# Note: Samba can be either a WINS Server, or a WINS Client, but NOT both
; wins server = w.x.y.z

# WINS Proxy - Tells Samba to answer name resolution queries on
# behalf of a non WINS capable client, for this to work there must be
# at least one WINS Server on the network. The default is NO.
; wins proxy = yes

# DNS Proxy - tells Samba whether or not to try to resolve NetBIOS names
# via DNS nslookups. The built-in default for versions 1.9.17 is yes,
# this has been changed in version 1.9.18 to no.
    dns proxy = no

# Case Preservation can be handy - system default is _no_
# NOTE: These can be set on a per share basis
; preserve case = no
; short preserve case = no
# Default case is normally upper case for all DOS files
; default case = lower
# Be very careful with case sensitivity - it can break things!
; case sensitive = no

#===== Share Definitions =====
[homes]
    comment = Home Directories
    browseable = no
    writable = yes

```

```

# Un-comment the following and create the netlogon directory for Domain Logons
; [netlogon]
;   comment = Network Logon Service
;   path = /home/netlogon
;   guest ok = yes
;   writable = no
;   share modes = no

# Un-comment the following to provide a specific roving profile share
# the default is to use the user's home directory
;[Profiles]
;   path = /home/profiles
;   browseable = no
;   guest ok = yes

# NOTE: If you have a BSD-style print system there is no need to
# specifically define each individual printer
[printers]
    comment = All Printers
    path = /var/spool/samba
    browseable = no
# Set public = yes to allow user 'guest account' to print
    guest ok = no
    writable = no
    printable = yes

# This one is useful for people to share files
;[tmp]
;   comment = Temporary file space
;   path = /tmp
;   read only = no
;   public = yes

# A publicly accessible directory, but read only, except for people in
# the "staff" group

```

```

;[public]
;  comment = Public Stuff
;  path = /home/samba
;  public = yes
;  read only = yes
;  write list = @staff

# Other examples.
#
# A private printer, usable only by fred. Spool data will be placed in fred's
# home directory. Note that fred must have write access to the spool directory,
# wherever it is.
;[fredsprn]
;  comment = Fred's Printer
;  valid users = fred
;  path = /homes/fred
;  printer = fred's_printer
;  public = no
;  writable = no
;  printable = yes

# A private directory, usable only by fred. Note that fred requires write
# access to the directory.
;[fredsdir]
;  comment = Fred's Service
;  path = /usr/somewhere/private
;  valid users = fred
;  public = no
;  writable = yes
;  printable = no

# a service which has a different directory for each machine that connects
# this allows you to tailor configurations to incoming machines. You could
# also use the %u option to tailor it by user name.
# The %m gets replaced with the machine name that is connecting.
;[pchome]
;  comment = PC Directories
;  path = /usr/pc/%m

```

```

; public = no
; writable = yes

# A publicly accessible directory, read/write to all users. Note that all files
# created in the directory by users will be owned by the default user, so
# any user with access can delete any other user's files. Obviously this
# directory must be writable by the default user. Another user could of course
# be specified, in which case all files would be owned by that user instead.
;[public]
; path = /usr/somewhere/else/public
; public = yes
; only guest = yes
; writable = yes
; printable = no

# The following two entries demonstrate how to share a directory so that two
# users can place files there that will be owned by the specific users. In this
# setup, the directory should be writable by both users and should have the
# sticky bit set on it to prevent abuse. Obviously this could be extended to
# as many users as required.
;[myshare]
; comment = Mary's and Fred's stuff
; path = /usr/somewhere/shared
; valid users = mary fred
; public = no
; writable = yes
; printable = no
; create mask = 0765

```

2.4 Proxy 伺服器

2.4.1 前言

在 Linux 中，最多人使用的 Proxy 伺服器，是 Squid 這個套件，本章將介紹該伺服器的安裝與設定，至於工作原理已於「網路作業系統」一書中詳細介紹，在此不再多做詳述。

2.4.2 安裝

目前 Linux 已朝向圖形化介面的方向發展，縮短與一般使用者的距離。在安裝方面，也愈來愈圖形化、人性化，只需在系統安裝時勾選想要安裝的各伺服器選項，安裝程式便會自動地將相關的檔案安裝完成，不需再像以往一般費時的一個個下載、安裝。不過在這裡還是將安裝方式分成「手動安裝」和「由安裝程式勾選」來介紹。

2.4.2.1 手動安裝

Linux 上的 Proxy 伺服器大部分都是使用 Squid，我們使用的 RedHat 9 內建的是 Squid-2.5.STABLE1-2 的版本，而目前最新的版本為 Squid-2.5.STABLE4 的版本(September 15, 2003)。

官方網頁：<http://www.squid-cache.org/>

下載處：

<http://www.squid-cache.org/Versions/v2/2.5/squid-2.5.STABLE4.tar.gz>

下載完畢後，請依下列步驟安裝：

```
[root@localhost root]# tar zxvf squid-2.5.STABLE4.tar.gz
[root@localhost root]# cd squid-2.5.STABLE4
[root@localhost squid-2.5.STABLE4]# ./configure
[root@localhost squid-2.5.STABLE4]# make
[root@localhost squid-2.5.STABLE4]# make install
```

第一行指令會將 squid-2.5.STABLE4.tar.gz 這個檔案解壓縮到 squid-2.5.STABLE4 這個目錄(資料夾)中，第二行指令是切換到該目錄，第三行是在執行組態設定檔，第四行是編譯相關檔案，最後一行則將套件安裝到系統中。

如果下載的是 rpm 的版本，則依下列步驟安裝：

```
[root@localhost root]# rpm -Uvh squid-latest.rpm
```

使用-Uvh 這樣的參數，如果你原先有安裝過 dhcp 伺服器，系統會幫你升級

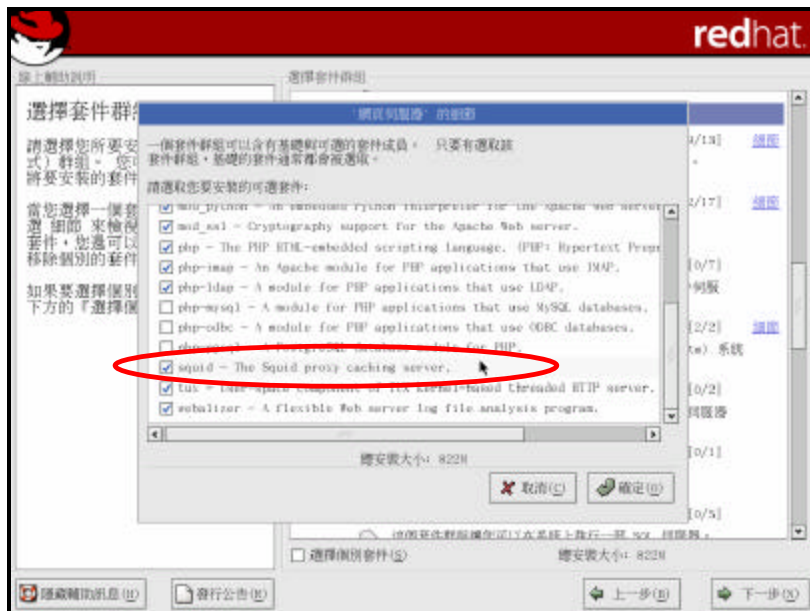
成新的版本。

2.4.2.2 由安裝程式勾選

在安裝 RedHat 9 (shrike) 的過程中，可以選擇想要安裝的套件，這種安裝方式比手動安裝簡單許多，不需要繁複的指令，只需要勾選想要安裝套件的選取方塊，安裝程式便會自動的將該套件加入安裝清單中，如果有其他相依的套件需要一併安裝，安裝程式也會詢問你是否要安裝(筆者建議安裝)。



點選網頁伺服器的細節



勾選 squid 選項

2.4.3 啟動 Proxy 伺服器

啟動 Proxy 伺服器需要先設定 Squid 的設定檔 Squid.conf，這個檔案會位在 /usr/local/squid/etc 中(如果安裝時採用預設路徑)。接下來請依以下步驟操作：

1. 以 root 的身分建立一個新的群組 nogroup。

```
[root@localhost root]# groupadd nogroup
```

2. 切換到 /usr/local 目錄下，改變目錄 squid 的擁有者。

```
[root@localhost root]# cd /usr/local  
[root@localhost local]# chown nobody.nogroup -R squid
```

3. 建立 squid 的 swap 目錄(即 cache 這個目錄)。

```
[root@localhost local]# cd /usr/local/squid/sbin  
[root@localhost sbin]# ./squid -z
```

經過上述三個步驟，在 /usr/local/squid/var 下會產生 cache 目錄，其中包含的目錄如以下的形式：/00/ /01/ /02/ ... /0F/，這些目錄每個包含 256 個子目錄，而最終就是緩衝的實際資料。

最後，啟動 Proxy 伺服器 squid 只要執行在 /usr/local/squid/bin 下的 RunCache 這個 Script：

```
[root@localhost sbin]# cd /usr/local/squid/bin  
[root@localhost bin]# ./RunCache
```

Red Hat Linux 本身有開發兩支程式：ntsysv 和 tksysv。前者是 shell 下的互動式選單模式，後者則有在 X Windows System 環境下的圖形介面。只需任執行其中一支程式，並勾選起 dhcpd 這個選項，下次系統啟動時便會自動的啟動 DHCP 伺服器(前提為安裝 RPM 的版本)。

2.4.4 Proxy 伺服器設定檔

Squid.conf 設定檔的參數相當的多，下面我們來看一看 squid.conf 檔的結構以及一些常用的選項：

squid.conf 配置檔的可以分為十三個部分，這十三個部分分別是：

1. NETWORK OPTIONS (相關的網路選項)
2. OPTIONS WHICH AFFECT THE NEIGHBOR SELECTION ALGORITHM
3. OPTIONS WHICH AFFECT THE CACHE SIZE (定義 cache 大小的有關選項)
4. LOGFILE PATHNAMES AND CACHE DIRECTORIES (定義日誌檔的路徑及 cache 的目錄)
5. OPTIONS FOR EXTERNAL SUPPORT PROGRAMS (外部支援程式選項)
6. OPTIONS FOR TUNING THE CACHE (調整 cache 的選項)
7. TIMEOUTS
8. ACCESS CONTROLS (存取控制)
9. ADMINISTRATIVE PARAMETERS (管理參數)
10. OPTIONS FOR THE CACHE REGISTRATION SERVICE (cache 註冊服務選項)
11. HTTPD-ACCELERATOR OPTIONS (httpd 加速選項)
12. MISCELLANEOUS (雜項)
13. DELAY POOL PARAMETERS

雖然 squid 的設定檔很龐大，但是如果你只是為一個中小型網路提供代理服務，並且只準備使用一台伺服器，那麼，你只需要修改配置檔中的幾個選項，所以在此僅將介紹幾個重要的參數：

http_access

根據 access controls list 允許或禁止某一類用戶使用。

範例：

```
http_access deny all
http_access allow all
```

http_port

squid 使用的通訊埠，預設為 3128。

icp_port

squid 用來傳送和接受 ICP 查詢的通訊埠，預設為 3130。

cache_peer

設定其他階層的 Proxy 伺服器，主要有 parent 與 sibling 兩種，parent 屬於上一層的 Proxy 伺服器，而 sibling 則是屬於同一階層的 Proxy 伺服器。

範例：

	cache_peer	主機名稱	型態	http_port	icp_port
如	cache_peer	p2.test.com.tw	parent	3128	3130
	cache_peer	sib.test.com.tw	sibling	3128	3130

cache_dir

設定 cache 目錄的路徑、大小。

範例：(預設)

```
cache_dir ufs /usr/local/squid/var/cache 100 16 256
```

指定 cache 路徑為 /usr/local/squid/var/cache，容量為 100 MB，第一層有 16 個目錄，第二層目錄有 256 個。

cache_mem

squid 使用的記憶體量，預設為 8 MB。

acl

access control list，

定義語法為：

```
acl aclname acltype string1 ...
```

```
acl aclname acltype "file" ...
```

acltype 可以是 src dst srcdomain dstdomain url_pattern urlpath_pattern time port proto method browser user 中的一種。

src 指定來源地址。可以用以下的方法指定：

```
acl aclname src ip-address/netmask ... (client 端 ip 位址)
```

```
acl aclname src addr1-addr2/netmask ... (範圍)
```

dst 指定目標位址

```
acl aclname dst ip-address/netmask ... (即客戶請求的伺服器的 ip 位址)
```

srcdomain 指定 client 端所屬的領域

```
acl aclname srcdomain foo.com
```

squid 將根據 client 端 ip 反向查詢 DNS。

dstdomain 指明請求伺服器所屬的領域

```
acl aclname dstdomain foo.com
```

由 client 端請求的 URL 決定。

time 指定訪問時間

```
acl aclname time [day-abbrevs] [h1:m1-h2:m2][hh:mm-hh:mm]
```

day-abbrevs:

S – Sunday

M – Monday

T – Tuesday

W – Wednesday

H – Thursday

F – Friday

A – Saturday

h1:m1 必須小於 h2:m2，表達式為[hh:mm-hh:mm]。

port 指定訪問埠

可以指定多個埠，比如：

```
acl aclname port 80 70 21 ...
```

```
acl aclname port 0-1024 ... (指定一個埠範圍)
```

proto 指定使用協定。

可以指定多個協議：

```
acl aclname proto HTTP FTP ...
```

method 指定請求方法。

```
acl aclname method GET POST ...
```

2.4.5 結語

Proxy 伺服器在大型的網路作業環境來說，是一個相當需要的伺服器。透過 Proxy 伺服器的幫助，不但可以縮短客戶端的回應時間，更可以減少對外的頻寬以及傳輸流量。

2.5 Apache

2.5.1 Apache Server

Internet 最近幾年成為極熱門的話題之一，造成這股熱潮的主因便是 World Wide Web(以下簡稱 WWW or Web)。世界各地的 Web Page 的製作者將自己精心製作的網頁放到網站上，讓所有其他的人參觀，透過這種互動，整個 Internet 形成一個龐大的資料庫，我們可以在上面找到各式各類我們想要的資訊。那麼如何建立一個網站呢？除了主機，作業系統與使用者所製作的網頁外，我們還需要安裝一套能將網頁放到網路上讓其它人來存取的軟體，也就是所謂的 Web Server。在 Linux 架設 WWW Server 一般都選擇 apache、cern 或 ncsa 的 httpd 這三個 WWW Serve 的軟體來安裝。

Apache 伺服器是目前最普遍使用到的，也是目前 Linux 系統中最廣泛被用到的網頁伺服器，幾乎在各版的 Linux 中有內附，在 Apache 的官方網頁 <http://www.apache.org> 可以下載到各作業系統的最新版本，在本章，我們會以手動的方式來逐步安裝。

2.5.2 Apache 編譯和安裝

1. 下載 Apache 套件：

相關的套件可以在 Apache 主頁 <http://www.apache.org/> 中下載到，也可以直接到下載頁面下載 <http://httpd.apache.org/download.cgi>，我們要用到是 Unix Source: httpd-2.0.XX.tar.gz 的這版本。

```
# lynx http://httpd.apache.org/download.cgi
```

下載後，必須進行認證的動作，來確認你下載到的是一個完整而不可修改的 Apache HTTP Server 版本。你可以利用 PGP 簽名來進行測試，第一步是從 Apache 的站台取得 KEYS 檔，然後將 ksyse 輸入到個人的 keys ring，請打下列指令：

```
# gpg < KEYS
```

或

```
# gpg --import KEYS
```

再下來的第二步是對 PGP 簽名做測試，PGP 的簽名檔可以在 Apache 的站台中找到，他的檔名是與該版本相同的並再後面加上 .asc 的字尾。你必須找到與你版本相同的檔案來做測試，請輸入以下命令：

```
# gpg httpd-2_0_XX.tar.gz.asc
```

或

```
# gpg --verify httpd-2_0_XX.tar.gz.asc
```

測試完成會得到以下的訊息：

```
Good signature from user "Martin Kraemer <martin@apache.org>".
```

2. 將 Apache 套件解壓縮：

將下載到的 Apache 套件解開壓縮：

```
# gzip -d httpd-2_0_NN.tar.gz
```

```
# tar xvf httpd-2_0_NN.tar
```

這樣子會在你現行的目錄下建立一個新的子目錄，並將檔案解開到該目錄下。你必須先用 `cd` 指令切換到你要的目錄。

3. 設定安裝組態：

接下來的步驟是對 Apache 的原始檔路徑和安裝電腦的平台進行組態的設定。有一個自動的程式 `configure` 程式可以幫我們處理，你可以在解開的根目錄下找到這個檔案：

如果只需使用預設的選項，則只要簡單的執行 `./configure`。

```
# ./configure --prefix=PREFIX
```

當 `configure` 執行時會花一些時間去分析安裝的系統，並建立 `Makefile` 檔，這檔案待會我們進行編譯時就會使用到。

4. 建立安裝檔並執行安裝：

現在我們可以開始建立安裝檔了，我們只需要使用 `make` 的指令，系統會花一些時間為我們編譯安裝檔，等編譯好之後，我們就可以用 `make install` 的指令，安裝程式就會依照 `configure` 的設定，將 Apache 的檔案放到指定的目錄下：

```
# make
```

```
# make install
```

5. 測試啟動和停止 Apache

一切安裝完成後，再還沒有針對 `httpd.conf` 檔案設定前，我們仍可以對測試 Apache HTTP server，請執行下列命令啟動：

```
# PREFIX/bin/apachectl start
```

啟動後請打開你的瀏覽器，並連結到 <http://localhost>，此時應該可以看到 DocumentRoot 目錄下設定的 `index.html` 首頁，這就表示 Apache 可以正常執行了。要停止 Apache 請執行底下命令：

```
# PREFIX/bin/apachectl stop
```

到此步驟結束，我們算是把 Apache 安裝上我們的工作台，接下來還需對 Apache 作一些設定的動作，才能正式啟動 Apache

2.5.3 編輯設定檔

安裝完 Apache 之後，還需要進行參數設定，要修改的是 `prefix/httpd.conf` 這個檔案，請用文字編輯器打開進行修改。

```
# vi PREFIX/conf/httpd.conf
```

`httpd.conf` 相關的語法，請參閱 Apache 的文件，在此僅介紹各參數所代表的意義。

1. 第一部分：全域環境設定

這一部分的變數定義是跟 Apache 操作時，整個會用到的。例如可連結的數量有幾個，組態檔的位置等。

```
ServerType standalone/inetd
```

定義 `httpd` 的啟動方式，`standalone` 為開機後立即執行，`inetd` 則需要時才臨時啟動。

```
LoadModule
```

載入使用到的模組，這部分可以不用去更動他，安裝 Apache 的過程會幫我們設定好。

```
ServerRoot /usr/local/etc/httpd
```

指定 server 的根目錄位置，以便 server 能找到組態設定，錯誤和日誌記錄檔

案。路徑最後面不要加上斜線。

Port 80

指定使用那一個 port 編號做為是 httpd 服務的開道,一般而言這個編號是設為 80 如果你要更改成其他值,比方說 Port 8000 則以後的 URL 就必須寫成為 <http://xx.xx.xx.xx:8000/>, 如果是使用 1203 以下的埠, 你需要從 root 下去啟動他。

ScoreBoardFile logs/apache_runtime_status

用來記錄 server 在執行程序時的資訊。預設是不指定的, 不指定時會被儲存在共用的記憶體區段中, 不能被利用。如果要指定, 要小心不要與另一個 Apache 共用到同一個記錄檔, 且此檔案要放在本地主機硬碟上

HostNameLookups on/off

設定要不要把啟動自動搜尋 HostName 的功能, 當本功能開啟時, 呼叫 httpd 服務時先去檢查 DNS 名稱跟 IP 位址是否符合, 所以在驅動時要特別注意 DNS 是不是運作正常以及網路是否暢通, 如果不正常的話, httpd 將不會被驅動。

PidFile logs/httpd.pid

要求 server 啟動時的程序識別碼記錄到該檔案中

BindAddress 193.13.128.0

強迫 Apache 只對某一個特定的位址回應, 可以用*號來表示所有的位址, 可以指定單一的 IP 或 Domain Name。

Timeout 300

設定接收和傳送逾時的時間

KeepAlive On

是否允許 TCP 連線續用, 設定 " Off " 表拒絕。

LogLevel debug/info/notice/warm/error/crit/alert/emerg

設定要錯誤訊息要記錄的等級

等級	敘述	範例
emerg	緊急事件 – 系統無法使用	"Child cannot open lock file. Exiting"

alert	直接回應的警告	"getpwuid: couldn't determine user name from uid"
crit	關鍵性的情況	"socket: Failed to get a socket, exiting child"
error	錯誤的情況	"Premature end of script headers"
warn	警告的情況	"child process 1234 did not exit, sending another SIGHUP"
notice	正常，但有意義的情況	"httpd: caught SIGBUS, attempting to dump core in ..."
info	資訊	"Server seems busy, (you may need to increase StartServers, or Min/MaxSpareServers)..."
debug	除錯時的訊息	"Opening config file ..."

CustomLog file/pipe

記錄所有有關於對 server 提出的請求。

MaxKeepAliveRequests 100

設定每一個連結預設的請求數，設定 0 時，則請求數可以無限制開放，建議設定一個高一點的數值，以達到最好的執行效力。

KeepAliveTimeout 15

設定來自客戶端相同的請求，所需的間隔時間。

Listen 12.34.56.78:80

用來指定不是來自於 port 設定的埠口進入之請求時，可接受的位址，這使該位址的進入變得很有彈性。

ExtendedStatus On

設定 Apache 在 server-status 處理器呼叫時，要產生完整的資訊，或者是基本的資訊。

2. 第二部分：主伺服器組態設定

這部分管理有關於主伺服器的設定，這部分也同時定義了虛擬主機的預設值。所有的指令都會在虛擬主機時再看到，而預設的部分，則會覆蓋主虛擬主機的定義。

```
User apache
```

```
Group apache
```

設定 apache 這程式的執行者，一般是使用 nobody，為了安全性的問題，最好是使用有權限的執行者。

```
ServerAdmin admin@your-domain.com
```

當伺服器產生錯誤相關的文件時，會顯示管理者的信箱，以方便使用者能夠向管理者回報問題的狀況。

```
ServerName localhost:80
```

給定伺服器一個名字和一個埠，雖然 Apache 可以透過 DNS 找出主機的名字，但可能因為 DNS 所傳回的並非真正伺服器的名稱，因此建議還是為該主機命名。如果你的主機沒有註冊 DNS 名稱，也可以輸入 IP 位址替代。

```
UseCanonicalName Off
```

此指令決定 Apache 是否要建立自我參照的 URL 和 SERVER_NAME 及 SERVER_PORT 變數，當設定為 Off 時，Apache 會用主機名稱和埠提供給客戶端，當設定為 On 時，Apache 會使用 ServerName 指令的值。

```
DocumentRoot "C:/Apache2/htdocs"
```

設定你網頁文件的存放位置，預設所有請求都會被帶到這個目錄下，但用符號連結和別名可以指引到別的地方去。

```
UserDir public-html
```

允許本伺服器的每個使用者可在自己的 home 目錄下建立一個名為 public-html 的目錄來存放個人網。此設定項目的預設值是 Disable，關閉不使用的。

```
<Directory /usr/local/apache/htdocs> </Directory>
```

<Directory> 和 </Directory> 是用來表示包住一個群組的指令，這群令可以用在指定的目錄和其下的所有子目錄。目錄的路徑可以使用完整的路徑，也可以使用正規式來表示所有符合的路徑。例如：<Directory ~
"~/www/.*/[0-9]{3}"> 會用來找尋所有 /www 下包含三個數字的目錄。這個標籤中又有五個指令可使用，如下表所示：

名稱	代表意義
Option	設定區塊功能，在後面如加上 All 表示使用者可在此目錄中進行任何存取，反之若設定為 None 則不允許使用者存取此目錄。加上 FollowSymLinks，表示 Apache Server 利用 Symbolic link 連結此目錄的檔案或目錄。
AllowOverride	設定是否改變先前使用者建立的 .htaccess 檔案來改變權限。若加上參數 None 表示不改變先權限設定；為 All 則以該目錄下的 .htaccess 檔案內容為新的權限設定，系統所有目錄預設為 None
deny	拒絕從指定的 IP 位址連結到此目錄
allow	允許從指定的 IP 位址連結到此目錄
order	當 deny 與 allow 設定衝突時，設定以何者為優先

```
<IfModule test>...</IfModule>
```

IfModule 的區段是用來包含指令模組的條件，當 IfModule 所指定的模組是存在時，所包含的指令群才有意義，如果是不存在的模組，則包含的指令則無效的。

模組的名稱可以使用 not 的語法。例如：

- ModuleName
- !ModuleName

```
DirectoryIndex index.html index.html.var
```

設定網頁目錄中預設開啟網頁的檔名，愈前面者愈優先開啟。

```
AccessFileName .htaccess
```

設定系統保護目錄設定檔的檔名，系統預設為 .htaccess。

```
TypeConfig /etc/mime.types
```

設定 "mime.types" 檔案儲存位置。mime.types 檔案中記載各種檔類型關聯的位置，當使用者開啟某類型的檔案時，瀏覽器會自動呼叫相關的軟體來編輯或執行此檔案。

```
Default Type text/plain
```

設定 Server 當遇到不能辨識的檔案類型，以一般文字檔案來處理。

Hostname Lookups Off

設定客戶端連接至本網站時記錄其主機名稱，若設為 On 記錄網域名稱，如 www.apache.org；若設為 Off 則記錄 IP 位址，如 204.62.129.132。

ErrorLog /var/log/httpd/error_log

指定記錄 Apache HTTP server 執行期間的錯誤訊息

ServerSignature On

設定是否於 server 自行產生的網頁(如錯誤訊息、FTP 目錄)中加入相關的訊息。若設為 On 則加上 server 的版本與主機名稱，Off 則都不加，email 則除了版本與主機名稱外還加上 server 管理者的 email 帳號。

Alias /icons/ "var/www/icons"

將敘述較長的目錄名稱及路徑以較簡易的別名來替代。

ScriptAlias /cgi-bin/ "/var/www/cgi-bin"

將 script 程式的路徑及目錄設成一個簡易的別名替代。

CacheRoot "/etc/httpd/proxy"

設定 Cache 的位置，後面還 Cache 的相關設定，系統預設 Cache 為不使用。

CacheSize 5

設定 Cache 的大小，單位為 KByte。通常如果系統負荷的了，我們會將 Cache 設的稍大，以求得較佳的效能表現。

CacheGcInterval 4

設定檢 Cache Size 的週期，單位為小時。

CacheMaxExpire 24

設定一檔案於 Cache 中最多的停留時間，系統預設為 24 小時。

3. 第三部分：虛擬主機設定

NameVirtualHost 12.34.56.78:80

NameVirtualHost 12.34.56.78

設定虛擬的 IP 位址。讀者可使用不同 IP 位址，或者相同的 IP 位址不同的通訊埠號(Port)來建立每一個虛擬主機。而每一個虛擬主機都必須跟隨著下列的

一組相關設定。

```
<VirtualHost *:80>
```

輸入可連上 Internet 的真實 IP 位址

```
ServerAdmin webmaster@dummy-host.example.com
```

設定 WebServer 的管理者 Email

```
DocumentRoot /www/docs/dummy-host.example.com
```

設定網頁存放的根目錄位置

```
ServerName dummy-host.example.com
```

設定虛擬主機的名稱

```
ErrorLog logs/dummy-host.example.com-error_log
```

設定錯誤訊息記錄檔的位置

```
CustomLog logs/dummy-host.example.com-access_log common
```

設定一般訊息記錄檔的位置

```
</VirtualHost>
```

虛擬主機設定結束

2.6 Mysql

2.6.1 Mysql 簡介

MySQL 是一個真正的多用戶、多線程 SQL 數據庫服務器。SQL (結構化查詢語言) 是世界上最流行的和標準化的數據庫語言。MySQL 是以一個客戶機/服務器結構的實現，它由一個服務器守護程序 `mysqld` 和很多不同的客戶程序和庫組成。

SQL 是一種標準化的語言，它使得存儲、更新和存取信息更容易。例如，你能用 SQL 語言為一個網站檢索產品信息及存儲顧客信息，同時 MySQL 也足夠快和靈活以允許你存儲記錄文件和圖像。

MySQL 主要目標是快速、健壯和易用。最初是因為我們需要這樣一個 SQL 服務器，它能處理與任何可不昂貴硬件平台上提供數據庫的廠家在一個數量級上的大型數據庫，但速度更快，MySQL 就開發出來。自 1996 年以來，我們一直都在使用 MySQL，其環境有超過 40 個數據庫，包含 10,000 個表，其中 500 多個表超過 7 百萬行，這大約有 100 個吉字節(GB)的關鍵應用數據。

MySQL 建立的基礎是業已用在高要求的生產環境多年的一套實用例程。儘管 MySQL 仍在開發中，但它已經提供一個豐富和極其有用的功能集。

MySQL 的官方發音是 “My Ess Que Ell” (不是 MY-SEQUEL)

MySQL 一些重要的特性：

- 使用核心線性的完全多線性。這意味著它能很容易地利用多 CPU (如果有)。
- C、C++、Eiffel、Java、Perl、PHP、Python、和 TCL API。見 20 MySQL 用戶端工具和 API。
- 可運行在不同的平台上，見 4.2 MySQL 支援的操作系統。
- 多種列類型：1、2、3、4、和 8 字節長度的有符號/無符號整數、FLOAT、DOUBLE、CHAR、VARCHAR、TEXT、BLOB、DATE、TIME、DATETIME、TIMESTAMP、YEAR、SET 和 ENUM 類型。
- 利用一個最佳化的一遍掃描多重聯結 (one-sweep multi-join) 非常快速地進行聯結 (join)。
- 在查詢的 SELECT 和 WHERE 部分支援全部運算子和函數，例如：
`mysql> SELECT CONCAT(first_name, " ", last_name) FROM tbl_name WHERE income/dependents > 10000 AND age > 30;`
- 通過一個高度最佳化的類程式庫實現 SQL 函數庫並且像他們能達到的一樣快速，通常在查詢初始化後不應該有任何內存分配。
- 全面支援 SQL 的 GROUP BY 和 ORDER BY 子句，支援聚合函數

- (COUNT(), COUNT(DISTINCT), AVG(), STD(), SUM(), MAX() 和 MIN())。
- 支援 ANSI SQL 的 LEFT OUTER JOIN 和 ODBC 語法。
 - 你可以在同一查詢中混用來自不同資料庫的表。
 - 一個非常靈活且安全的權限和密碼系統，並且它允許根據主機的認證。密碼是安全的，因為當與一個服務器連接時，所有的密碼傳送被加密。
 - ODBC for Win32。所有的 ODBC 2.5 函數和其他許多函數。例如，你可以用 Access 連接你的 MySQL 服務器。
 - 具備索引壓縮的快速 B 樹表。
 - 每個表允許有 16 個索引。每個索引可以由 1 到 16 個列或列的一部分組成。最大索引長度是 256 個字節(在編譯 MySQL 時，它可以改變)。
 - 一個索引可以使用一個 CHAR 或 VARCHAR 字段的前碼。
 - 固定長度和可變長度記錄。
 - 用作臨時表的內存散列表。
 - 大資料庫處理。我們正在對某些包含 50,000,000 個記錄的資料庫使用 MySQL。
 - 所有列都有缺省值，你可以用 INSERT 插入一個表列的子集，那些沒用明確給定值的列設置為他們的缺省值。
 - 為了可移植性使用 GNU Automake , Autoconf 和 libtool。
 - 用 C 和 C++ 編寫，並用大量不同的編譯器測試。
 - 一個非常快速的線性的記憶體分配系統。
 - 沒有內存漏洞。用一個商用內存漏洞監測程序測試過 (purify)。
 - 包括 myisamchk，一個檢查、最佳化和修復資料庫表的快速實用程序。
 - 全面支援 ISO-8859-1 Latin1 字符集。例如，斯堪的納維亞的字符 a, a and o 在表和列名字被允許。
 - 所有資料以 ISO-8859-1 Latin1 格式保存。所有正常的字符串比較是忽略大小寫的。
 - 根據 ISO-8859-1 Latin1 字符集進行排序 (目前瑞典語的方式)。通過在源程式中增加排序順序數組可以改變它。為了理解一個更高級的排序例子，看一看捷克語的排序程式。MySQL 支援可在編譯時指定的很多不同的字符集。
 - 表和列的別名符合 SQL92 標準。
 - DELETE、INSERT、REPLACE 和 UPDATE 返回有多少行被改變(受影響)。
 - 函數名不會與表或列名衝突。
 - 服務器能為用戶端提供多種語言的出錯消息。
 - 用戶端端使用 TCP/IP 連接或 Unix 套接字 (socket) 或 NT 下的命名管道連接 MySQL。
 - MySQL 特有的 SHOW 命令可用來檢索資料庫、表和索引的信息，EXPLAIN 命令可用來確定最佳化器如何解決一個查詢。

2.6.2 Mysql 安裝

1. 下載 Mysql :

目前最新最穩定的 Mysql 版本是 4.0 版，可以在 Mysql 的主頁

<http://www.mysql.com/> 中下載到，或直接到 Mysql 的下載頁面

<http://www.mysql.com/downloads/index.html> 下載，我們將使用原始碼來安裝，請下載 `mysql-standard-4.0.XX-pc-linux-i686.tar.gz` 的版本。

2. 快速的安裝

請依照下面的命令進行安裝你的 Mysql

1. 移動到你要安裝的目錄下，並自 Mysql 站台取得 Mysql 的原始程式碼壓縮檔。

2. 新增一個群組和使用者，以獨立給 Mysql 使用執行

```
# groupadd mysql
```

```
# useradd -g mysql mysql
```

3. 將壓縮檔在現行的目錄下解開：

```
# gunzip < mysql-VERSION.tar.gz | tar -xvf -
```

這時會在現行目錄下產生一個 `mysql-VERSION` 的目錄

4. 切換進去 `mysql-VERSION` 的目錄

```
# cd mysql-VERSION
```

5. 利用 `configure` 設定安裝的環境，並使用 `make` 建立安裝檔

```
# ./configure --prefix=/usr/local/mysql
```

```
# make
```

如果想知道 `configure` 有那些參數可以使用，可以執行 `./configure --help` 指令，一般來說，使用預設的服務就可以了。

6. 進行安裝

```
# make install
```

請使用 `root` 的權限，安裝過程會花上一些時間，請等待程式執行結束。

7. 如果之前並未安裝過 Mysql，則需要建立一份 `MYSQLI` 的權限表

```
# scripts/mysql_install_db
```

8. 改變 Mysql 的所有權限

```
# chown -R root /usr/local/mysql
# chown -R mysql /usr/local/mysql/var
# chgrp -R mysql /usr/local/mysql
```

第一步是將所有檔案的所有者設為 root，第二步是將有關於資料的目錄所有者設為 mysql，第三步則是將所有的群組設為 mysql

9. 自動啟動 mysql

如果想要 Mysql 自動啟動，可以複製 support-files/mysql.server 到系統的啟動檔裡。

10. 初始化並測試 Mysql

當全部裝設完畢，可以執行下列命令來測試你的 Mysql

```
# /usr/local/mysql/bin/mysqld_safe --user=mysql
```

2.6.3 Mysql 簡單操作

接下來這章我們對 mysql 的操作做一個簡單的介紹，我們可以利用 mysql 這隻程式來對 Mysql 資料庫進行一些監控的動作，mysql 可讓你連結資料庫、執行 queries 和觀看傳回的結果，利用 mysql 可以方便與資料庫進行互動。如果想知道 mysql 有那些功能可以使用，可以執行：

```
# mysql --help
```

1. 連結資料庫和結束連結

連結資料庫時，必須提供資料庫使用者的帳號和密碼，如果你登入的主機和資料庫的主機不同時，你也須要指定料庫的主機名稱，在 mysql 後面接 -h 指定主機，-u 指定使用者帳號，-p 表示輸入密碼：

```
# mysql -h host -u user -p
```

```
Enter password: *****
```

在 Enter password 字下提示你輸入密碼，打入的密碼會以 *** 遮蓋。成功登入後，會看到以下的訊息畫面：

```
# mysql -h host -u user -p
```

```
Enter password: *****
```

```
Welcome to the MySQL monitor. Commands end with ; or \g.
```

```
Your MySQL connection id is 25338 to server version: 4.0.14-log
```

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

```
mysql>
```

mysql>的提示字元表示你可以在此輸入 mysql 的命令。

有些資料庫允許使用 anonymous 的使用者，則你可不用輸入任何參數進入

```
#mysql
```

當連結進資料庫後，你可以在 mysql>提示下隨時結束連結資，只要輸入下列的命令

```
mysql>QUIT
```

```
Bye
```

2. 對資料庫進行 QUERIES

請先連結上資料庫，我們在 mysql>提示列下進行輸入簡單的命令，來查尋 Mysql 的版本和現在日期：

```
mysql> SELECT VERSION(), CURRENT_DATE;
```

```
+-----+-----+
| VERSION() | CURRENT_DATE |
+-----+-----+
| 3.22.20a-log | 1999-03-19 |
+-----+-----+
```

```
1 row in set (0.01 sec)
```

```
mysql>
```

在這指令說明下，我們可以看幾個 mysql 特性：

- 每一個 SQL 組成的指令，都需以分號做結尾(有些指令是例外的，例如 QUIT 不需加分號)
- 當你下了命令之後，mysql 送到伺服器去處理，伺服器處理完後傳回結果，並重新回到 mysql 提示列等待下次的命令輸入
- mysql 會將結果用表格的方式展示，第一列為項目標籤，第二列之後為相對應的結果。
- 最後 mysql 會展示回傳了多少資料，和使用了多少時間

3. 建立並使用資料庫

我們可以先使用 SHOW 指令來觀看現在 server 中存在那些資料庫，請執行底下指令：

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| mysql   |
| test    |
+-----+
```

在所有的 server 上都會有 mysql 和 test 兩個資料庫。Mysql 內的是用來記錄 Mysql server 的相關設定用的，最好不要去動他。而 test 則是開放給使用者測試用的，別人也可以有更改刪除你在此資料庫資料的權限。

- 建立資料庫

如果你有建立資料庫的權限，你就可以使用 creat 指令建立一個資料庫

```
mysql> CREATE DATABASE menagerie;
```

建立之後，你就可以選擇使用他，請用 USE 的指令選取資料庫

```
mysql> USE menagerie
Database changed
```

當你每次要使用時，你都要指定使用的資料庫。你可以用 USE 的指令指定使用的資料庫，也可以在進入 mysql 時，除了輸入主機名稱，使用者帳號和密碼之外，也可以指令使用的資料庫

```
# mysql -h host -u user -p menagerie
Enter password: *****
```

- 建立資料表單

建立完成資料庫之後，仍然是空的，用 SHOW TABLES 指令觀看資料庫內的內容

```
mysql> SHOW TABLES;
Empty set (0.00 sec)
```

使用 CREATE TABLES 的指令來建令資料表單：

```
mysql> CREATE TABLE pet (name VARCHAR(20), owner VARCHAR(20),  
-> species VARCHAR(20), sex CHAR(1), birth DATE, death DATE);
```

現在你已經建立一個 pet 資料表單，你可以用 SHOW 來檢視

```
mysql> SHOW TABLES;  
+-----+  
| Tables in menagerie |  
+-----+  
| pet |  
+-----+
```

想檢察表單的欄位，可以使用 DESCRIBE 指令來觀看

```
mysql> DESCRIBE pet;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type          | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| name  | varchar(20)  | YES  |     | NULL    |      |  
| owner | varchar(20)  | YES  |     | NULL    |      |  
| species | varchar(20) | YES  |     | NULL    |      |  
| sex   | char(1)      | YES  |     | NULL    |      |  
| birth | date         | YES  |     | NULL    |      |  
| death | date         | YES  |     | NULL    |      |  
+-----+-----+-----+-----+-----+-----+
```

- 載入資料到資料表

資料集如果增加的數量不多，可以用 INSERT 的語法一次增加一筆資料。如果是大量的資料要載入，那也可以將資料寫在文字檔裡，再由 mysql 一次讀進來。

```
mysql> INSERT INTO pet  
-> VALUES  
( 'Puffball', 'Diane', 'hamster', 'f', '1999-03-30', NULL );
```

在此，日期是以文字的方式輸入，格式為 YYYY-MM-DD，亦可指定 NULL 來表

示日期的資料遺失。

使用檔案的方式，必須將每一筆資料寫在一行，用 tab 來分隔每個資料，並依欄位順序排列，對遺失的數值不再用 NULL 表示，而改以用 \N。假設我做好一個資料檔為 pet.txt，我們將此資料檔載入到資料庫中：

```
mysql> LOAD DATA LOCAL INFILE "pet.txt" INTO TABLE pet;
```

- 檢索資料表

資料載入了資料表內後，我們可以使用 SELECT 語法簡單的查看目前的資料

```
mysql> SELECT * FROM pet;
```

name	owner	species	sex	birth	death
Fluffy	Harold	cat	f	1993-02-04	NULL
Claws	Gwen	cat	m	1994-03-17	NULL
Buffy	Harold	dog	f	1989-05-13	NULL
Fang	Benny	dog	m	1990-08-27	NULL
Bowser	Diane	dog	m	1979-08-31	1995-07-29
Chirpy	Gwen	bird	f	1998-09-11	NULL
Whistler	Gwen	bird	NULL	1997-12-09	NULL
Slim	Benny	snake	m	1996-04-29	NULL
Puffball	Diane	hamster	f	1999-03-30	NULL

當資料發現有錯誤需要更新時，我們可以重作一分文字檔，先將所有的資料刪除，再重新讀入新的資料檔，例如我們發現 Bowser 的生日是 1989 年，我們重新更正資料：

```
mysql> DELETE FROM pet;
```

```
mysql> LOAD DATA LOCAL INFILE "pet.txt" INTO TABLE pet;
```

這樣的做法當資料大時很不實際，因此我們可以改用 UPDATE 語法來更正資料

```
mysql> UPDATE pet SET birth = "1989-08-31" WHERE name = "Bowser";
```

UPDATE 語句會針對條件修改符合條件的資料，而不需要重新載入全部的資料。

- 檢索特別的一行或一列

利用 SELECT 和 WHERE 設定條件，我們可直接選擇特定的幾筆資料出來

```
mysql> SELECT * FROM pet WHERE name = "Bowser";
```

name	owner	species	sex	birth	death
Bowser	Diane	dog	m	1989-08-31	1995-07-29

也可以給予範圍的條件，例如選取生日在 1998 年之前的

```
mysql> SELECT * FROM pet WHERE birth >= "1998-1-1";
```

name	owner	species	sex	birth	death
Chirpy	Gwen	bird	f	1998-09-11	NULL
Puffball	Diane	hamster	f	1999-03-30	NULL

或者是使用複合條件，條件之間需用 AND 或 OR 來合併，也可以同時使用

```
mysql> SELECT * FROM pet WHERE (species = "cat" AND sex = "m")
-> OR (species = "dog" AND sex = "f");
```

name	owner	species	sex	birth	death
Claws	Gwen	cat	m	1994-03-17	NULL
Buffy	Harold	dog	f	1989-05-13	NULL

如果想讀取某些部分的欄位名稱，可以在 FROM 給定欄位名

```
mysql> SELECT name, birth FROM pet;
```

name	birth
------	-------

name	birth
Fluffy	1993-02-04
Claws	1994-03-17
Buffy	1989-05-13
Fang	1990-08-27
Bowser	1989-08-31
Chirpy	1998-09-11
Whistler	1997-12-09
Slim	1996-04-29
Puffball	1999-03-30

結合 FROM 和 WHERE 的條件，我們可以輕易地檢索到我們想要的資訊

```
mysql> SELECT name, species, birth FROM pet
-> WHERE species = "dog" OR species = "cat";
```

name	species	birth
Fluffy	cat	1993-02-04
Claws	cat	1994-03-17
Buffy	dog	1989-05-13
Fang	dog	1990-08-27
Bowser	dog	1989-08-31

2.6.4 Mysql 權限控制

權限系統控制關於一個使用帳號連線到主機後，可以對某資料庫做什麼動作？這些動作包括了可以 select、 insert、 update 和 delete。

1. 使用帳號和密碼

在此 Mysql 所使用的帳號和密碼並不與作業系統使用的相同，因此你可以在進入 mysql 時使用與作業系統不同帳號的 -u 選項登入。

2. Mysql 所提供的權限

有關權限資訊使用到的資料表 user、 db、 host、 tables_priv 和 columns_priv

儲存在名為 mysql 的資料庫中)。在 Mysql 啟動時和在權限修後生效時，伺服器讀入這些資料庫表內容。

下表為 Mysql 所提供的權限名稱，包括了 Mysql 所會用到權限名稱和在授權表內相對應的欄位名稱，以及使用的範圍：

權限	欄位名	範圍
select	Select_priv	表
insert	Insert_priv	表
update	Update_priv	表
delete	Delete_priv	表
index	Index_priv	表
alter	Alter_priv	表
create	Create_priv	資料庫、表或索引
drop	Drop_priv	資料庫或表
grant	Grant_priv	資料庫或表
references	References_priv	資料庫或表
reload	Reload_priv	伺服器管理
shutdown	Shutdown_priv	伺服器管理
process	Process_priv	伺服器管理
file	File_priv	在伺服器上的文件存取

- select、insert、update 和 delete

以上權限是允許在資料庫中對資料表實施操作

- index

允許建立或刪除索引。

- alter

允許你使用 ALTER TABLE。

- create 和 drop

允許你建立新的資料庫和資料表，或刪除已存在的資料庫和資料表。

- grant

允許把自己所擁有的權限授權給其他的用戶。

- file

允許使用 LOAD DATA INFILE 和 SELECT ... INTO OUTFILE 語句讀和寫伺服器上的文件，任何被授予這個權限的用戶都能讀或寫 mysql 伺服器上任何文件。

3. Mysql server 進行系統權限控制

Mysql 會以連結時登入用的主機名稱和帳號，來決定該使用者的所具有的權

限，Mysql 允許來自不同的主機但相同的使用者名稱，並能給予這兩個不同來源的使用者各別的使用權限設定。

Mysql 的存取控制分為兩個階段

- 當連接 mysql 檢查你是否有連結的權限
- 每次對 mysql server 下達指令時，server 都會檢查你是否有執行此操作的權限

4. 建立初始的 mysql 使用權限

在 Mysql 安裝完畢業，server 會為系統建立一個超級使用者帳號 root，且為來自本地主機的連結。此帳號並沒有設上任何的密碼，因此任何人都可以以此帳號不需輸入密碼而進入，這是相當危險的。

另外 server 會建立一個匿名帳號，權限為來自本地主機的連結。此帳號允許對 test 和 test_開頭的資料庫做任何事，因此任何本地主機的使用者，亦可以匿帳號進入。

為了安全起見，我們應該對 root 帳號設定一個密碼，我們可以使用 password() 的函數來指定密碼：

```
# mysql -u root mysql
mysql> UPDATE user SET Password=PASSWORD('new_password')
      WHERE user='root';
mysql> FLUSH PRIVILEGES;
```

另外，你也可以使用 mysqladmin 來設定密碼

```
# mysqladmin -u root password new_password
```

5. 新增一個用戶權限

你可以用 GRANT 或 INSERT 的方式新增一個新的用戶，並給予適當的權限，在這之前，必須擁有 root 的權限，並且允許對名為 mysql 資料庫有 insert 和 reload 的權限。

透過 GRANT 語句新增帳號是比較簡單而清楚的方式，也比較不容易發生問題：

```
# mysql --user=root mysql
```

```
mysql> GRANT ALL PRIVILEGES ON *.* TO monty@localhost
IDENTIFIED BY 'something' WITH GRANT OPTION;
mysql> GRANT ALL PRIVILEGES ON *.* TO monty@"%"
IDENTIFIED BY 'something' WITH GRANT OPTION;
mysql> GRANT RELOAD,PROCESS ON *.* TO admin@localhost;
mysql> GRANT USAGE ON *.* TO dummy@localhost;
```

另外你也可以使用 INSERT 的方式將新帳號的資料直接寫入到 mysql 資料庫中：

```
# mysql --user=root mysql
mysql> INSERT INTO user
VALUES('localhost','monty',PASSWORD('something')),
'Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y')
mysql> INSERT INTO user VALUES('%','monty',PASSWORD('something')),
'Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y')
mysql> INSERT INTO user SET Host='localhost',User='admin',
Reload_priv='Y', Process_priv='Y';
mysql> INSERT INTO user (Host,User>Password)
VALUES('localhost','dummy','');
mysql> FLUSH PRIVILEGES;
```

2.7 Ftp

2.7.1 Proftpd 簡介

ProFTPD 是一個 Unix 平台上或是類 Unix 平台上(如 Linux, FreeBSD 等)的 FTP 服務器程序，它是在自由軟件基金會的版權聲明(GPL)下開發、發布的免費軟件，也就是說任何人只要遵守 GPL 版權聲明，都可以隨意修改源代碼。ProFTPD 設計目標是實現一個安全且易設定的 FTP Server。目前 Unix 或類 Unix 平台上 FTP Server 十分有限，最常使用的恐怕就是 wu-ftp 了。雖然 wu-ftp 有著極佳的效能同時也是一套很好的軟件，然而它卻欠缺了許多 Win32 平台上 FTP Server 的一些特色，同時 wu-ftp 過去也有不少的安全漏洞陸續被發現。ProFTPD 的原創者本身就曾經花非常多的時間尋找 wu-ftp 的漏洞加以改進並且增加許多功能。然而十分不幸的是，他很快地發現顯然 wu-ftp 需要全部重新的改寫才能補足欠缺的設定能力以及缺乏的一些功能。ProFTPD 不是從其它 FTP Server 的既有原始碼修改而產生的，相反的，它是完全獨立而完整、重新改寫的 FTP Server。

ProFTPD 的主要包括如下特點：

- 單一主設置文件，包含許多指令以及其支配的組態，Web Serve 的設置不陌生的話相信一定會覺得十分容易操作設定。
- 每個目錄都可以定義一個 ".ftppass" 設置文件，就如同 Apache 的 ".htaccess" 文件一樣可以設定該目錄的存取權限。
- 可設定多個虛擬 FTP server，而匿名 FTP 服務更是十分容易。
- 可根據系統的負載(load)選擇以單獨運作(stand-alone)方式或是由 inetd 啟動。
- 匿名 FTP 的根目錄不需要特定的目錄結構、系統二進制執行文件或其它系統文件。
- ProFTPD 不執行任何外部程序以免造成安全漏洞。
- 具有隱藏目錄或隱藏文件，源自 Unix 形式的檔案權限，或是 user/group 類型的檔案權限設定。
- 能夠以一般使用者在單獨運作(stand-alone)模式下執行，以減少某些藉由攻擊方式取得 root 權的可能性。注：此功能與 Unix 的操作系統有關。
- 支持系統記錄以及 utmp/wtmp。記錄的方式與 wu-ftp 的標準完全兼容，也支持記錄內容的延伸格式。
- 支持 Shadow! nbsp;password，包括了帳號使用期限設定的功能。

2.7.2 Proftpd 下載和安裝

Proftpd 的原始程式可以在 Proftpd 的官方網頁 <http://proftpd.linux.co.uk/> 中找到，也可以在國內各大 ftp 站找到，目前最新版本是 proftpd-1.2.9.tar.gz 這隻程式，請下載 ver1.2.9。下載到原始程式碼後，我們就可在我們主機上面進行安裝。

```
# cd /usr/local/src
```

切換到我們要安裝的目錄下

```
# tar -zxvf /root/proftpd-1.2.9.tar.gz
```

此步驟會在現行目錄下解壓縮並產生一個 proftpd-1.2.9 的目錄

```
# cd proftpd-1.2.9
```

```
# ./configure --prefix=PREFIX \
```

```
> --enable-shadow --enable-autoshadow \
```

```
> --with-modules=mod_module1:mod_module2:mod_module3
```

執行組態設定檔，PREFIX 是預設要安裝的目錄，--enable-shadow 與 --enable-autoshadow 可以讓系統的/etc/shadow 為我們的 FTP 登入時進行密碼驗證。

```
# make
```

```
# make install
```

將安裝檔編譯好，並把檔案安裝到指定的目錄下

為了讓我們的 Proftpd 能在 inetd 下執行，我們需要修改/etc/inetd.conf 這個檔案，請用文字編輯器打開，找到底下的文字

```
ftp stream tcp nowait root in.ftpd in.ftpd
```

將上面這段文字替換成下面的文字

```
ftp stream tcp nowait root in.proftpd in.proftpd
```

接著從 process listening 中找到你的 inetd process，並送出 SIGHUP，他會自己重新整理，並重新分配。再來你也要將 in.ProFTPD 加到你系統的 hosts.allow 裡。

在 Proftpd 裡有幾個常用到的程式，必須先了解一下

- proftpd.cong

這個就是主要的 proftpd 的設定檔了！有關於 Proftpd 相關的設定都在這裡面，關於他的設定語法，可以參考官方網站的解說

http://proftpd.linux.co.uk/docs/directives/configuration_full.html

，在我們後面介紹的，將只是簡單的設定動作。

- proftpd

proftpd 的 dameno 執行程式，要執行 proftpd 要先啟動他，啟動 proftpd 時會呼叫他的設定檔，我們要指定 proftpd 設定檔給他

```
#proftpd -c 設定檔的檔名
```

- ftpcount

查尋目前在主機上面使用 proftpd 的連線數，在指令列下達 ftpcount 即可！

- ftpshut [-l 分鐘] -d [分鐘] 時間 "訊息"

指定要在多久之後關閉 ftp，為了某些原因要關閉 ftp 時，可以先提出警訊給目前連線的使用者，並對於新要求的連線予以拒絕。

-l：指定關閉前的幾分鐘開始不接受新要求的連線

-d：指定關閉前幾分鐘將已建立的連線給予中止

時間：指定什麼時候或多久後關閉，時間可以有兩種格式

+minutes：指再過幾分鐘後關閉

MMHH：在今天的 MMHH 時關閉

訊息：告知使用者的訊息

- ftpwho

用來觀看目前線上有多少位使用者

```
# ftpwho
standalone FTP daemon [8451]:
10194 badbird [ 0m11s] 0m6s (idle)
Service class - 1 user
```

2.7.3 Proftpd 的設定檔

Proftpd.conf 的設定檔語法與 Apache HTTP server 很像，習慣 Apache 設定的人，應該對 Proftpd 也能很輕易的上手。一般來說指令項目後面接著參數，也有區段標籤語法可以包住一堆指令集合，如果是註解會在該行前面以#開頭。

常用的區段標籤有

<DIRECTORY 完整目錄名稱>...</DIRECTORY>

用來設定某一個目錄的相關指令

<ANONYMOUS 匿名登入時候的匿名者根目錄>...</ANONYMOUS>

用來設定匿名進入時的設定值

<LIMIT 動作>...</LIMIT>

用來限制使用者不可執行的動作，動作的種類有：

- CWD : Change Working Directory, 變換目錄之意；
- MKD : MaKe Directory, 可否建立目錄；
- RNFR : ReName FRom, 可否更改檔名；
- DELE : DELEt, 可刪除檔案語法；
- RMD : ReMove Directory, 可否移除目錄；
- RETR : RETRieve, 下載之意！由 Server 傳送資料到 Client；
- READ : 可否讀取
- WRITE : 可否寫入
- STOR : STORe, 上傳之意，由 Client 傳送資料到 Server ！
- ALL : 全部的動作！

當 proftpd 裝設好時，會有設定好簡單的 proftpd.conf，大概內容如下：

```
# This sample configuration file illustrates configuring two  
# anonymous directories, and a guest (same thing as anonymous but  
# requires a valid password to login)
```

```
ServerName "Proftpd server"
```

主機的名稱，會顯示在 client 的 ftp 程式上的資訊

```
ServerType standalone
```

啟動 proftpd 的方式，一般有兩種，分別是 standalone 和 inetd。如果是 super daemon 啟動，則設定為 inetd。如果是獨立方式啟動，則設定為 standalone

```
SysLogLevel emerg
```

設定 server 的記錄檔的詳細程度，共有 emerg、alert、crit、error、warn、notice、info|debug 這幾種

```
SystemLog /var/log/proftpd.log
```

系統記錄檔的位置

```
# Port 21 is the standard FTP port.
```

```
Port 21
```

使用通訊埠，預設為 21

```
# If you don't want normal users logging in at all, uncomment this
```

```
# next section
```

```
#<Limit LOGIN>
```

```
# DenyAll
```

```
#</Limit>
```

```
# Set the user and group that the server normally runs at.
```

```
User nobody
```

```
Group nogroup
```

proftpd 預設的服務啟動者！後面接的使用者與群組必須在 /etc/passwd 與 /etc/group 已裡面存在！

```
DefaultRoot ~ badbird
```

設定預設登入根目錄，~是家目錄，後面接的是群組

```
# To prevent DoS attacks, set the maximum number of child processes
```

```
# to 30. If you need to allow more than 30 concurrent connections
```

```
# at once, simply increase this value. Note that this ONLY works
```

```
# in standalone mode, in inetd mode you should use an inetd server
```

```
# that allows you to limit maximum number of processes per service
```

```
# (such as xinetd)
```

```
MaxInstances 30
```

同一時間可以接受的連線數，預設為 30

```
# Set the maximum number of seconds a data connection is allowed
```

```
# to "stall" before being aborted.
```

```
TimeoutStalled 300
```

設定使用者停止傳送資料後幾秒要斷開連線

```
# We want 'welcome.msg' displayed at login, and '.message' displayed
```

```
# in each newly chdir'd directory.
```

```
DisplayLogin welcome.msg
```

```
DisplayFirstChdir .message
```


設定登入時顯示的歡迎檔，和更換目錄時顯示的訊息檔

```
# Our "basic" anonymous configuration, including a single  
# upload directory ("uploads")
```

```
<Anonymous ~>
```

設定匿名登入者的登入目錄為家目錄

```
Umask 002
```

設定下載速度比

```
# Allow logins if they are disabled above.
```

```
<Limit LOGIN>
```

限制登入的條件

```
AllowAll
```

AllowAll 表示全部允許

```
</Limit>
```

```
# Maximum clients with message
```

```
MaxClients          5 "Sorry, max %m users -- try again later"
```

允許最多 5 個匿名登入，超過時，會顯示後面的訊息

```
RequireValidShell   off
```

不要讓 proftpd 檢查 /etc/passwd 檔案內 ftp 使用者 shell。

```
AnonRequirePassword off
```

當匿名登入時，設定為不使用密碼。

```
User                ftp
```

```
Group                ftp
```

process 取得 user 和 group 權限

```
# We want clients to be able to login with "anonymous" as well as "ftp"
```

```
UserAlias            anonymous ftp
```

設定匿名帳號的別名

```
# Limit WRITE everywhere in the anonymous chroot
```

```
<Limit WRITE>
```

```
DenyAll
```

寫入的權限被取消

```
</Limit>
```

```
# An upload directory that allows storing files but not retrieving  
# or creating directories.
```

```
<Directory upload/*>
```

```
<Limit READ>
```

```
DenyAll
```

```
</Limit>
```

```
<Limit STOR>
```

```
AllowAll
```

```
</Limit>
```

```
</Directory>
```

設定上傳的資料夾，權限為可以存檔，但不能讀取或建立目錄

```
</Anonymous>
```

2.7.4 編輯歡迎檔

welcome.msg 可以文字檔編輯，其中可以使用到幾個參數來幫主管理者表達變動的資訊給使用者觀看，可以使用的參數如下：

- %T 目前的時間
- %F 所在硬碟剩下的容量
- %C 目前所在的目錄
- %R Client 端的主機名稱
- %L Server 端的主機名稱
- %U 使用者帳號名稱
- %M 最大允許連線人數
- %N 目前的主機連線人數
- %E FTP 主機管理員的 email
- %i 本次上傳的檔案數目
- %o 本次下載的檔案數量
- %t 本次上傳+下載的檔案數量

建立一個特殊帳號和 guest 使用群組

我們可以建立一個群組，並給予一些特殊的上下載權限。
請先建立三個使用者帳號 ftpuser1, ftpuser2, ftpuser3, 並讓這三個帳號的群組皆為 ftpguest :

```
# groupadd ftpguest
# useradd -M -g ftpguest -s /bin/false ftpuser1
# useradd -M -g ftpguest -s /bin/false ftpuser2
# useradd -M -g ftpguest -s /bin/false ftpuser3
# passwd ftpuser1
```

接下來我們建立一個資料夾，並開放給 ftpguest 能寫入

```
# mkdir -p /var/ftp2
# mkdir -p /var/ftp2/upload
# mkdir -p /var/ftp2/work
# chmod -R 775 /var/ftp2
```

建立用來記錄上下載比例的兩個檔案

```
# touch /var/ftp2/work/ratio.dat
# touch /var/ftp2/work/ratio.tmp
# chown -R ftpuser1:ftpguest /var/ftp2
# chmod 666 /var/ftp2/work/*
```

再下來打開 proftpd.conf 檔，沿用原來的主機、實體設定程和匿名設定，再加入我們底下有關於 /var/ftp2 的設定：

```
DefaultRoot /var/ftp2 ftpguest
```

預設根目錄位置和使用群組

```
DisplayLogin welcome.msg
```

設定登入時的訊息

```
Ratios on
```

```
SaveRatios on
```

設定使用上傳下載比

```
RatioFile /work/ratio.dat
```

```
RatioTempFile /work/ratio.tmp
```

設定上下載的記錄

```
<Directory /var/ftp2/work>
```

```
<Limit All>
```

```
Denyall
```

```
</Limit>
```

```
</Directory>
```

將/var/ftp2/work 的目錄設定為不可使用

```
UserRatio ftpuser1 0 0 2 100000
```

```
UserRatio ftpuser2 0 0 1 30000
```

```
UserRatio ftpuser3 0 0 1 30000
```

用 UserRatio 指令設定上傳下載比例，語法為：

```
UserRatio "使用者帳號" fileratio filequota byteratio bytequota
```

fileratio 是指定是否要以檔案數為上下傳的基準，0 表不使用

filequota 預設能下載的檔案數，0 表不設限

byteratio 此為上傳和下載的比例，[1:byteratio]

bytequota 預設能下載多少 Kbytes 檔案

```
<Directory /var/ftp2>
```

```
Umask 002
```

```
TransferRate RETR 50 group ftpguest
```

```
<Limit WRITE>
```

```
Denyall
```

```
</Limit>
```

```
</Directory>
```

設定/var/ftp2 的使用權限，並禁止寫入

```
<Directory /var/ftp2/upload/*>
```

```
<Limit READ>
```

```
Denyall
```

```
</Limit>
```

```
<Limit WRITE>
```

```
Allowall
```

```
</Limit>
```

```
</Directory>
```

設定上傳的目錄/var/ftp2/upload/*，並開放寫入

如此我們就可以開放給 ftpuse1, ftpuse2, ftpuse3 三個帳號特殊的使用權限了。

2.8 Sendmail

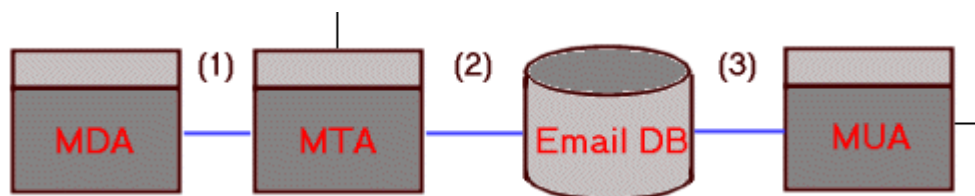
2.8.1 郵件伺服器

目前使用電子郵件統來收發信件，已經是日常生活的一部分了，有一個笑話說，未來你跟你的好朋友失去了連絡，可能就是因為你弄丟了他的 email。可見電子郵件對我們的影響是多們的深遠。

Mail 的應用軟體主要有傳輸部份以及讀取界面部份，以 Unix 環境來說，傳輸部份（指 SMTP）內建在 Sendmail (或 Smail) 裡面，而讀取部份就是 MUA。

當使用者完成寫信動作，MUA 把該信的內容交付給實際負責信件投遞的軟體，負責信件投遞這層面的軟體就是 MTA (Mail Transport Agent)。以上 MUA、MTA 這些名詞其實一點也不重要，比較重要的是對 Mail 的運作的理解。

SMTP (Simple Mail Transfer Protocol, RFC 821) 是規範 Internet 電子郵件遞送的標準（另外 RFC 822 規範信件格式）。SMTP 的投遞方式是點對點 (End-to-end) 交談式傳送，這傳送是即時的，中間可以不經過第三者轉遞。不過由於 Internet 上的 Mail 主機系統橫跨相當多不同的網路，因此存在所謂 Mail Gateway (或 Mail Relay)。



2.8.2 安裝 sendmail

要安裝 sendmail 需先取得安裝檔案，相關的檔案可以在官方網頁找到 <http://www.sendmail.org>，目前最新穩定版本是 Sendmail 8.12.10，請下載此版本。

將下載來的檔案解壓縮，請執行下列命令

```
# tar zxvf sendmail-8.12.10.tar.gz
```

解開後會在現行目錄下產生一個 sendmail-8.12.10 的目錄

```
# cd sendmail-8.12.10
```

切換到我們解開的目錄下

```
# sh Build
```

編譯 sendmail 相關檔案

編譯結束後，切換到 cf/cf 下的目錄並複製 config.mc 這個檔

```
# cd cf/cf
```

```
# cp generic.linux.mc config.mc
```

將 config.mc 設定檔轉換成 .cf 格式

```
# sb Build confi.cf
```

切換回 sendmail 的根目錄，就可以進行安裝指令了

```
# cd /temp/sendmail-8.12.10
```

```
# sh Build install
```

安裝完成之後，便要啟動 sendmail

```
# /etc/rc.d/init.d/sendmail start
```

2.8.3 sendmail 的設定檔

- 設定帳號別名

設定帳號別名，可以將長帳號的電子郵件，用較短的別名取代。帳號別名記錄在 /etc/aliases 裡，請用文字編輯器打開，並加入你要的帳號，和其別名，例如：

```
jason:      jasonkao1975
jasonkao:   jasonkao1975
james      jameslee1977
mary       marylin1978
office:    jasonkao1975, jameslee1977, marylin1978
```

每一個帳號結尾用冒號來區隔，並且同帳號可以建立多個別名，同一個別名也可以指定給多個帳號，當信件寄給 office 時 jasonkao1975, jameslee1977, marylin1978 都會同時收到。

- 設定郵件服務限定帳號

如果你想對某些使用者只提供郵件服務，而不想提供其他服務，例如 telnet 及 ftp，那麼你可以執行下列命令：

```
chsh -s /bin/false 使用者帳號
```

如此之後，該帳號就只能使用郵件服務了。

- 設定郵件限制網域

為了防止濫發垃圾信件，此設定可以指令某些網域的使用者才可以收發信件。用文字編輯打/etc/mail/access 檔案來編修。

指定網址 指定動作

指定網址：指一個電子郵件位址、某個主機 IP 或網域名稱

指定動作：可以使用的動作有四種，如下

OK	允許收發指定網址的信件
RELAY	允許中繼指定網址的所有要求轉寄的信件
REJECT	拒絕收發指定網址的信件
DISCARD	刪除指定網址所發出的信件

編修完之後，記得要行下列指令，修改才會生效喔。

```
makemap -r hash /etc/mail/access.db < etc/mail/access
```

2.8.4 安裝 POP3 和 IMPA

安裝 sendmail 之後，你已經可以利用自己的郵件伺服器發信了，但如果想要由郵件伺服器中取回電子郵件，則還需要安裝 POP 及 IMAP，大部分的 LINUX 版本都會有這兩個套件，方便我們進行安裝，如果你是使用 RedHat 請用 rpm 來安裝，請先取得 IMPA 的 rpm 套件。

```
rpm -ivh imap-2001a-15.i386.rpm
```

在 RedHat 套件中，已包含了 POP 和 IMAP 兩個伺服器，會同時安裝上去。

安裝完成之後，我們還需要進行設定檔的編修，請用文字編輯器打開 /etc/services，我們需要把 POP2，POP3，IMPA 三個通訊埠給打開：

```
第 73 行-->#pop-2 109/tcp postoffice # POP version 2
```

```
改成-->pop-2 109/tcp postoffice # POP version 2
```

```

第 74 行-->#pop-2 109/udp
改成-->pop-2 109/udp
第 75 行-->#pop-3 110/tcp # POP version 3
改成-->pop-3 110/tcp # POP version 3
第 76 行-->#pop-3 110/udp
改成-->pop-3 110/udp
第 91 行-->#imap2 143/tcp imap # Interim Mail Access
改成-->imap2 143/tcp imap # Interim Mail Access
第 92 行-->#imap2 143/udp imap # Protocol v2
改成-->imap2 143/udp imap # Protocol v2
第 125 行-->#imap3 220/tcp # Interactive Mail Access
改成-->imap3 220/tcp # Interactive Mail Access
第 125 行-->#imap3 220/udp # Protocol v3
改成-->imap3 220/udp # Protocol v3

```

再接下來，我們還需要修改啟動 POP 及 IMAP 的設定。現在在用文字編輯器打開/etc/xintend.d/imap

```

service imap
{
    socket_type          = stream
    wait                 = no
    user                 = root
    server               = /usr/sbin/imap
    log_on_success      += HOST DURATION
    log_on_failure      += HOST
    disable              = no
    將 disable = yes 改為 no
}

```

再來還需要啟動 POP3 服務，請編輯/etc/xintend.d/ipop3

```

service pop3
{
    socket_type          = stream
    wait                 = no
    user                 = root
    server               = /usr/sbin/imap
    log_on_success      += HOST DURATION
    log_on_failure      += HOST
}

```



```
disable = no
    將 disable = yes 改為 no
}
```

最後 POP2 也要啟動，請編輯/etc/xinted.d/ipop2

```
service pop2
{
    socket_type = stream
    wait = no
    user = root
    server = /usr/sbin/imap
    log_on_success += HOST DURATION
    log_on_failure += HOST
    disable = no
    將 disable = yes 改為 no
}
```

最後重新讓 xinted 讀取設定檔

```
# /etc/rc.d/init.d/xinted reload
```

2.8.5 建立虛擬郵件地址

你可以修改 sendmail.cf 讓 sendmail 接收不同 domain 的郵件了。但如果目的地的信箱是在本 sendmail 機器上的話，所使用的帳號將會是 Linux 的系統帳號。舉個例子說：

我們修改 /etc/sendmail.cf 讓 sendmail 可以接收另外兩個 domain 的郵件：

```
#
#--- 下面是虛擬 domain: ---
v1.test.domain
v2.test.domain
```

負責這兩 domain 的郵件主機分別為 mail.v1.test.domain 和 mail.v2.test.domain。這裡，我們假設這兩台虛擬主機的 IP Alias 和路由都已經設定好了。同時 DNS 那邊也已經設定好這兩個 domain 了，MX 和 A 記錄都指向本主機。

然後，我們建立兩個帳號：v1test1 和 v2test2。當 sendmail 接收到 v1test1@v1.test.domain 和 v2test2@v2.test.domain 這樣的郵件時，它可以輕鬆的將郵件分寄給 v1test1 和 v2test2 這兩位使用者。他們只要將 POP

server 分別設為 mail.v1.test.domain 或 mail.v2.test.domain ，或乾脆用本機的真实名稱：redhat52.siyongc.domain ，就可以將信件取回去閱讀。

但如果 v1.test.domain 和 v2.test.domain 都想有一個帳號 test 來接收各自的郵件呢？注意：他們可不是相同的人哦，怎麼辦呢？

我們可以用虛擬郵件伺服器來處理：修改 /etc/mail/virtusertable，加入這兩行：

```
test@v1.test.domain v1test1
test@v2.test.domain v2test2
```

然後執行：

```
makemap -r hash /etc/mail/virtusertable.db < /etc/mail/virtusertable
service sendmail restart
```

這樣的話，v1test1 就可以收到 test@v1.test.domain 的郵件了，而 test@v2.test.domain 則會轉給 v2test2 這位使用者。這時候，v1test1 的 POP server 可以用 mail.v1.test.domain，而 v2test2 則可以用 mail.v2.test.domain。

但有一個問題是：在 v1 和 v2 這兩 domain 下，他們不能共用 test 這個帳號來登錄：要麼將 test 給 v1 用，讓 v2 繼續用 v2test2；如果兩人都用 test 的帳號來取信，是否還有其他的方法呢？

我們先檢察底下的程式是否存在：

```
/usr/lib/linuxconf/lib/vpop3d
```

請依底下的驟設定

1. 完成並確定 IP Alias、routing、DNS 等相關設定，為 domain 本身指定一個 A 記錄。下面是 v1.test.domain 的正確示範：

```
$TTL 86400
@ IN SOA linpus64.dmz.domain. root.linpus64.dmz.domain.
      ( 00071401 7200 3600 3600000 86400 )
@ IN NS linpus64.dmz.domain.
@ IN MX 10 mail.v1.test.domain.
@ IN A 203.30.35.132
mail IN A 203.30.35.132
      IN MX 5 mail.v1.test.domain.
```

2. 將 /etc/sendmail.cf 備份好：
`cp /etc/sendmail.cf /etc/sendmail.cf.bak`
3. 修改 /etc/inetd.conf ，找到 pop-3 那句，將之修改如下：
`pop-3 stream tcp nowait root /usr/sbin/tcpd
/usr/lib/linuxconf/lib/vpop3d ipop3d`
4. 接著執行：
`service inet restart`
5. 再來執行：
`linuxconf`
6. 在 Config 下選擇 Networking --> Server tasks --> Mail delivery system (sendmail) --> Basic --> Configure virtual email domain.
7. 按 Tab 鍵跳到 Add ，在 Virtual domain (fqdn) 後面填上虛擬 郵件的 domain 名稱，如：v1.test.domain。
8. 按 Tab 鍵跳到 Accept ，並 Enter。
9. 按 Tab 鍵跳到 Quit ，並 Enter。
10. 當看到 “ Generating /etc/sendmail.cf ” 警告的時候，按 Tab 鍵跳到 Yes ，並 Enter。(如果再有警告字眼的話，也是按 Tab 鍵跳到 Yes)，最後 OK。
11. 在 Config 下選擇 Users accounts --> Special accounts --> Virtual POP accounts (mail only)。
12. 選擇剛才建立的虛擬郵件主機，並 Enter。
13. 按 Tab 跳到 Add，在 Login name 輸入虛擬帳號名稱，如：test。
14. 按 Tab 鍵跳到 Accept ，並 Enter。

15. 輸入第一次密碼並 Enter，然後才輸入第二次密碼。如果您錯過了，多按幾次 Tab 就是了。
16. 按 Tab 鍵跳到 Accept，並 Enter。然後 OK。
17. 按 Tab 鍵跳到 Quit，並 Enter。連續三次，直至見到 “ Activate the changes ” 字句，再直接 Enter。

完成以上的步驟，就完成了虛擬郵件主機的設定了。您只要將 client 的 POP server 指向虛擬郵件主機(mail.v1.test.domain)就可以將郵件下載回來。

Ch3 Windows 系列之安裝設定

3.1 IIS (Internet Information Server)

3.1.1 簡介

IIS 是一套由微軟公司所推出的網際網路資訊伺服器，說它是資訊伺服器一點也不誇張，他本身提供了一般最常使用到的網路服務，它一向又與其自家的網路作業系統緊密的結合在一起，為的是要確保使用團體在建置 Internet 或是 Intranet 的安全性，由於嚴格的安全管制，使得管理 IIS 相當容易，並利用與 Microsoft Proxy Server、Certificate Server、Site Server、BackOffice 以及其他應用程式緊密結合之便，提供一套內建整合性、極度安全的作業平台。因此無論是使用 Internet 或是 Intranet 都可以在不同的平台上迅速地建置網站。

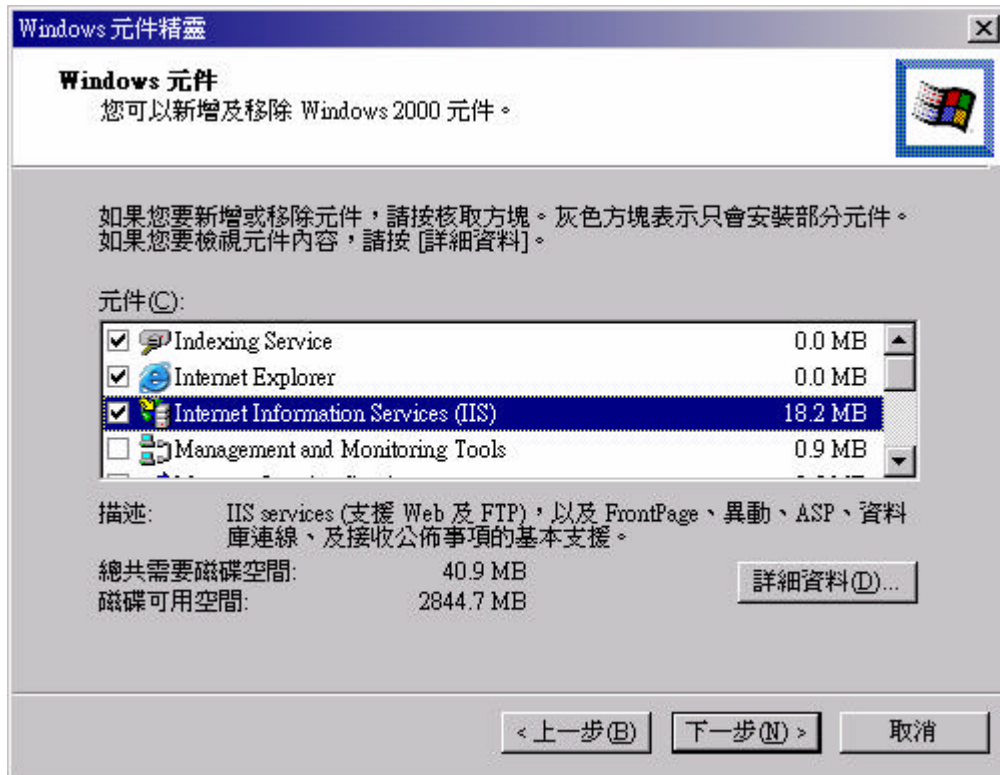
Windows 2000 Server 整合了 IIS 5.0，在預設的情況下，安裝 Windows 2000 Server 時便會自動安裝 IIS 5.0。

IIS 5.0 主要提供了下面四種服務：

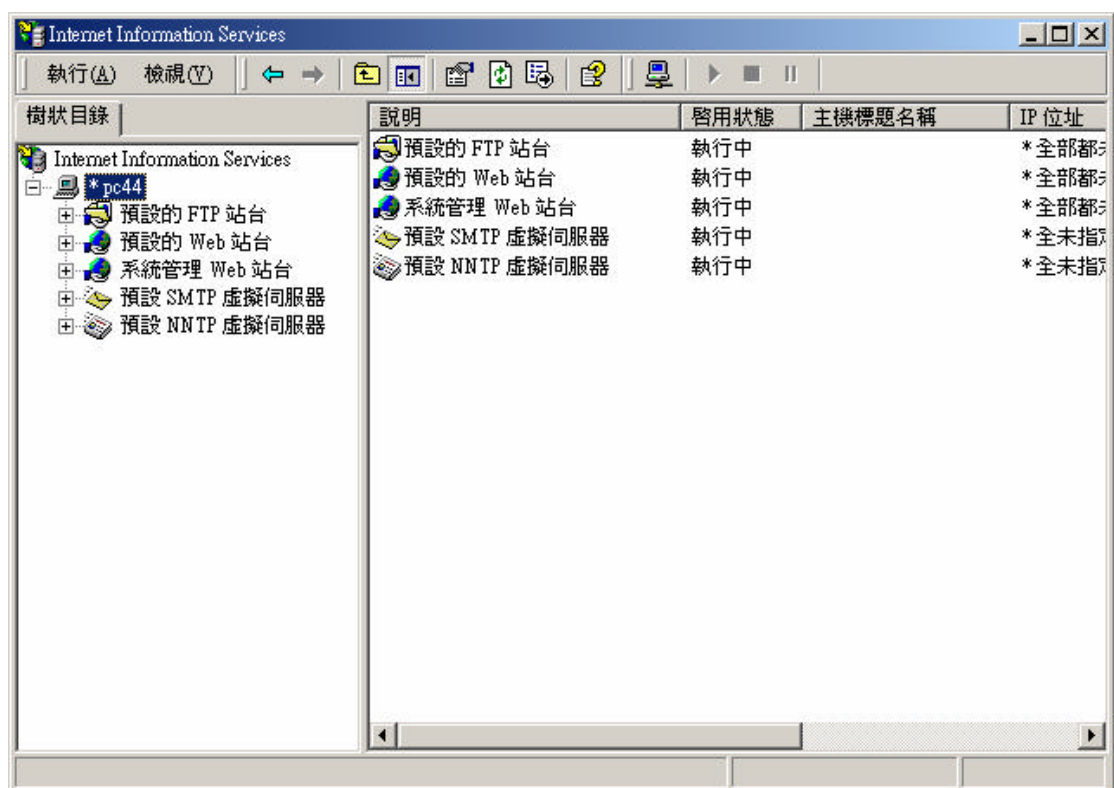
- World Wide Web Publishing Service (WWW Service)
網站伺服器 (Web Server)。
- FTP Publishing Service (FTP Service)
檔案傳輸伺服器 (FTP Server)。
- SMTP Service
郵件伺服器 (Mail Server)。
- NNTP Service
新聞伺服器 (News Server)。

3.1.2 安裝

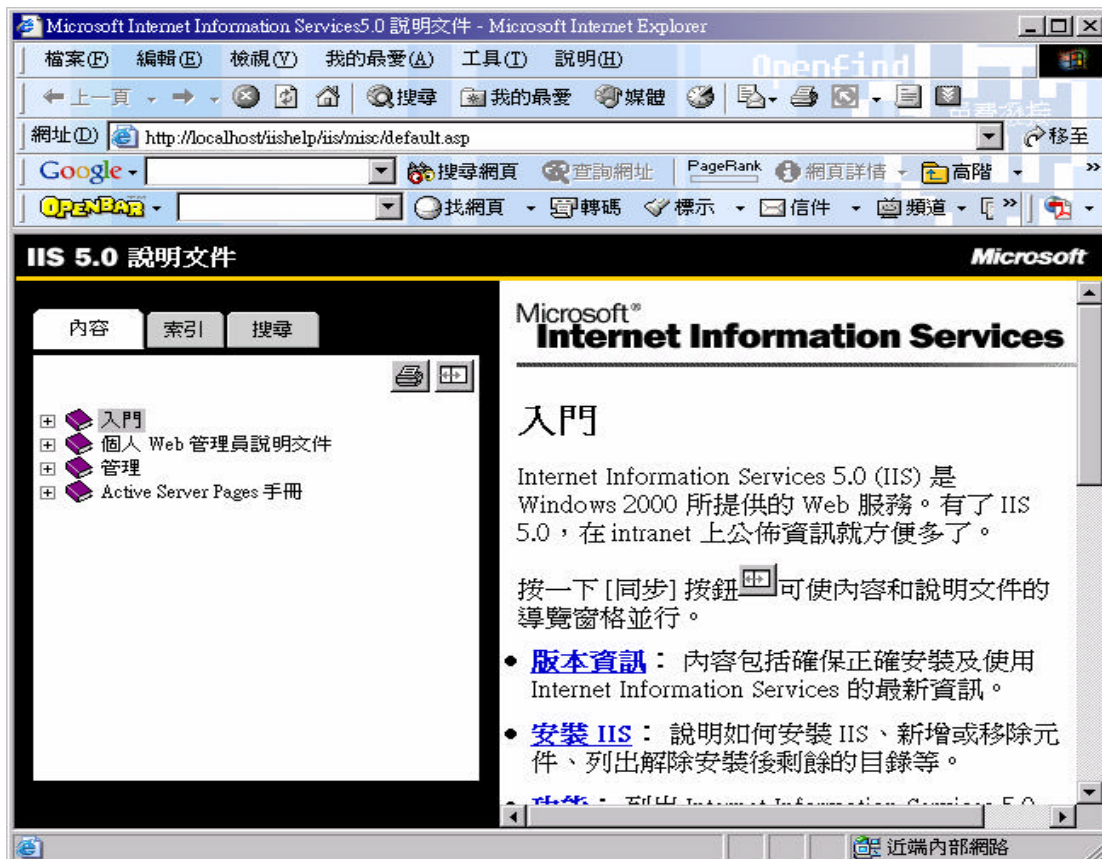
步驟一：從「控制台」→「新增或移除程式」→「新增/移除 Windows 元件」，勾選「Internet Information Services (IIS)」安裝 IIS 伺服器全部套件，或點選下方「詳細資料」進行細部的套件選取。



步驟二：安裝完後，可由「系統管理工具」 「Internet 服務管理員」管理整個 IIS 伺服器。



步驟三：「預設的 Web 站台」裡的內容是 IIS 的歡迎畫面及說明文件。



3.2 DNS (Domain Name System)

3.2.1 簡介

DNS 是一個架構在樹狀階層式結構上的一種分散式領域名稱服務，就像電話號碼一樣，每一部在 Internet 上的電腦都有一個唯一的位址，我們稱它作 IP address (IP 位址)。隨著網路的蓬勃發展，愈來愈多的數字化位址讓我們無法記住每一部電腦的 IP address，所以我們改用較熟悉的且有意義的語言來記，就是所謂的完整領域名稱 (FQDN)，一個完整領域名稱 (Fully Qualified Domain Name, FQDN) 是由主機名稱與領域名稱所組成。

每當我們要以完整領域名稱連結網路上的主機時，透過 DNS 將完整領域名稱解析成 IP address，我們就可以順利的悠遊 Internet 了。

3.2.2 原理

■ 根領域 (Root domain)

根領域是 DNS 樹狀階層結構中的最高層，以句點 “.” 表示，紀錄著所有頂層領域 (Top-Level Domain, TLD) DNS server 的資料，透過這些 DNS Server，我們就可以從 Root DNS Server 一層一層的往下找，直到找到實際負責該區域的 DNS Server。

■ 頂層領域 (Top-Level domain)

頂層領域是根據組織類別或地理區域來劃分，下列為頂層領域的例子：

頂層領域名稱	說明
com	民間企業
edu	教育學術單位
org	非商業組織
tw	台灣的國家領域名稱
cn	中國大陸的國家領域名稱
biz	商業或民間企業 New
museum	博物館 New

■ 第二層領域 (Second-Level domains)

第二層領域包含了兩部份：頂層領域名稱和獨立的下一層領域名稱，可供個人或團體登錄。

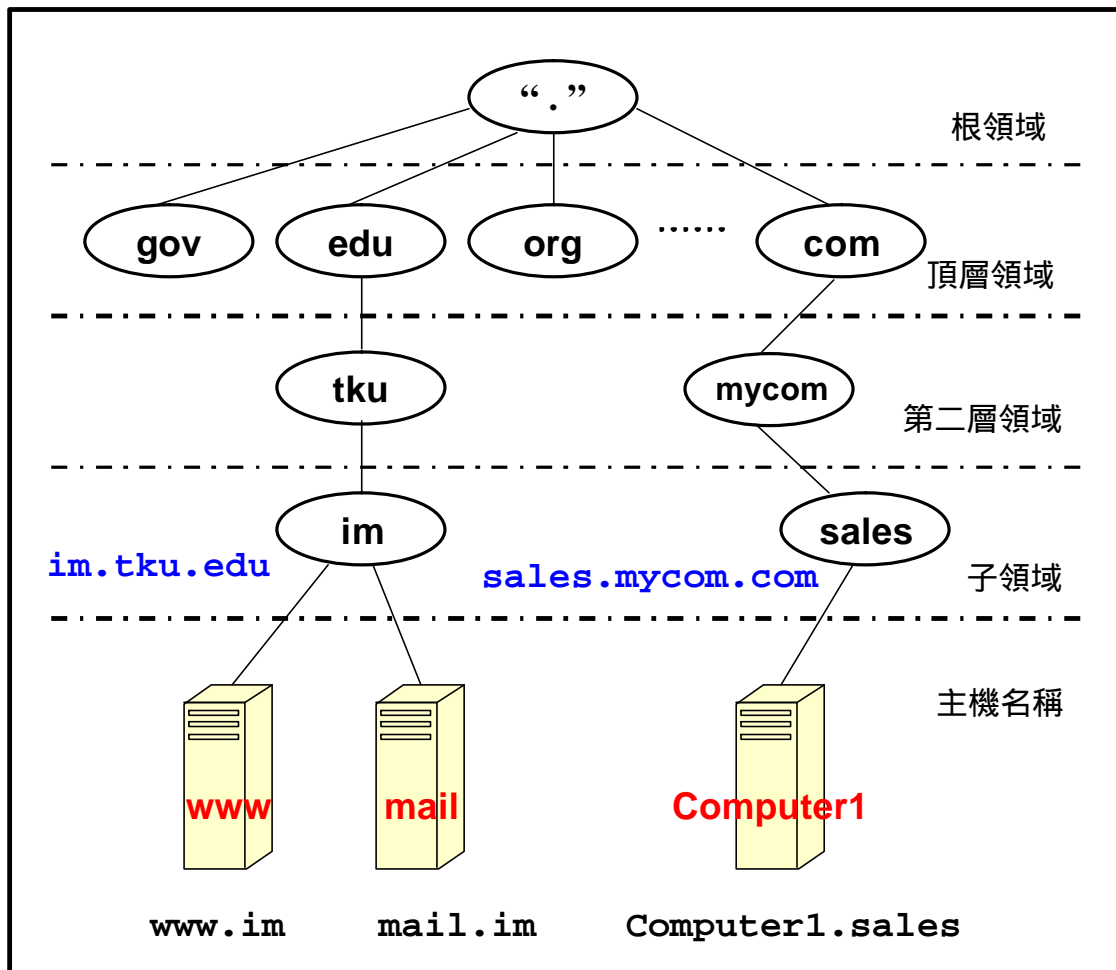
領域名稱	說明
ibm.com	國際商業機器
tku.edu	淡江大學
w3.org	World Wide Web 組織
president.gov.tw	台灣的總統府

■ 子領域 (Subdomain)

企業或團體依內部的部門、群組或地區分別建立其子領域，以利管理，如：
sales.mycom.com。

■ 主機名稱 (Host Names)

主機名稱就是在電腦在網路上的名稱。



我們以下圖為例來說明 DNS 服務的流程：

步驟一：Client 端向其 DNS 伺服器查詢 www.im.tku.edu.tw 的 IP 位址。

步驟二：Client 端所屬的 DNS 伺服器若無法在其資料庫中找到跟
www.im.tku.edu.tw 相對應的 IP 位址，Client 則轉而詢問 Root DNS 伺
服器。

步驟三：Root DNS 伺服器根據登記的資料，請 Client 端詢問管理 tw 這個區域

的 DNS Server。

步驟四：tw 伺服器再根據登記的資料，查出 `www.im.tku.edu.tw` 這個名稱登記在負責 edu 區域的 DNS 伺服器，所以 Client 端詢問 edu 的 DNS 伺服器。

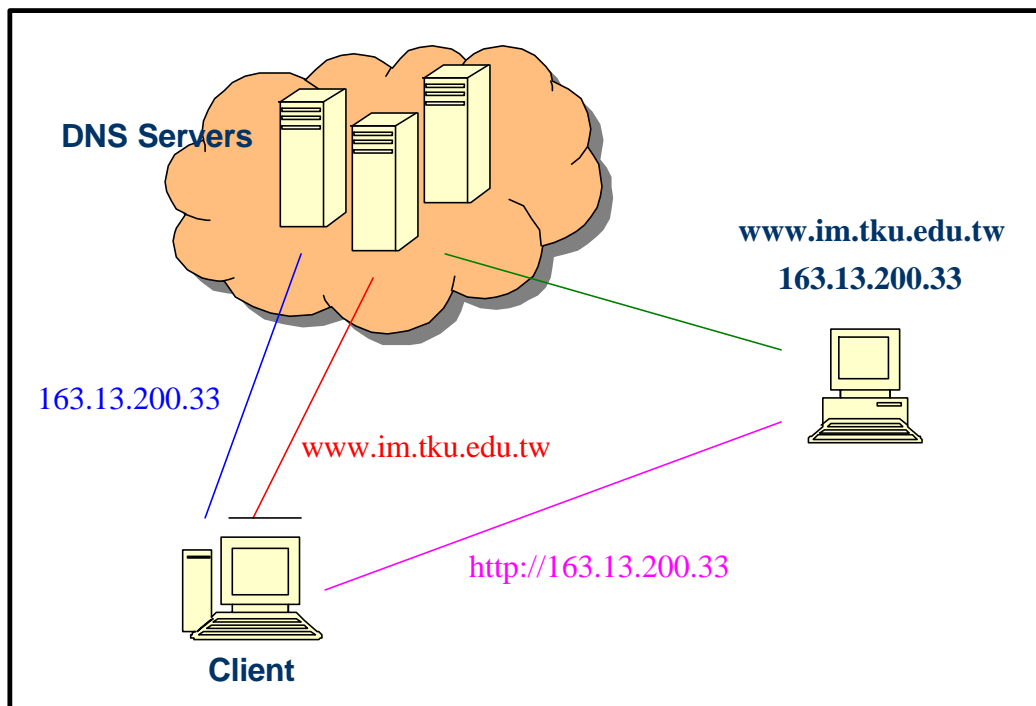
步驟五：相同的，Client 再轉而詢問 tku 的 DNS 伺服器。

步驟六：在 tku 的 DNS 伺服器上查出相對應的 IP 位址。

步驟七：將 IP 位址傳回給 Client 端。

步驟八：Client 端以對 `163.13.200.33` 這個 IP 位址發出需求。

這個過程看來繁複，但實際上所花的時間非常的短。

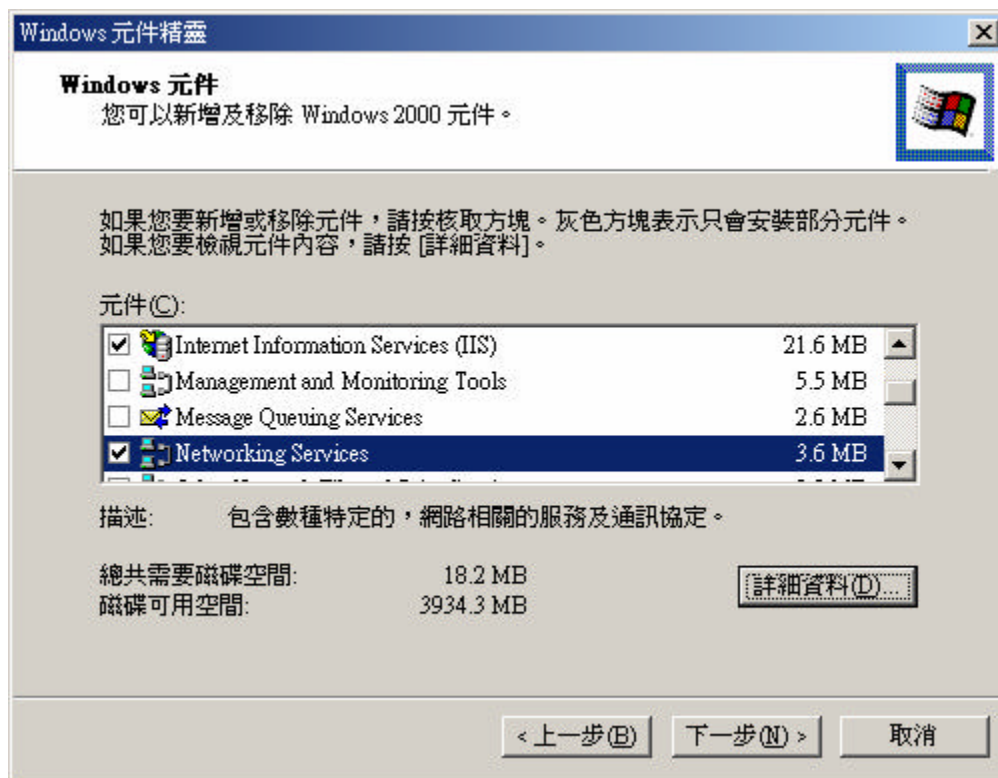


3.2.3 安裝

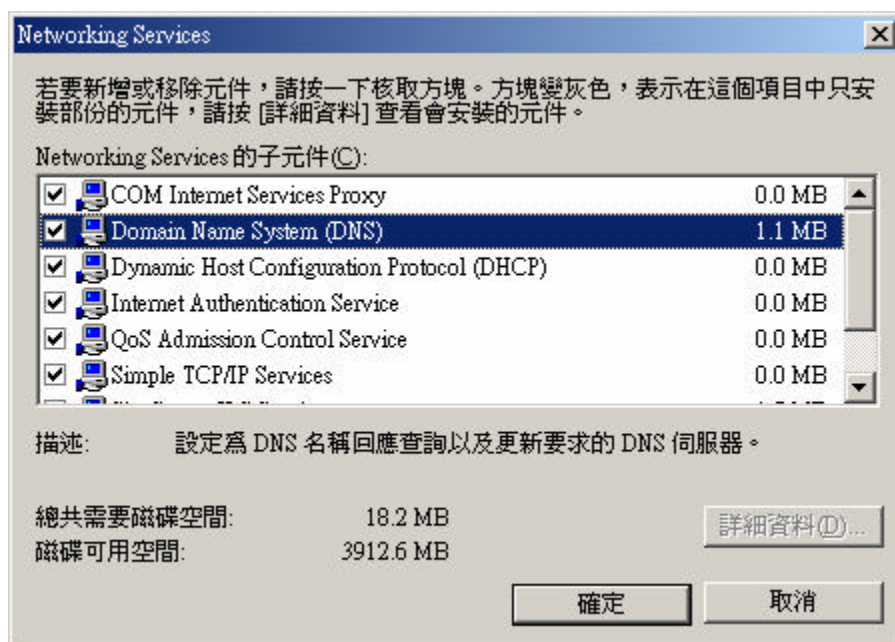
步驟一：首先由「控制台」選擇「新增/移除程式」，在新增/移除程式對話窗左邊點選「新增/移除 Windows 元件」。



步驟二：在 Windows 元件精靈對話窗中，選擇「Networking Services」，然後再點選「詳細資料」按鈕。



步驟三：勾選「Domain Name System (DNS)」，再點選「確定」。



步驟四：回到 Windows 元件精靈對話窗中，點選「下一步」，系統便會開始進行安裝 DNS 的動作。

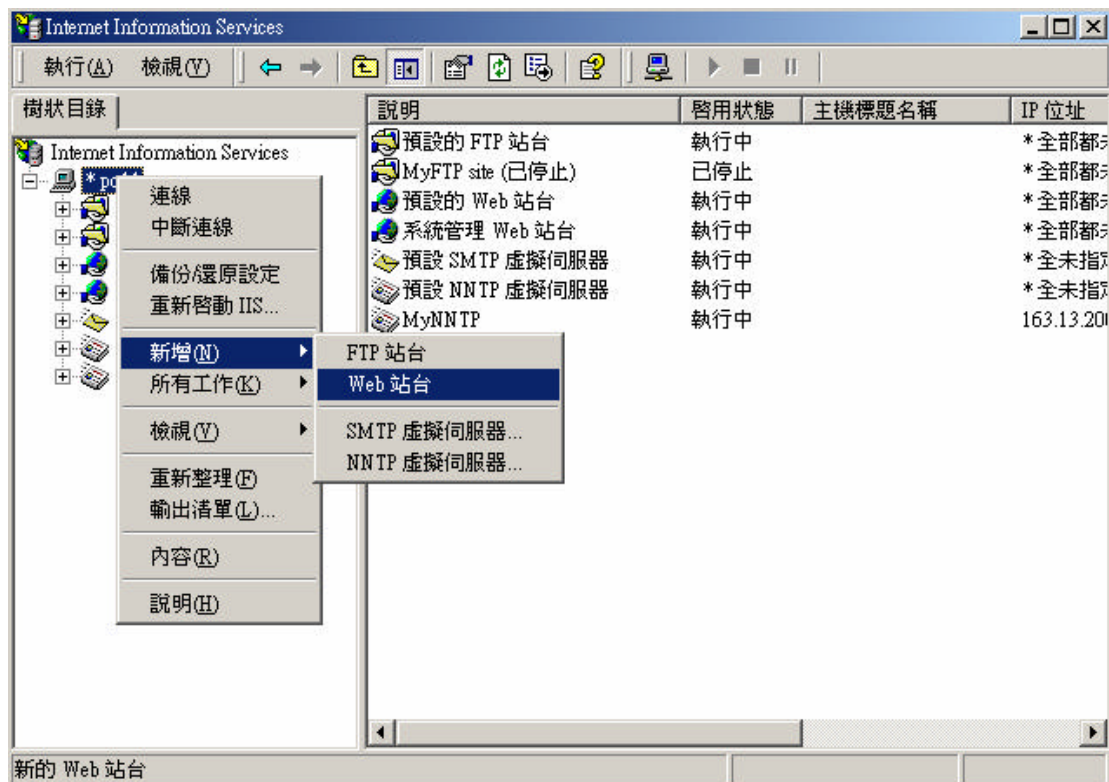
3.3 Web site

3.3.1 簡介:

從事電子商務必定需要能有自己的專屬 Web 站台，IIS 提供 Web 站台的架設，配合微軟的 FrontPage 可使使用者輕易架設出有自己風格的站台，並提供使用者更容易管理好站台的設定。

3.3.2 安裝

步驟一：在管理工具中，本機電腦按滑鼠的右鍵，點選「新增」 「Web 站台」，接下來點選「下一步」略過歡迎使用畫面。



步驟二：輸入顯示在管理工具中的 Web 站台名稱。

Web 站台新增精靈

Web 站台說明
描述 Web 站台來協助管理員識別它。

請輸入 Web 站台說明。

說明(D):
MyWeb

< 上一步(B) 下一步(N) > 取消

步驟三：輸入 Web 站台的 IP 位址，若選擇預設的「全未指定」，則此 Web 站台對應到該伺服器主機中所有的 IP 位址(可能不只一個)，因此，連結任一個 IP 位址，都可以連到 Web 站台。

Web 站台新增精靈

IP 位址及連接埠設定
請指定新 Web 站台的 IP 位址及連接埠設定。

請輸入此 Web 站台的 IP 位址(E):
(全未指定)

這個 Web 站台應該使用的 TCP 連接埠: (預設值: 80)(T)
80

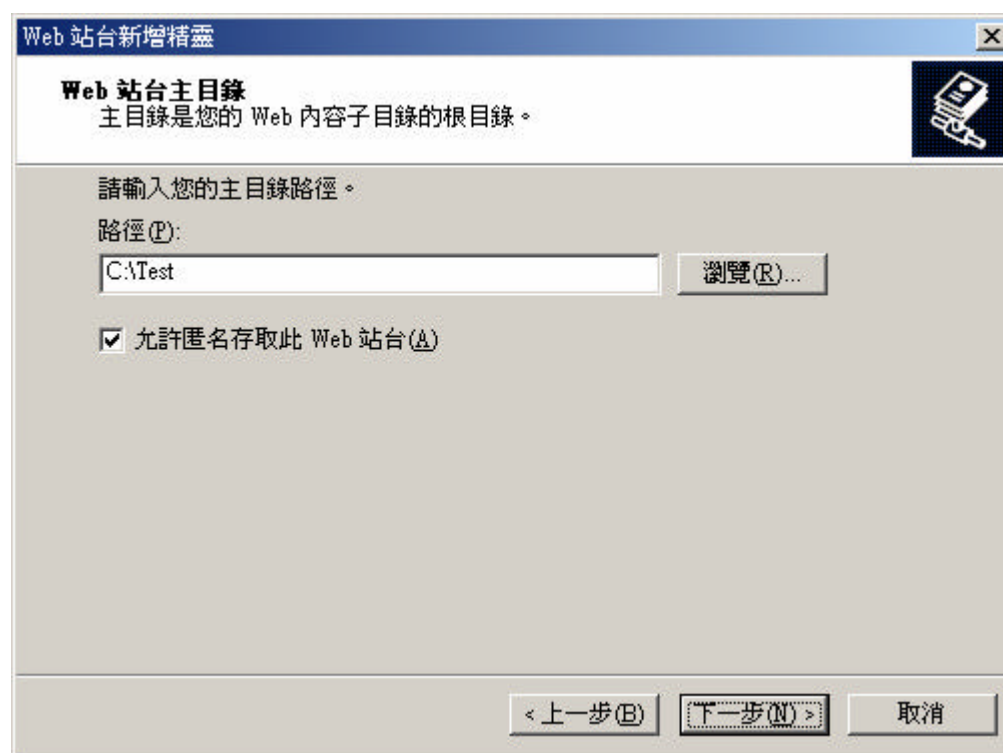
此站台的主機標題: (預設值: 無)(H)

這個 Web 站台應該使用的 SSL 連接埠: (預設值: 443)(L)

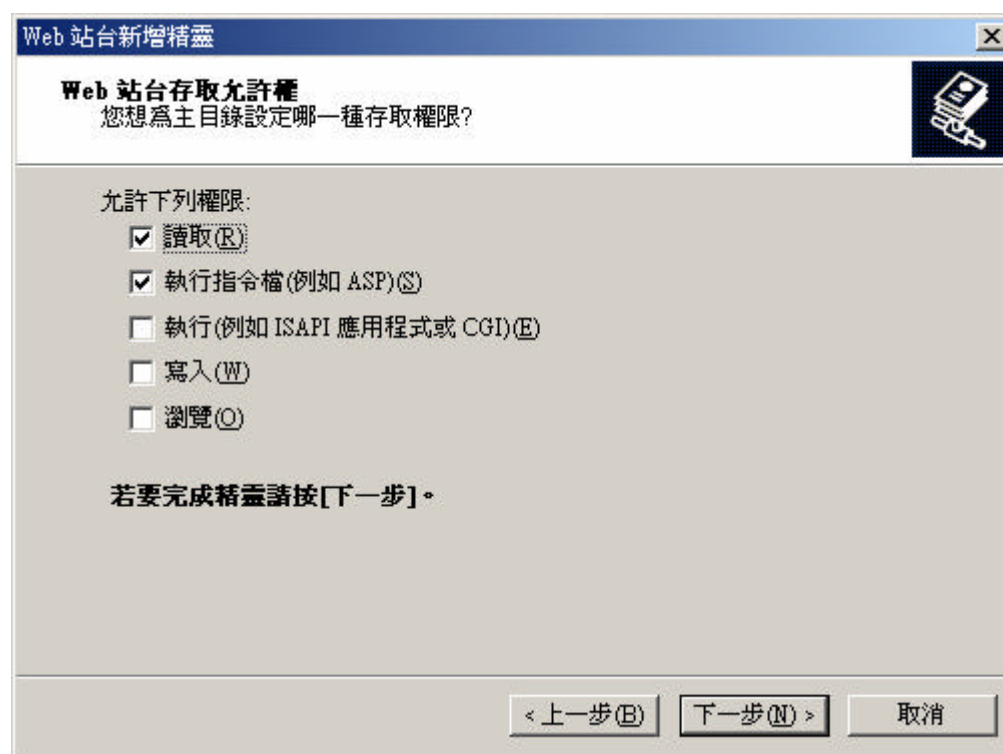
若需進一步資訊，請參閱 IIS 說明文件。

< 上一步(B) 下一步(N) > 取消

步驟四：輸入 Web 站台的根目錄。



步驟五：設定 Web 站台根目錄的存取權限。點選「下一步」後，即可完成新增 Web 站台的動作。



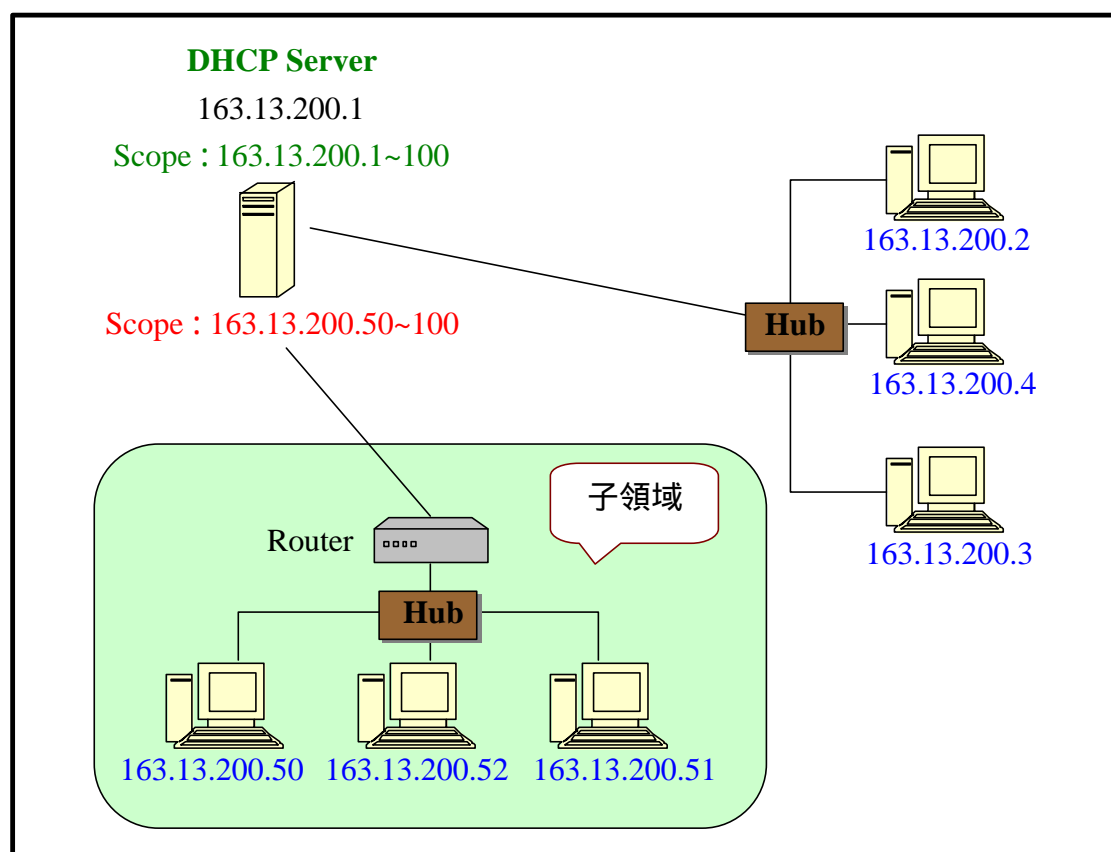
步驟六：在新增 Web 站台完成後，由於 TCP 通訊埠的重複，須先停止「預設的 Web 站台」，再將所編輯完成的網頁檔案移至 Web 站台的根目錄中即可。

(IIS 的 Web 站台預設是自動開啟 Default.htm，而不是一般常用的 Index.htm，所以首頁的檔名要改成 Default.htm)

3.4 DHCP (Dynamic Host Configuration Protocol)

3.4.1 簡介

在一個 TCP/IP 的網路中，每一台電腦都需要一個 IP 位址，為了簡化管理者的工作，制定了 DHCP 這個開放性的標準協定。管理者透過一台 DHCP 伺服器，給定 IP 範圍、參數及規則，即可輕鬆的管理整個 TCP/IP 網路的 IP 位址，相較於一個一個的輸入，不僅有較高的效率，也減少了因為輸入錯誤而所引起的問題。



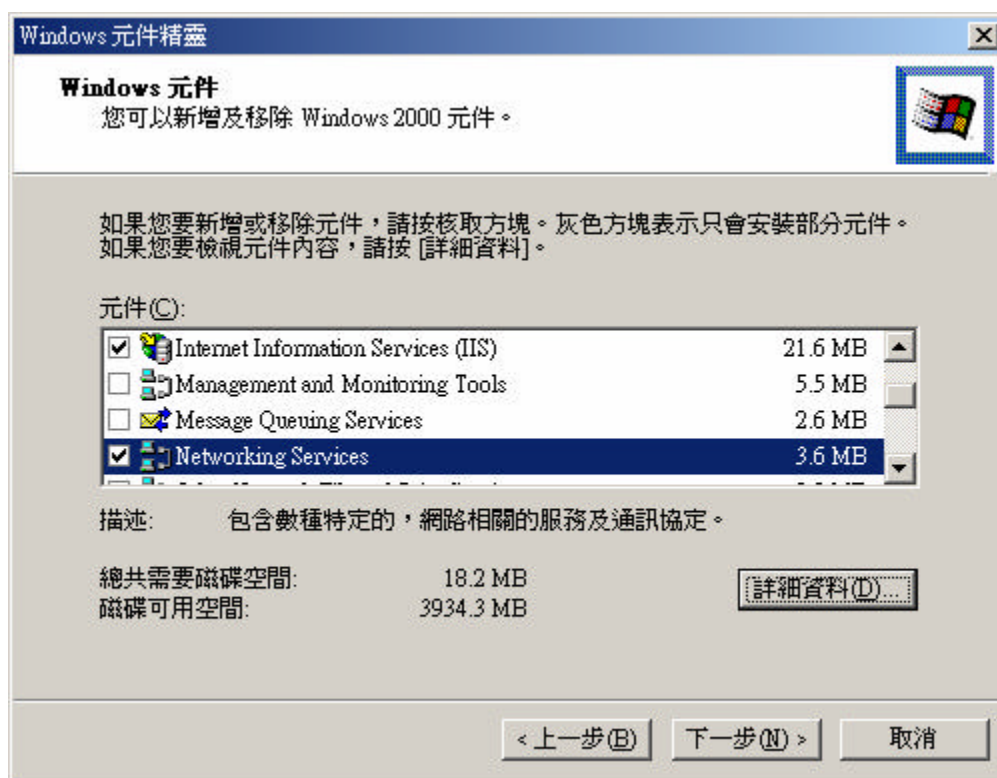
用戶端開機後，會透過廣播 (Broadcast) 向 DHCP Server 要求租用 (Lease) 一個 IP 位址，DHCP Server 收到後會傳回一個尚未租用的 IP 位址以及其他相關資訊 (如：子網路遮罩、DNS 伺服器位址、預設閘道位址等) 給用戶端。由於是透過廣播，在子網路中就必須透過 DHCP 中繼代理者 (DHCP Relay Agent)，DHCP Server 才有辦法接收到用戶端所傳送出來的廣播封包，而 DHCP 中繼代理者的角色一般都是由支援 (Bootstrap Protocol, BOOTP) 的 Router 來扮演。

3.4.2 安裝

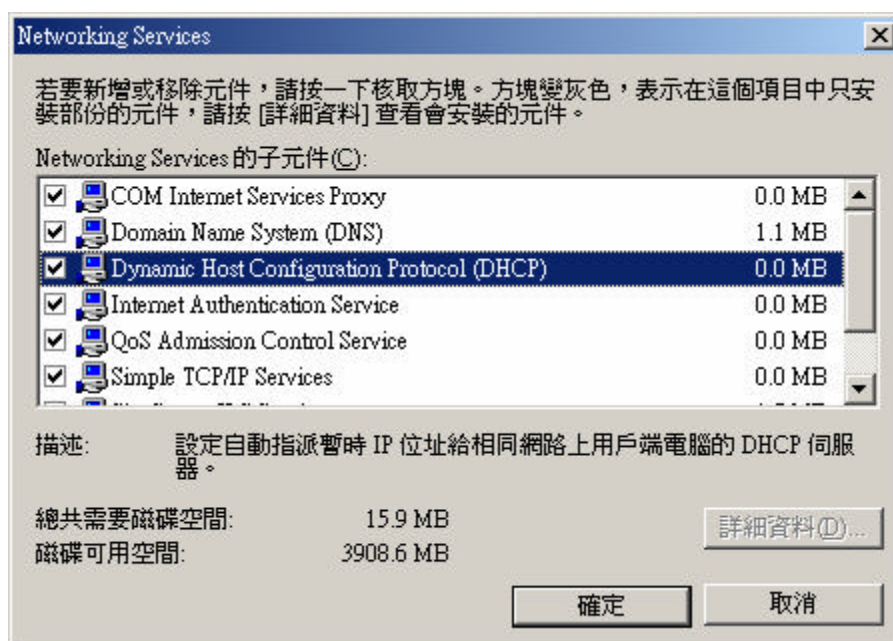
步驟一：由「控制台」選擇「新增/移除程式」，在新增/移除程式對話窗左邊點選「新增/移除 Windows 元件」。



步驟二：在 Windows 元件精靈對話窗中，選擇「Networking Services」，然後再點選「詳細資料」按鈕。



步驟三：勾選「Dynamic Host Configuration Protocol (DHCP)」，再點選「確定」。



步驟四：回到 Windows 元件精靈對話窗中，點選「下一步」，系統便會開始進行安裝 DHCP 的動作。

3.5 SMTP(Simple Mail Transfer Protocol)

3.5.1 簡介

「SMTP」(Simple Mail Transfer Protocol)是郵件傳遞通訊協定，支援這個通訊協定的伺服器能夠將郵件工具送來的郵件寄送到 Internet 或網路上目的地的 POP3 伺服器。

IIS 提供 SMTP 虛擬伺服器，就算網路上沒有安裝郵件伺服器也一樣可以在 Web 站台的網頁提供寄送郵件的服務。

SMTP 虛擬伺服器只是虛擬的伺服器，因為他並不是一個真正的郵件伺服器，只能算是一個電子郵件的路由匝道，以便 Web 站台能夠透過 SMTP 虛擬伺服器將網頁傳送的郵件轉送到真正的郵件伺服器。

微軟真正的郵件伺服器產品為 Exchange Server 2000，SMTP 虛擬服務使用「IMS」(Internet Mail Service)連接到 Exchange Server，虛擬伺服器傳送的郵件屬於 Internet 的電子郵件，就傳送到 Internet，如果屬於內部 Intranet 的電子郵件，就直接存入 Exchange Server 使用者的郵件資料夾。

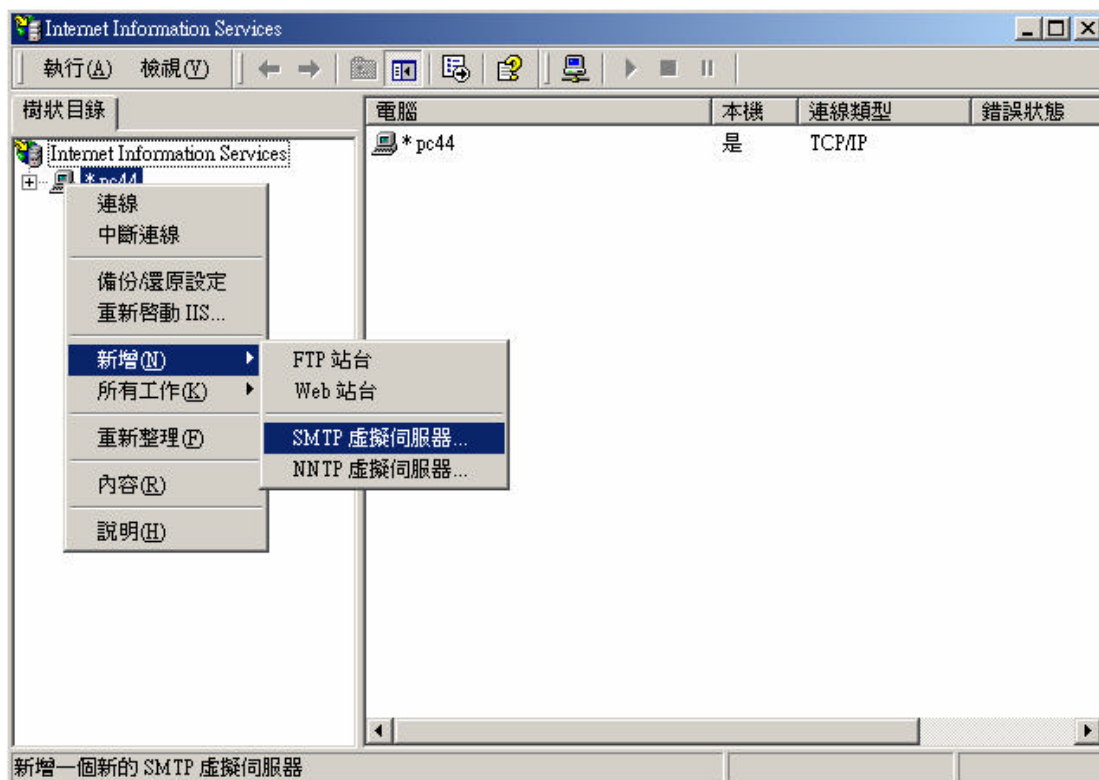
所以 Outlook Express 等郵件工具能夠寄送電子郵件到 SMTP 虛擬伺服器，但是無法使用 Outlook Express 透過 SMTP 虛擬伺服器取得電子郵件，而是由真正的郵件伺服器 Exchange Server 接收電子郵件，SMTP 虛擬伺服器扮演的角色有兩種：

- 1、轉寄郵件的匝道---自動將網頁的郵件轉寄到郵件伺服器，獲釋設定「智慧主機」或「遠端網域」將郵件轉送到指定的目的地。
- 2、虛擬的郵件伺服器---如果設定 SMTP 虛擬伺服器不轉送郵件到真正的郵件伺服器，所有寄到網域的電子郵件，不論使用者是誰都會儲存在同一個 SMTP 虛擬伺服器的資料夾，角色上有如一個真正的郵件伺服器。

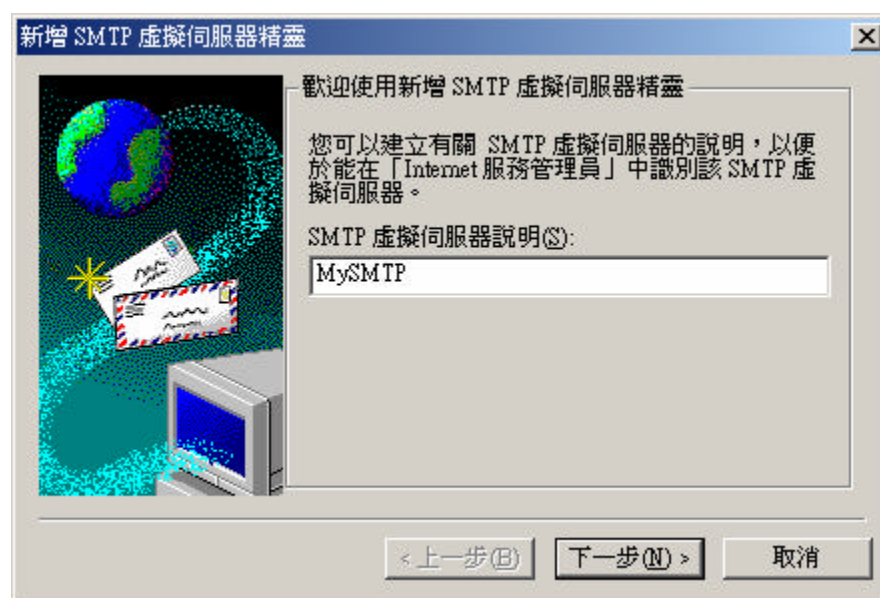
當設定 SMTP 虛擬伺服器收取郵件時，因為郵件並沒有轉送到真正的郵件伺服器，Outlook Express 等郵件工具並沒有辦法在 SMTP 虛擬伺服器收取郵件。

3.5.2 安裝

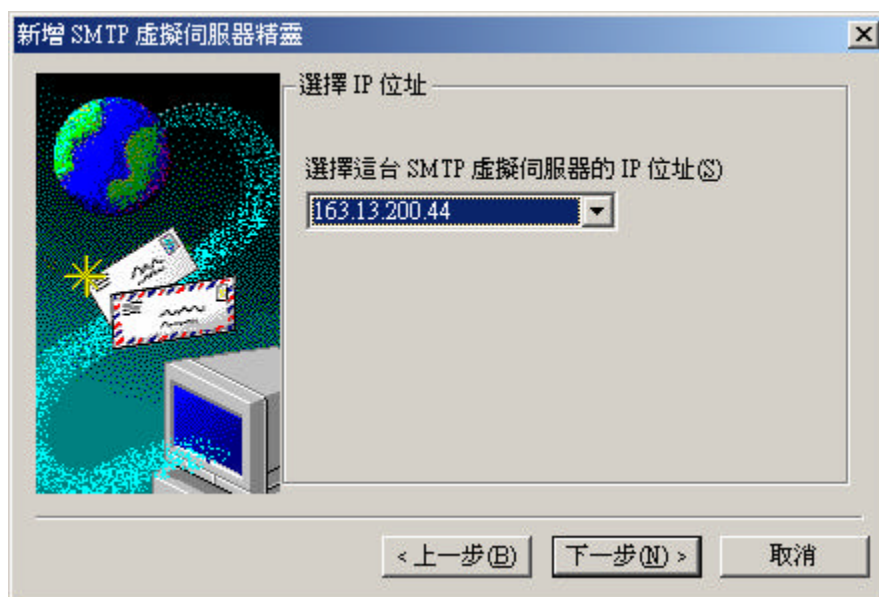
步驟一：在管理工具中，本機電腦按滑鼠的右鍵，點選「新增」 「SMTP 虛擬伺服器」，並點選「下一步」略過歡迎使用畫面。



步驟二：輸入顯示在管理工具中的 SMTP 虛擬伺服器名稱。



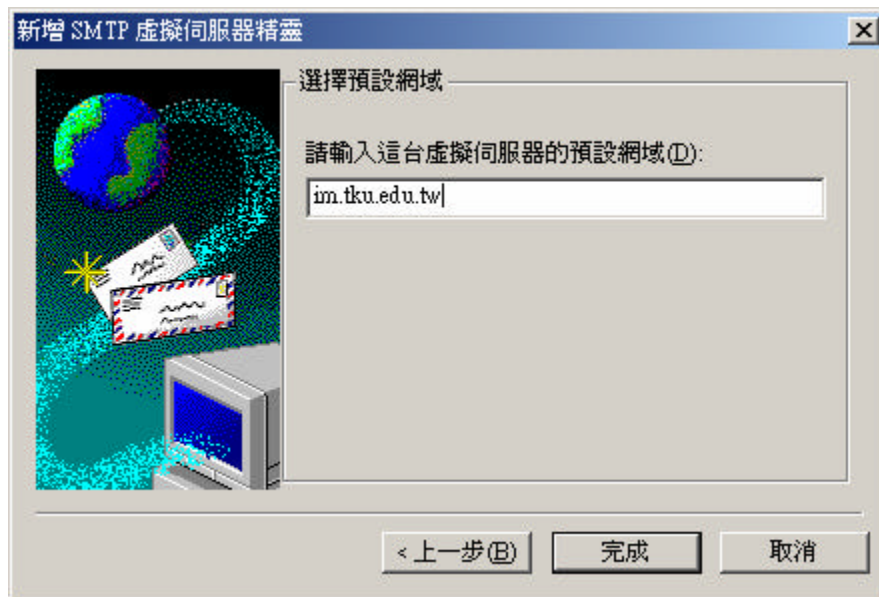
步驟三：指定 SMTP 虛擬伺服器的 IP 位址



步驟四：輸入 SMTP 主目錄的路徑



步驟五：輸入 SMTP 虛擬伺服器的預設網域後點選「完成」即可完成安裝。



3.6.FTP (File Transfer Protocol)

3.6.1 簡介

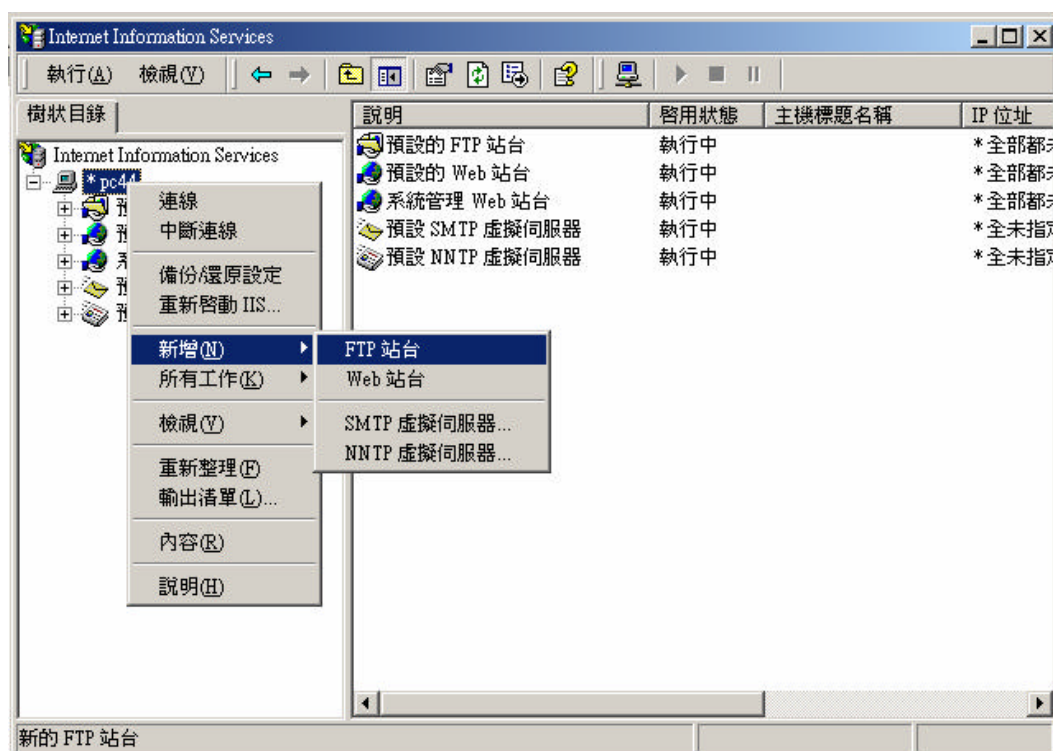
「FTP」(File Transfer Protocol) 是一種檔案傳輸的通訊協定，能夠再不同的電腦和主機間交換檔案，FTP 站台就是使用 FTP 通訊協定所建立的檔案交流園地，我們稱為 FTP 伺服器。

目前 Internet 眾多的共享或免費軟體等檔案，大多是存放在 FTP 站台，允許使用者直接以匿名 (Anonymous) 和電子郵件地址的密碼存取檔案資源，我們稱此為「匿名 FTP」(Anonymous FTP)。

現今有許多專門架設 FTP 站台的軟體，而 IIS 中的 FTP 服務亦有相同之功能，其可搭配 Web 站台建立遠端更新網頁檔案的機制，只需透過 FTP 工具就可以在連線 Internet 的電腦維護網站的內容。

3.6.2 安裝

步驟一：在管理工具中，本機電腦按滑鼠的右鍵，點選「新增」 「FTP 站台」，並點選「下一步」略過歡迎使用畫面。



步驟二：輸入顯示在管理工具中的 FTP 站台名稱。

FTP 站台新增精靈

FTP 站台說明
描述 FTP 站台來協助管理員識別它。

請輸入 FTP 站台說明。

說明(D):
MyFTP site

< 上一步(B) 下一步(N) > 取消

步驟三：輸入 FTP 站台的 IP 位址，若選擇預設的「全未指定」，則此 FTP 站台對應到該伺服器主機中所有的 IP 位址(可能不只一個)，因此，連結任一個 IP 位址，都可以連到 FTP 站台。

FTP 站台新增精靈

IP 位址及連接埠設定
您必須指定 FTP 站台的 IP 位址及連接埠設定。

請輸入此 FTP 站台的 IP 位址。

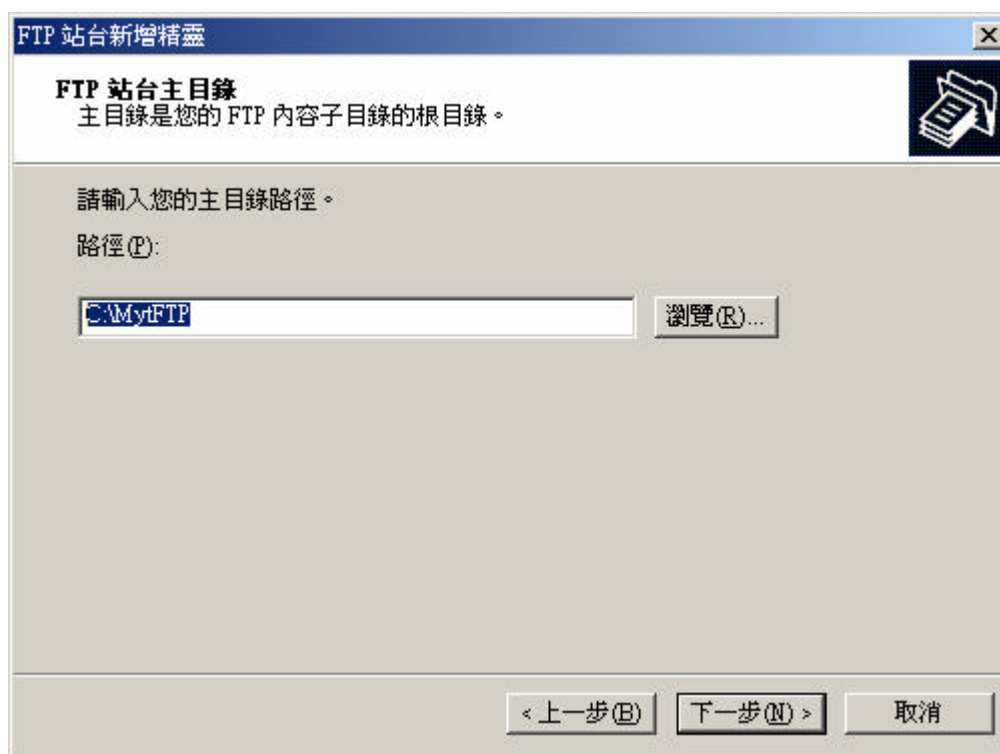
IP 位址(I):
(全未指定)

請為此 FTP 站台輸入 TCP 連接埠: (預設值: 21)

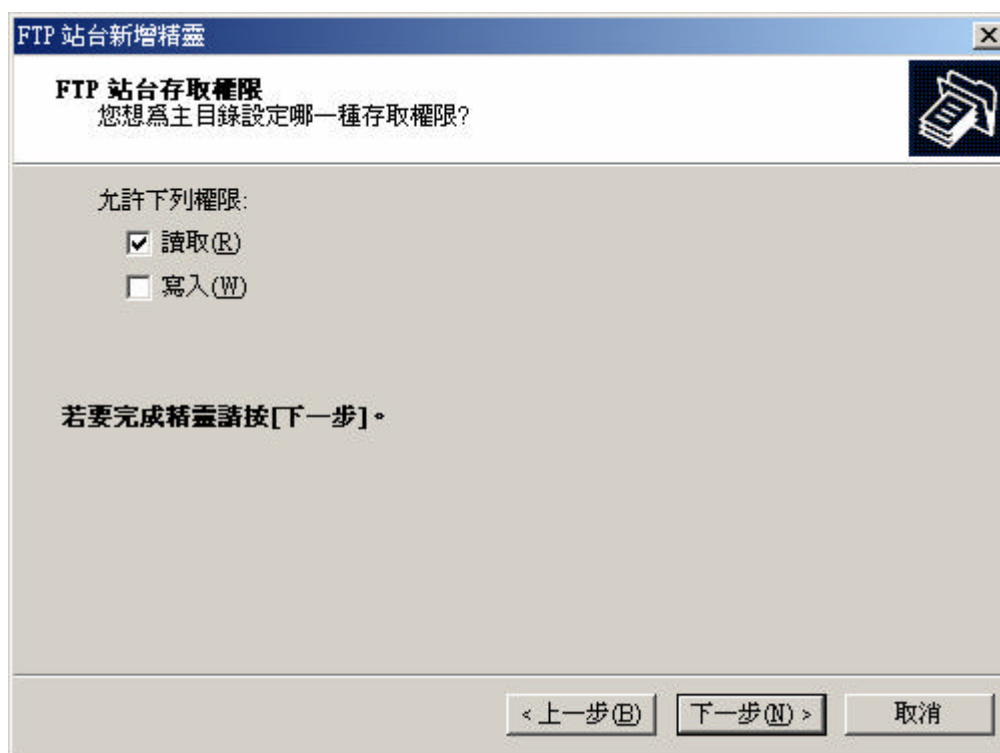
TCP 連接埠(T):
21

< 上一步(B) 下一步(N) > 取消

步驟四：輸入 FTP 站台的根目錄。



步驟五：設定 FTP 站台主目錄的存取權限，點選「下一步」後即可完成 FTP 站台的新增動作。



3.7 NNTP (Network News Transport Protocol)

3.7.1 簡介

「NNTP」(Network News Transport Protocol) 虛擬伺服器為 IIS 預設的服務，這是一個完整功能的新聞伺服器，可以在 Internet / Intranet 建立網路論壇 Usenet News。

News 網路論壇是一個動態的資料庫，在這個資料庫內包含來自世界各地的資訊，任何人都可以透過網路在 News 交換訊息。而這些資料的來源都是網友所提供，News 讀者可以在其上閱讀或提供資料，因為它是一個開放的討論園地，資料內容包羅萬象，舉凡藝術、科學、體育、電腦、工程等在 News 上都有專屬的討論園地，我們稱為討論組或「新聞群組」(Newsgroup)。

News 的新聞群組類似網域名稱，在最上層分為下列幾大類，如下表所示：

名稱	說明
comp	與電腦知識相關的新聞群組
sci	科學知識方面的新聞群組
soc	社會方面的新聞群組
talk	屬於針鋒相對的爭議話題
news	關於 Usenet 本身的新聞群組
res	娛樂類的新聞群組
misc	其他的新聞群組
tw	台灣地區的新聞群組

在上表的大分類後，依不同的主題再區分為更細的新聞群組以「.」分隔，如同網域名稱，如下所示：

```
tw.bbs.comp.unix  
tw.bbs.comp.mswindows
```

NNTP 通訊協定屬於一種主從架構的通訊協定，server 端的伺服器提供新聞服務，在 IIS 是 NNTP 虛擬伺服器，client 端使用電子郵件和新聞工具接收新聞，例如：Outlook Express，使用的埠號為 119。

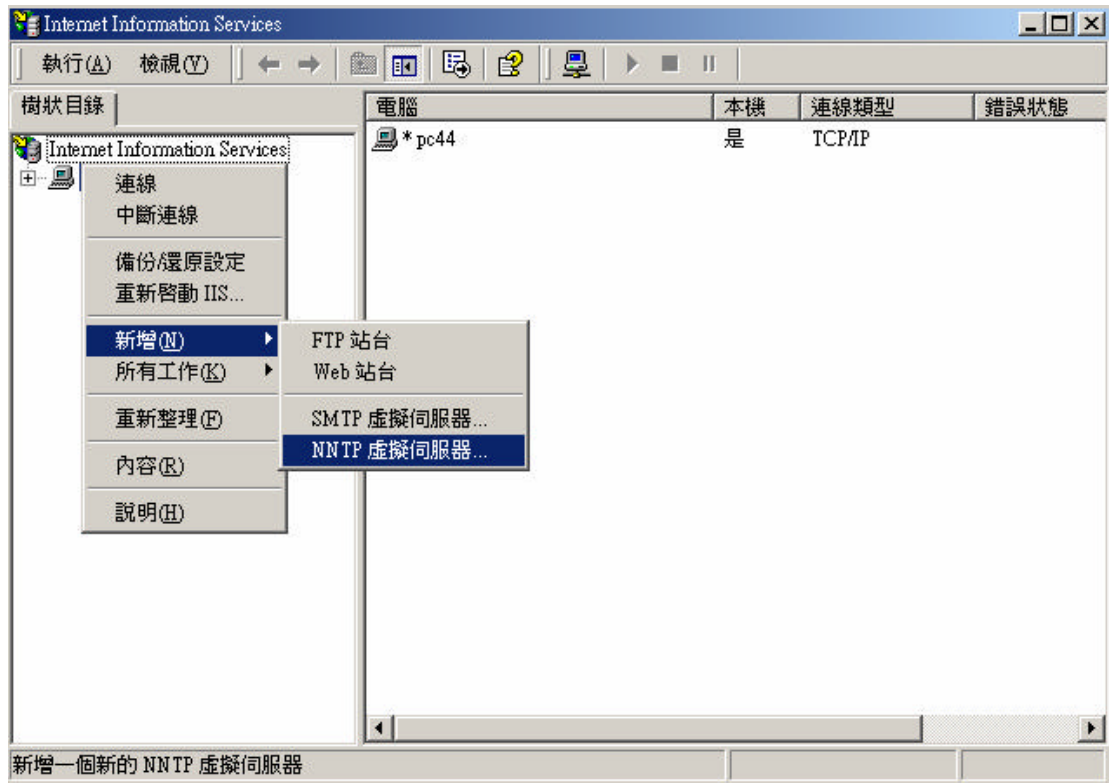
IIS 的 NNTP 虛擬伺服器相容於其他的 NNTP 新聞伺服器和 client 端程式，例如：Outlook Express，只需在 Internet 服務管理員設定和新增好新聞群組，NNTP 虛擬伺服器的新聞論壇馬上可運作，不過不能從其他新聞論壇取得新聞，他屬於獨立的新聞論壇服務。

在使用上只需建立新聞伺服器的連線，client 端的工具就能夠張貼新聞到指定的新聞群組，再建立 IIS 的 NNTP 虛擬伺服器時，我們需要的工作，如下所示：

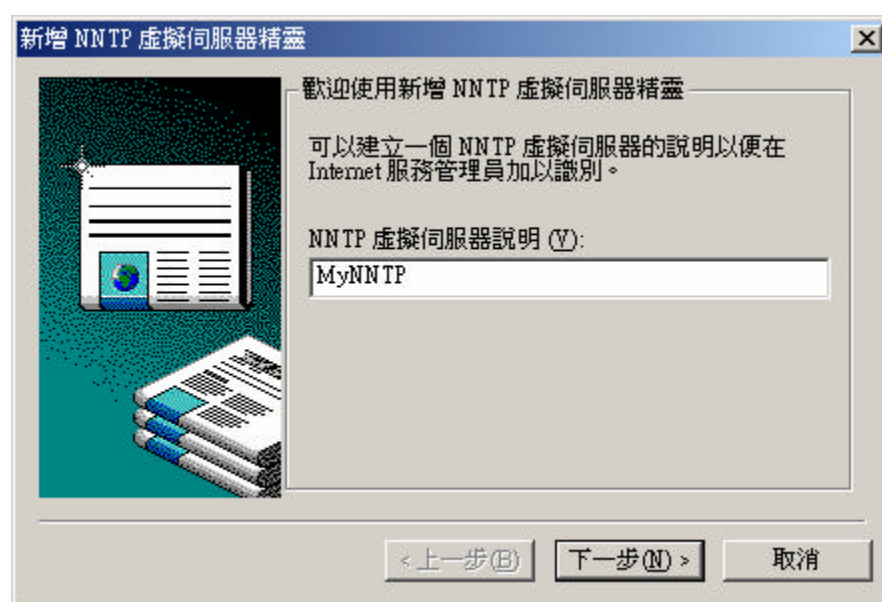
- 1、新增網路論壇的新聞群組：NNTP 虛擬伺服器並沒有辦法取得其他的新聞伺服器的新聞，新要自己建立所需的新聞群組。
- 2、設定新聞到期原則：因為新聞文章的篇數不可能無限制的成長，對於比較就或文章佔用太的的硬碟空間，我們需要設定新聞管理規則有效的管理新聞論壇的新聞。

3.7.2 安裝

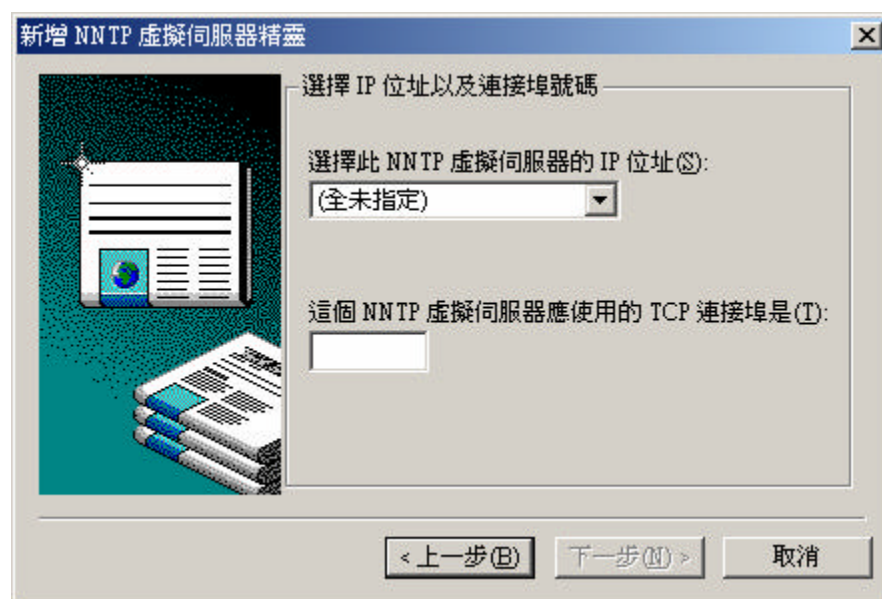
步驟一：在管理工具中，本機電腦按滑鼠的右鍵，點選「新增」 「NNTP 虛擬伺服器」，並點選「下一步」略過歡迎使用畫面。



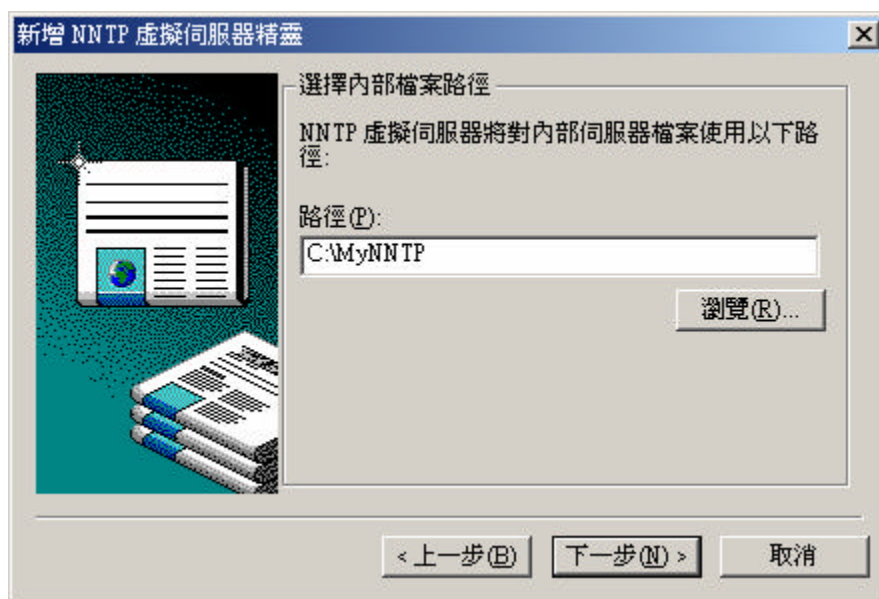
步驟二：



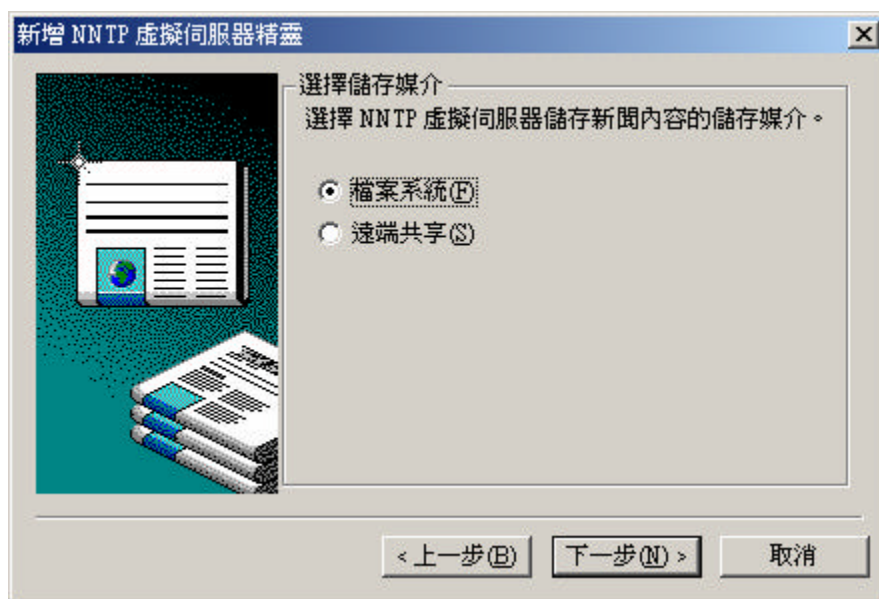
步驟三：



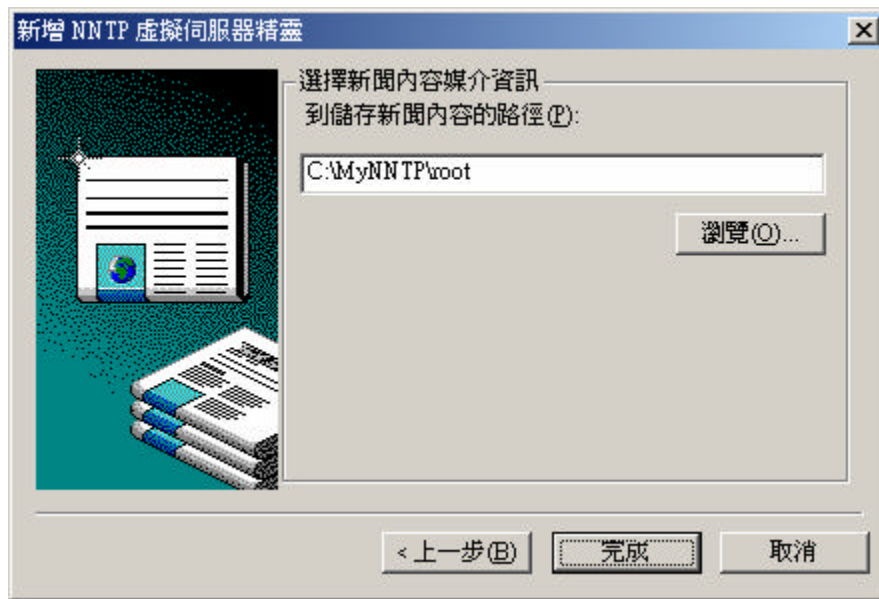
步驟四：



步驟五：



步驟六：



3.8 SQL Server 2000

3.8.1 簡介:

3.8.1.1 資料庫 (Database) :

資料庫是一個儲存空間，是一個電子化的資料檔案櫃，它最主要用來儲存資料表和資料。運用資料庫比用人工處理更具有簡潔性、迅速、減少人力介入、即時性及集中控制等好處。

資料庫的好處有：

- 1、避免資料的重複及不一致性—資料庫透過資料集中化做控管可降低資料的重複性，另外，資料庫具有「傳播更新」(propagating updates)的特性，當一個資料更新時，會茲棟地更新其他重複、相同的資料，可有效降低不一致的問題；利用「正規化」(Normalization)減少資料本身的重複性，也可有效降低發生不一致的機率。
- 2、資料可做分享—不同的應用系統可透過如 SQL (Structured Query Language) 或 DML (Data Manipulation Language) 在同一時間來存取、共享同一份資料。
- 3、妥善的安全和整合性控制—DBA (Database Administrator) 透過正確通道來存取限制需求；另外，在集中控制下，可以定義資料更新時完整性檢查。
- 4、應用程式發展的生產力可提高—一般的 DBMS (Database Management System) 提供許多發展工具，如 Oracle Developer 2000 等。
- 5、資料可具有獨立性—應用不必依賴於任一特定儲存結構或存取策略，即將資料描述和使用它的程式加以分離。

資料庫系統的三大層次：

- 1、外部層 (External Level) --
 - 、個別使用者的觀點，即每位使用者透過資料定義語言來定義個人所需的外部觀點，使用者在使用上只看到概念層的一部份，其他部分通常被遮掉而不能存取。
 - 、外部層使用者對資料存取大多以查詢動作為主。
 - 、關連式資料庫系統對應外部層，係提供以架在 Base Table 上的 View 來呈現給終端使用者。

- 2、概念層 (Conceptual Level) --
 - 、定義整個資料庫的所有資料內容，系統以表格形式呈現給使用者，以及負責安全性及整合性的檢查。
 - 、概念層是真正存放 Base Table 的，定義中通常包含整合性限制條件，但不考慮儲存結構及存取方法，如儲存記錄順序、索引方式等細節。
- 3、內部層 (Internal Level) --
 - 、定義各種不同型態的記錄、索引方法、記錄實際儲存的順序及儲存欄位的表示法。
 - 、內部層不處理實體記錄，也不處理實際裝置，如磁卷、磁柱、磁軌及磁區大小等。

以上三層各層間的相對應定義都存放在系統目錄下，由 DBA 負責管理及維護。

資料獨立性 (Data Independent)：簡單來說即在各層次間，某一層次修改不影響其他層次。

資料的獨立性可分為：1、邏輯獨立—當概念層結構改變時，不影響外部層 (應用程式)。

2、實體獨立—當內部層結構改變時，不影響外部層 (應用程式)。

(即應用程式對存取技術和儲存結構改變具有免疫力)

資料獨立性的理由有二：

- 1、不同的應用程式對於相同資料，有時需要不同的 View—當使用者對於使用介面或輸出有不同需求時，修改外部層的應用程式，並不影響內部層的存取方式及儲存結構。
- 2、當使用者對資料需求改變時，DBA 必須能自由地改變資料結構及存取技術，而不必修改現有的應用程式。

依資料庫管理系統所用的資料模式來分，資料庫模式主要可分為：階層式 (Hierarchical)、網狀式 (Network)、關聯式 (Relational) 三大模式。

- 階層式資料模式 (Hierarchical Data Mode)：
 - 早期普遍流行的資料模式。
 - 資料結構關聯由資料樹組成。
 - 不易使用。
- 網狀式 (Network Data Mode)
 - 階層式模式的延申。
 - 不易使用
- 關聯式 (Relational Data Mode)
 - 目前應用最普遍的模式

- 以 2 維的資料表格表示資料關係模式。
- 資料表格間以『資料項目』互相關聯
- 較易使用

3.8.1.2 SQL 語言介紹：

SQL 全名是“結構化查詢語言(Structured Query Language)”，最早的是 IBM 的聖約瑟研究實驗室為其關聯式資料庫管理系統 SYSTEM R 開發的一種查詢語言，它的前身是 SQUARE 語言。SQL 語言結構簡潔，功能強大，簡單易學，所以自從 IBM 公司 1981 年推出以來，SQL 語言，得到了廣泛的應用。如今無論是像 Oracle ,Sybase, Informix, SQL server 這些大型的資料庫管理系統，還是像 Visual Foxpro, PowerBuilder 這些微機上常用的資料庫開發系統，都支援 SQL 語言作為查詢語言。

Structured Query Language 包含 4 個部分：

資料查詢語言 DQL-Data Query Language SELECT

資料操縱語言 DML-Data Manipulation Language INSERT, UPDATE, DELETE

資料定義語言 DDL-Data Definition Language CREATE, ALTER, DROP

資料控制語言 DCL-Data Control Language COMMIT WORK, ROLLBACK WORK

SQL 的歷史

在 70 年代初，E.E.Codd 首先提出了關係模型。70 年代中期，IBM 公司在研製 SYSTEM R 關聯式資料庫管理系統中研製了 SQL 語言，最早的 SQL 語言(叫 SEQUEL2)是在 1976 年 11 月的 IBM Journal of R&D 上公佈的。

1979 年 ORACLE 公司首先提供商用的 SQL，IBM 公司在 DB2 和 SQL/DS 資料庫系統中也實現了 SQL。

1986 年 10 月，美國 ANSI 採用 SQL 作為關聯資料庫管理系統的標準語言(ANSI X3.135-1986)，後為國際標準化組織(ISO)採納為國際標準。

1989 年，美國 ANSI 採納在 ANSI X3.135-1989 報告中定義的關聯資料庫管理系統的 SQL 標準語言，稱為 ANSI SQL 89，該標準替代 ANSI X3.135-1986 版本。

該標準為下列組織所採納：

國際標準化組織(ISO)，為 ISO 9075-1989 報告“Database Language SQL With Integrity Enhancement”

美國聯邦政府，發佈在 The Federal Information Processing Standard Publication(FIPS PUB)127

目前，所有主要的關聯資料庫管理系統支援某些形式的 SQL 語言，大部分資料庫打算遵守 ANSI SQL89 標準。

SQL 的優點：

SQL 廣泛地被採用正說明了它的優點。它使全部用戶，包括應用程式師、DBA 管理員和終端用戶受益非淺。

(1) 非程序式語言--SQL 是一個非程序式的語言，因為它一次處理一個記錄，對資料提供自動導航。SQL 允許用戶在高層的資料結構上工作，而不對單個記錄進行操作，可操作記錄集。所有 SQL 語句接受集合作為輸入，返回集合作為輸出。SQL 的集合特性允許一條 SQL 語句的結果作為另一條 SQL 語句的輸入。SQL 不要求用戶指定對資料的存放方法。這種特性使用戶更易集中精力於要得到的結果。所有 SQL 語句使用查詢優化器，它是 RDBMS 的一部分，由它決定對指定資料存取的最快速度的手段。查詢優化器知道存在什麼索引，哪兒使用合適，而用戶從不需要知道表是否有索引，表有什麼類型的索引。

(2) 統一的語言--SQL 可用於所有用戶的 DB 活動模型，包括系統管理員、資料庫管理員、應用程式師、決策支援系統人員及許多其他類型的終端用戶。基本的 SQL 命令只需很少時間就能學會，最高級的命令在幾天內便可掌握。SQL 為許多工提供了命令，包括：

- 查詢資料

- 在表中插入、修改和刪除記錄

- 建立、修改和刪除資料物件

- 控制對資料和資料物件的存取

- 保證資料庫一致性和完整性

以前的資料庫管理系統為上述各類操作提供單獨的語言，而 SQL 將全部任務統一在一種語言中。

(3) 是所有關聯資料庫的公共語言--由於所有主要的關聯資料庫管理系統都支援 SQL 語言，用戶可將使用 SQL 的技能從一個 RDBMS 轉到另一個。所有用 SQL 編寫的程式都是可以移植的。

SQL 語言與英文有部分極為相似，以下簡單的介紹 SQL 語言的語法。

Select：應該是最常使用的敘述，它的目的是將符合條件的資料從資料庫裡面選取 (select)，故得名。它的語法如下：

```
select [column name 1], [column name 2], ... from [table name] where  
[expression] order by [column name] [desc|inasc]
```

其中 column name 是資料庫的表格 (table) 內的欄位名稱。from 設定要從哪個表格取出資料。而 where 子句則是被選取的資料必須符合的條件，也可以省略不寫。order by 子句則是說明依照哪個欄位來排序，desc 表示降冪排序，而 inasc 則反之。order by 子句也可以省略不寫。舉例來說，假如我有一個表格叫做 grade，其中 name 這個欄位存放學生的姓名，而 score 這個欄位存放學生的成績，則：

```
select name, score from grade
```

這樣會選取資料表裡面所有的紀錄。

```
select name, score from grade where score > 90
```

這樣會選取資料表裡面，「成績」高於 90 分的紀錄。

```
select name, score from grade where score > 60 order by score desc
```

這樣會選取資料表裡面成績及格（大於 60 分）的紀錄，並且以成績大小排序輸出，成績高的在前（降冪排序）。

```
select max(6, 10)
```

這樣會輸出 6, 10 兩數中最大的數。本例為 10。

```
select * from score where max(score)
```

這樣會選取資料表裡面成績最高的那個人。在 column 內的 * 表示資料表裡面所有的欄位名稱。

where 子句內有內建數種函數，如 max(), min() 等等。可查詢資料庫的使用手冊來了解如何使用。

insert 敘述：

顧名思義，insert 敘述就是用來插入一筆資料到資料庫裡面的指令。它的語法如下：

```
insert into [table name] [column name(s) ...] values (value1, value2, ...)
```

其中如果值是字串，要記得用引號括起來。如果省略 column name 的話，那麼值的順序就是按照當初建立資料表的順序。同樣以 grade 這個表格為例：

```
insert into grade name, score values ('吳此仁', 88)
```

表示在 grade 資料表內插入一筆資料，姓名為吳此仁，成績 88 分。

```
insert into grade name values ('小蘭')
```

表示在 grade 資料表內插入一筆資料，姓名為小蘭。

update 敘述：

update 敘述用來更新現有並且符合指定條件的資料。它的語法如下：

```
update [table name] set [column name]=[value], [column name]=[value], ....  
where [expression]
```

舉例來說：（還是以 grade 這個表格為例）

```
update grade set score = 0
```

這樣會把所有人的成績都設成 0。

```
update grade set score = 80 where name = '小蘭'
```

會把小蘭的成績設為 80。

delete 敘述：

從資料表裡面刪除一筆或多筆資料。語法如下：

```
delete from [table name] where [expression]
```

舉例來說：

```
delete from grade
```

這樣會把 grade 資料表清空。

```
delete from grade where score < 60
```

這樣會把成績不及格的資料清除。

3. 8.1.3 SQL Server 2000 簡介：

SQL server 是一個關連式資料庫 (relational database)，它除了支援傳統關連式資料庫元件 (如資料庫、資料表) 和特性 (如資料表的 join) 外，另外也支援現今關連式資料庫常用的物件如預存程序 (stored procedure) 檢視 (view) 等。另外從它的名字中我們也可以知道，它支援目前關連式資料庫必定要支援的標準查詢語言 - SQL (Structured Query Language)。SQL server 另外一項重要的特點是它支援資料庫複寫 (replication) 的功能，也就是當你在一資料庫上執行異動時，可以將其異動結果傳至遠端 SQL server 相同的資料庫上，讓兩邊資料庫的資料保持同步。

SQL server 在 client-server 架構中是扮演 server 端的角色。其主要職責是儲存資料和提供一套方法來管理這些資料，並且應付來自 client 的連線和資料存取需求。由於 SQL server 是扮演 server 端的角色，是資料的提供者，所以在 SQL server 內看不到類似 GUI 設計的功能，也就是說 SQL server 並不提供工具讓你設計一個輸入或查詢的操作畫面，另外也看不到和報表設計有關的工具，因為對 SQL server 所扮演的角色而言，這部是它的職責所在，這些使用者操作介面的設計工作是 client 端 (如 VB、Delphi) 的事情。

通常我們將 SQL server 稱為一資料庫的引擎，因為他是整個資料庫應用系統中的核心，但是光只有這個資料庫引擎還不能構成一個完整的應用系統，還需要利

用前端開發工具如 VB、Delphi 等產品發展出使用者介面才能稱唯一完整的資料庫應用系統。

SQL server 2000 是一個由微軟 (Microsoft) 公司所出版的關聯式資料庫管理及分析系統其可分為企業版、標準版、個人版、開發版、WindowsCE 版。各版本差異如下：

- SQL Server 2000 企業版--是可適用於各種公司行號的完整 SQL Server 版本。它可針對具關鍵性的商業領域與網際網路規劃，提供最先進的延展性與可靠性的功能需求，包括分散式分割檢視表、記錄傳送與加強的容錯移轉叢集功能。它能充分發揮高階硬體的效能，最高可支援到 32 顆 CPU 與 64 GB 記憶體(RAM)。SQL Server 2000 企業版包括了先進的分析功能(OLAP)，用來處理具有大量維度的最大型 cubes。
- SQL Server 2000 標準版--對於不需要先進的延展性與可靠性功能，或 SQL Server 2000 企業版中一些更先進的分析功能的中小型企業，此版本是最實惠的選擇。標準版也可以用在最高可達 4CPU 及 2 GB 記憶體(RAM) 的對稱性多重處理系統。
- SQL Server 2000 個人版--包括了全套完整的管理工具與標準版中大部分的功能，但是是為個人使用專門設計的版本。除了在微軟的伺服器作業系統中執行外，個人版也可以在非伺服器作業系統中執行，例如：Windows 2000 Professional, Windows NT Workstation 4.0 及 Windows 98。此外，它也可以支援雙處理器系統。雖然此版本可以支援各種規模的資料庫，因此它的最佳化效能是針對個人使用者與小型工作群組，超過 5 位同時使用者所產生的工作量時，效能將會降低。
- SQL Server 2000 程式開發版--是 SQL Server 提供開發者使用 SQL Server 建立各類型的應用程式。它包含了企業版的全部功能，但依據特別的開發與測試使用者授權合約書(EULA)的功能，此版本禁止產品的部署使用。
- SQL Server 2000 Desktop Engine (MSDE)--提供了 SQL Server 2000 的基本資料庫引擎功能。它不包括使用者介面、管理工具、分析能力、用戶端存取使用權、開發者函數館或線上叢書。同時也限制資料庫規模與使用者的工作量。它是 SQL Server 2000 各種版本中耗用資源最少的版本，因此很適合內嵌或離線資料儲存。
- SQL Server 2000 Windows CE 版--是針對執行 Windows CE 的各種裝置與家電所設計的 SQL Server 2000 版本。它的程式與 SQL Server 2000 的其他版本相容，因此開發者能夠利用現有的技術與應用程式，將關聯式資料儲存的功能延展到到新類型裝置的各種解決方案之上。

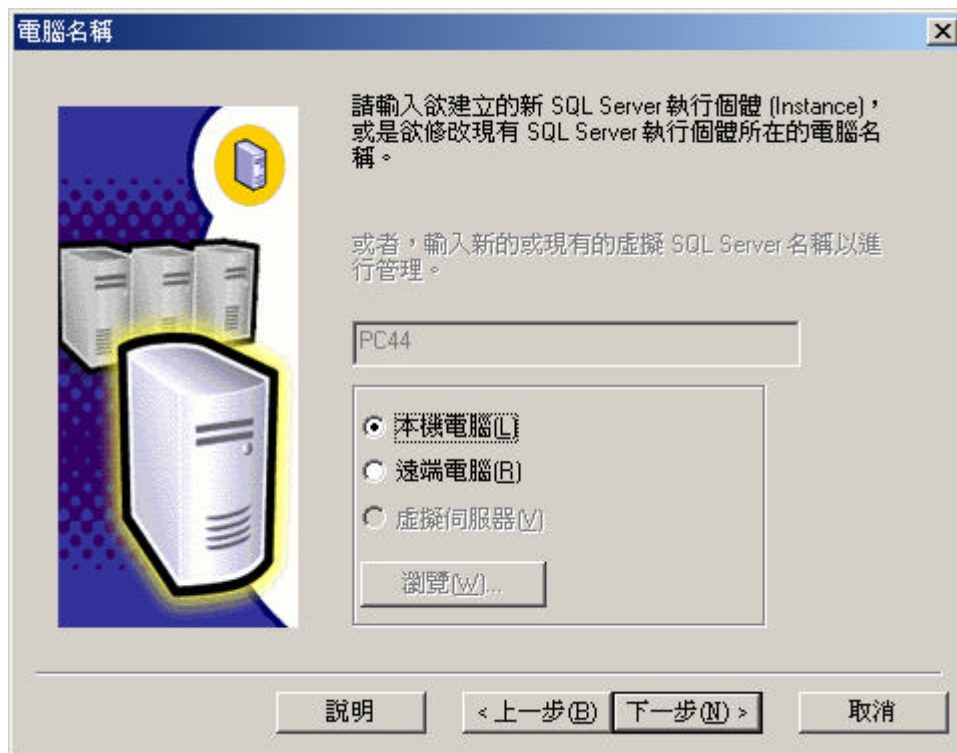
SQL Server 7.0 到 SQL Server 2000 最重大一項改變就是新增支援 XML，有了此一支援，經由 SQL 指令向 SQL Server 查詢所得到的結果可直接轉換為 XML 文件，並且在 browser 下顯示。除此之外，也可以經由 XML 文件內的內容對資料庫做新增、修改、刪除的動作。

3.8.2 安裝：

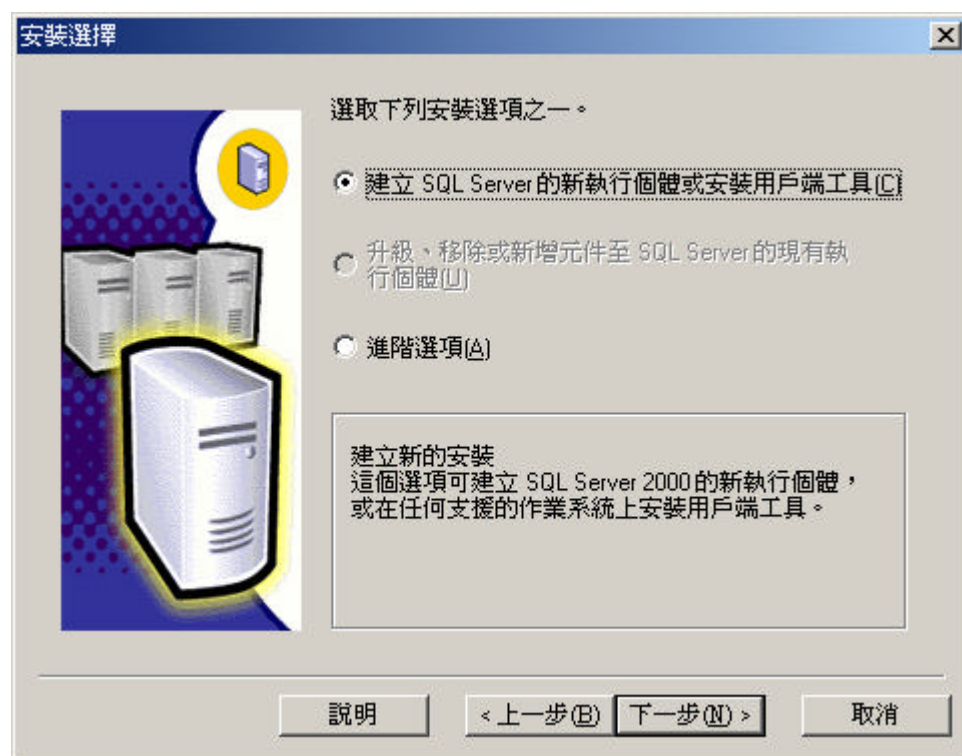
步驟一：將 SQL Server 2000 光碟片放入光碟機內，由 autorun 功能選定要安裝的元件後，即可進入安裝精靈，如下圖，按「下一步」。



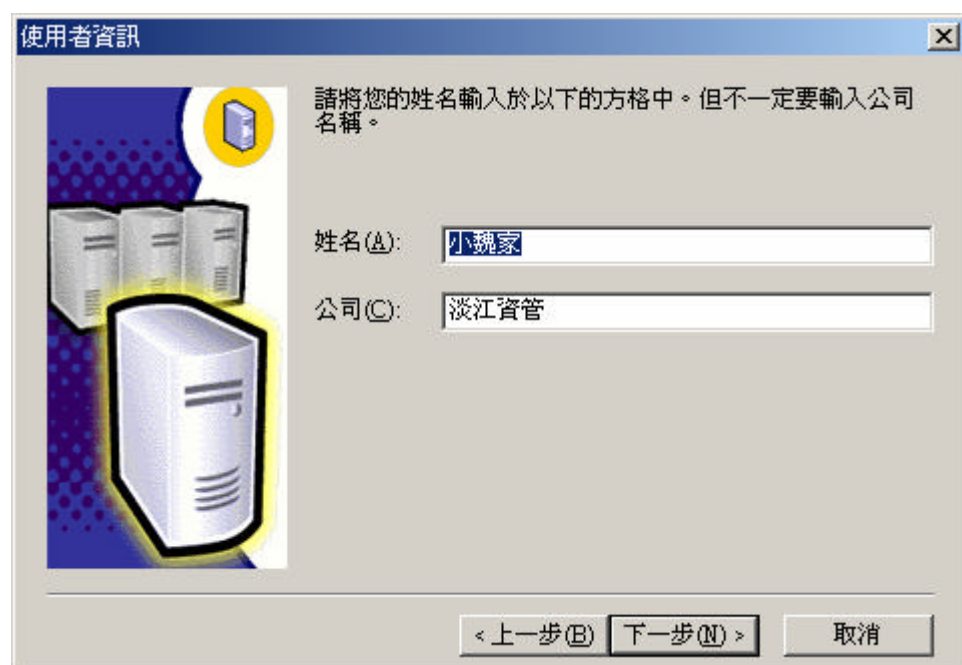
步驟二：接下來會出現如下圖，其中「遠端電腦」是指要透過網路將 SQL Server 安裝在遠端電腦上，我們選擇「本機電腦」後按「下一步」。



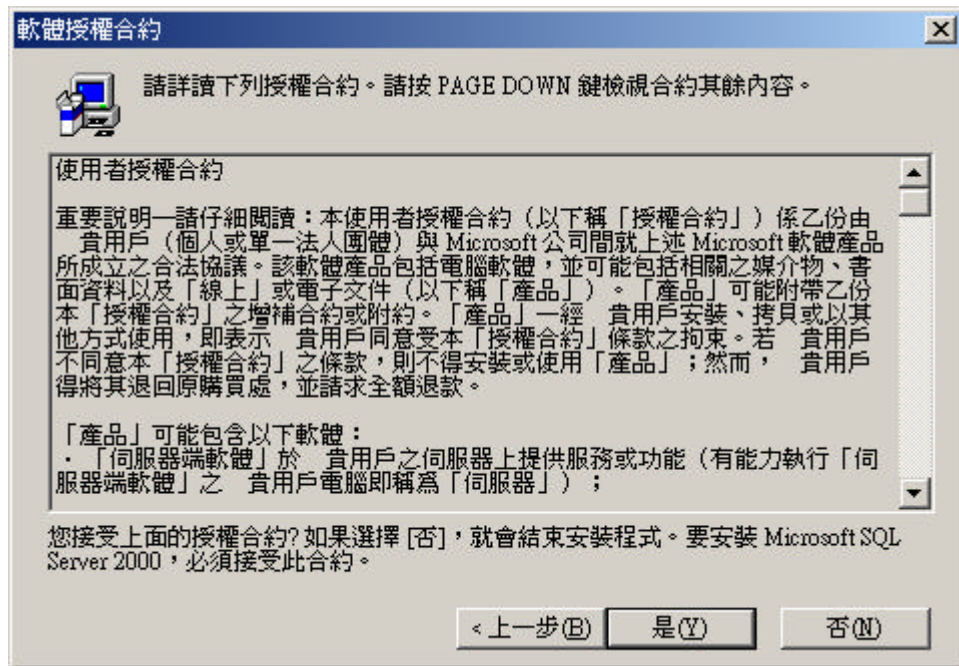
步驟三：之後出現如下圖，選擇安裝選項後按「下一步」。



步驟四：之後出現如下圖，輸入使用者資訊畫面，請輸入相關資訊後按「下一步」。

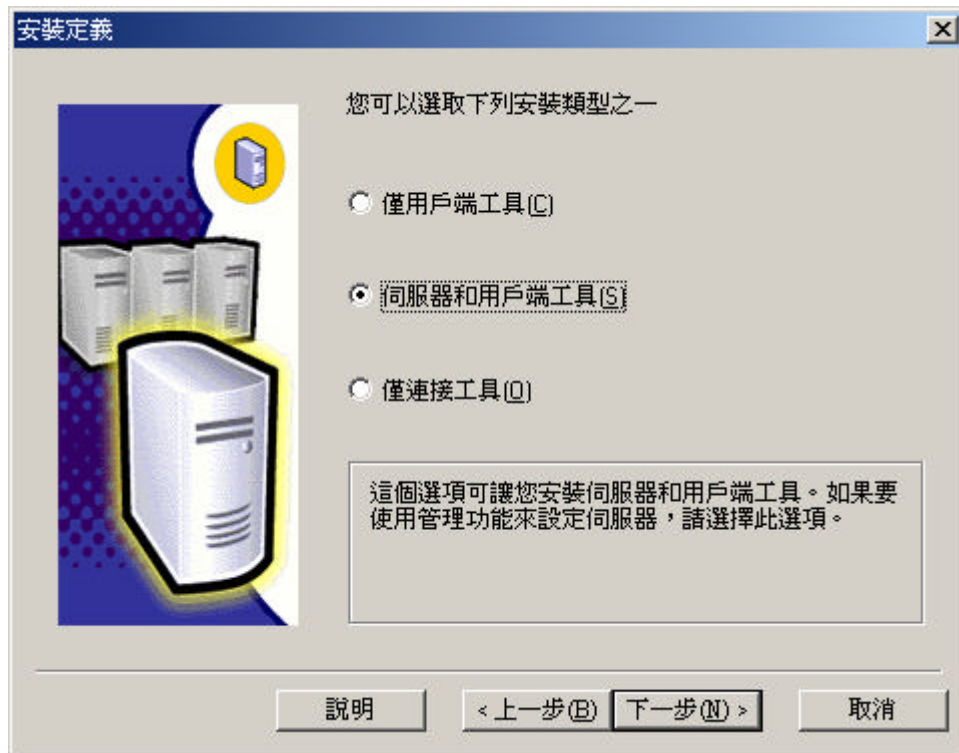


步驟五：出現軟體授權合約畫面，按「是」。

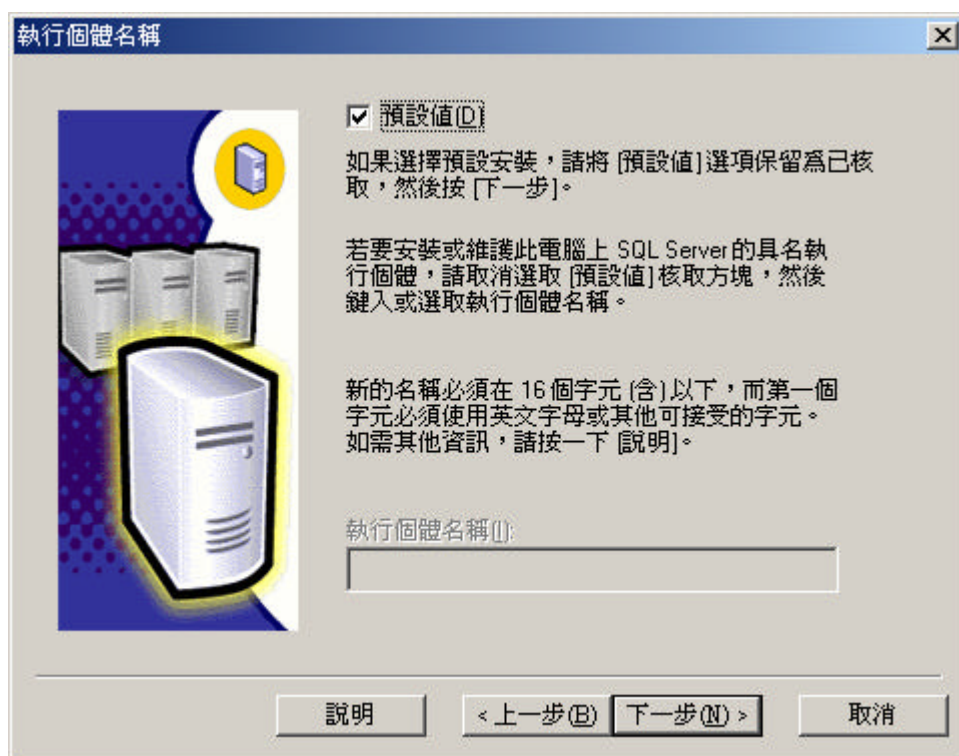


步驟六：出現輸入 CD-Key 的畫面，輸入產品版權序號後按「下一步」。

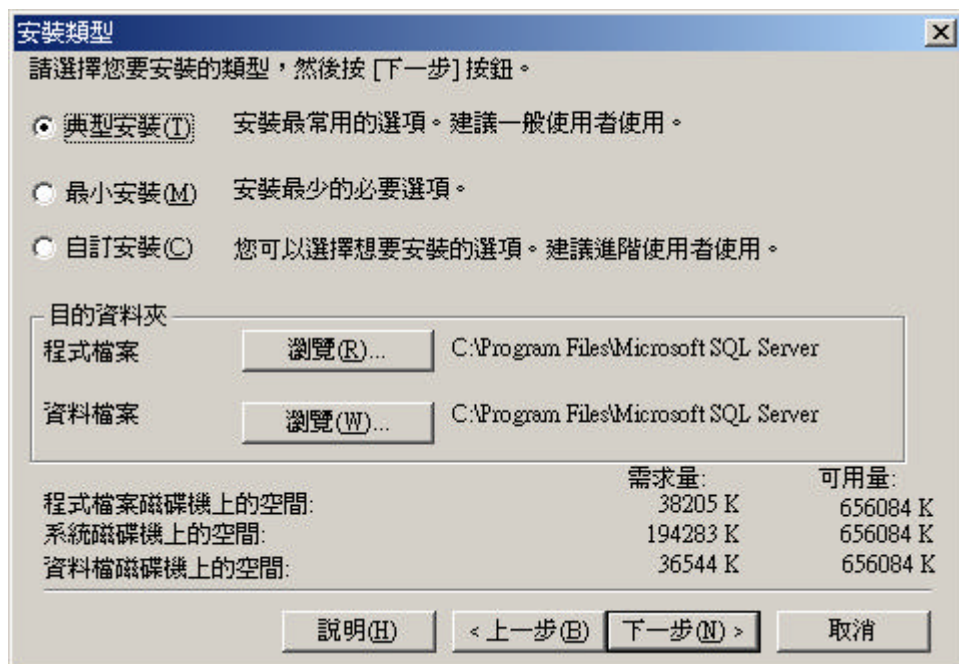
步驟七：選擇安裝「伺服器 and 用戶端工具」後按「下一步」。



步驟八：皆下來出現輸入執行個體名稱，由於 SQL Server 2000 可以在同一台主機上安裝兩個以上的個體，因此可以對個別個體命名，當安裝第一個個體時，如果點選「預設值」的話，將以系統主機的名稱加以命名。可以不要點選「預設值」而在底下的執行個體名稱輸入想要的個體名稱，完成動作後按「下一步」。



步驟九：接下來出現選擇安裝型別畫面，通常我們都選「典型安裝」，使用預設的安裝選項值即可，對系統較瞭解的人也可以選擇「自訂安裝」再依需要做設定。



步驟十：接下來出現服務帳戶的畫面，此畫面要求輸入一個 NT 網域上的帳號作為啟動 SQL Server 用，此一帳號用來登入 NT 網域，經由登入 NT 網域，MS SQL Server 才會整合成為 NT 網域上的一個服務。

此一帳號必須事先在 NT Server 上使用「網域使用者管理員」建立，而且必須設定為密碼永遠有效，並且可在任一時段登入 NT 網域，另外，該帳號必須加入安裝 SQL Server 這部主機上的 administrator 此一區域群組內成為其組員。

除了 MS SQL Server 這個主要的服務外，另外還有一個 SQL Server Agent 服務，也可以替它指定一個啟動執行帳號，或者使用和啟動 MS SQL Server 此一服務相同的帳號。

完成上述動作後按「下一步」。

服務帳戶

每個服務都使用相同的帳戶，並且自動啟動 SQL Server 服務(E)

個別自訂服務的設定(Z)

服務

SQL Server(S)

SQL Server 代理程式(A)

服務設定

使用本機系統帳戶(L)

使用網域使用者帳戶(R)

使用者名稱(U): Administrator

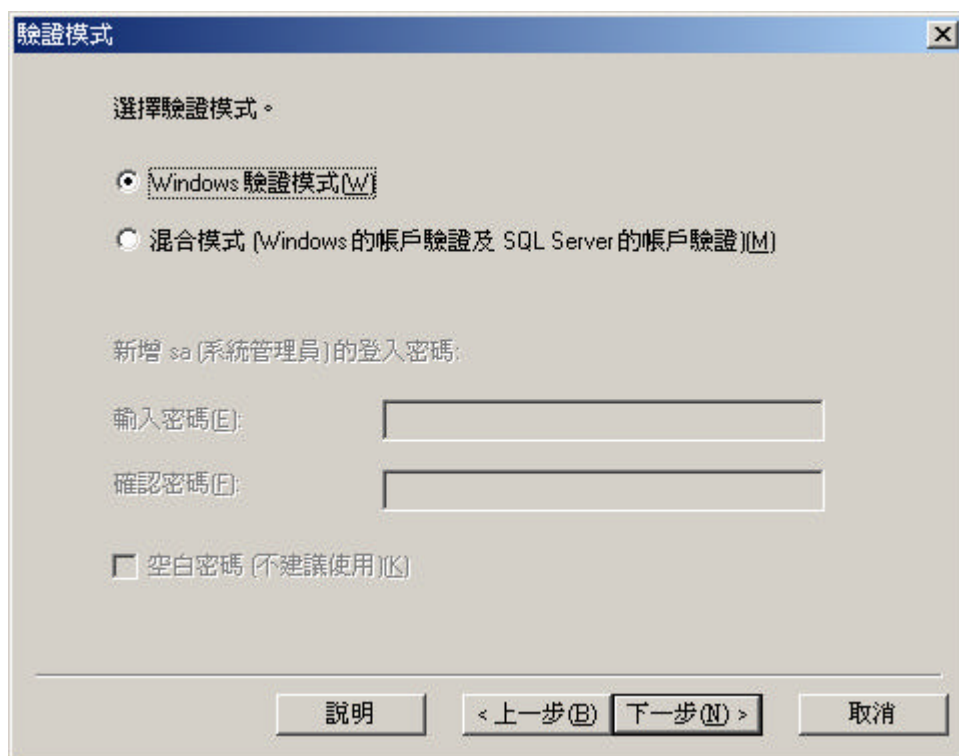
密碼(P): xxxxx

網域(D): PC44

自動啟動服務(O)

說明(H) < 上一步(B) 下一步(N) > 取消

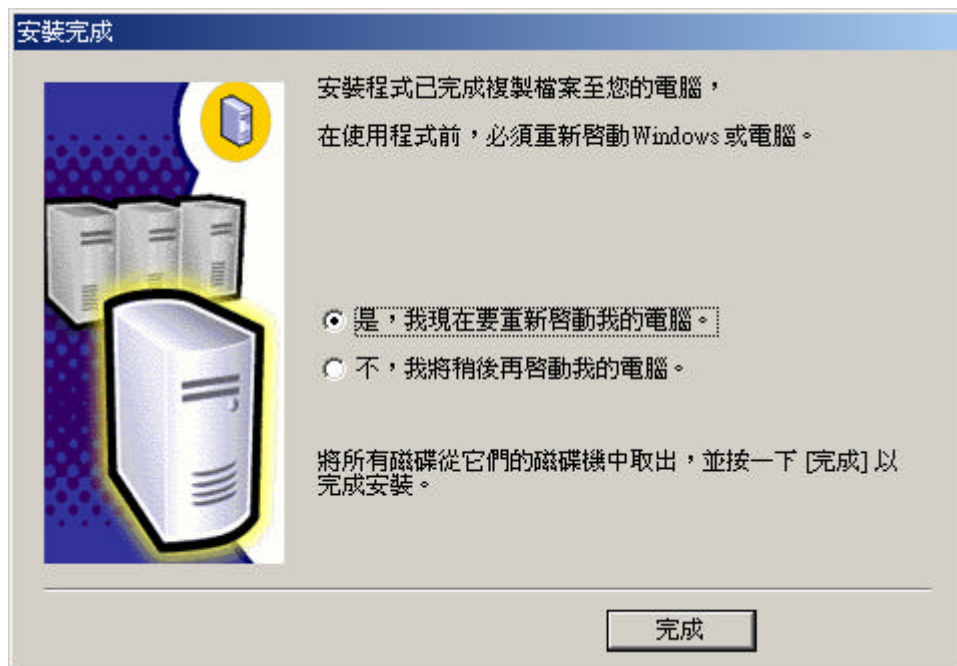
步驟十一：接下來出現驗證模式畫面，在此可設定此 SQL Server 的連線驗證模式，可以選擇 Windows 驗證模式或混合模式，在此選擇 Windows 驗證模式後按「下一步」。



步驟十二：之後如果無須更改之前的設定即可按「下一步」開始複製檔案。



步驟十三：當所有檔案複製完成後，會出現安裝完成畫面，如下圖，依指示重新啟動電腦完成所有安裝動作即可。



3.9 憑證伺服器與 SSL

3.9.1 名詞簡介：

3.9.1.1 憑證：

Windows 2000 作業系統中有一個 Certification Service，它就是發行授權憑證的服務，而有許多的安全服務都是使用憑證服務的方式，透過公開金鑰，來建構使用者身份驗證，以及其他安全性服務的基礎。使用者可利用他們取得的憑證資訊，發送加密過的電子郵件，也可以對檔做簽證(即所謂的電子簽章)，甚至使用者可以將憑證資訊儲存在智慧卡 (smart card)上，透過刷卡的方式，就能登入系統。

3.9.1.2 憑證授權單位：

憑證授權單位 (Certificate Authority, CA)是提供發行憑證服務的單元，CA 必須負責審核每一位要求憑證資訊的使用者身份，以決定是否要將憑證資訊發行給該名使用者。

CA 的授權服務是以微軟的公開金鑰系統 (Public Key Infrastructure, PKI)的編碼技術來建立，並適用於網際網路的安全性架構，所以，CA 可以由外界的商用性來擔任，或者自行在公司內部網路建構憑證授權單位。CA 是以階層式的架構存在的，最上層的 CA 又稱為 Root CA(根憑證授權單位)，在 Root CA 底下可以有許多子 CA，又稱次級憑證授權單位 (Subordinate CA)。

Windows 2000 作業系統裡提供了兩種不同型態的憑證授權單位，一種是與 Active Directory 整合的企業型憑證授權單位 (Enterprise CA)，此類 CA 是應用在公司內部與 AD 整合的情況，所有的授權憑證是存放在 AD 上；另一種是獨立型憑證授權單位 (Stand-alone CA)，獨立型憑證授權單位通常會應用在與非企業內部的 CA 做信任的架構下，它的授權憑證則是存放在該 CA 的電腦本機上之硬碟裡。

3.9.1.3 伺服器憑證：

伺服器憑證 (Server Certificate) 是一個數為憑證，其中包含了 Web 伺服器資訊、核發伺服器憑證之 CA 資訊，伺服器憑證也包含一個伺服器公開金鑰，用來在用戶端與伺服器之間建立安全的連線。若要啟動 Web 伺服器之 SSL(Secure Socket Layer)安全功能，必須安裝一個伺服器憑證。

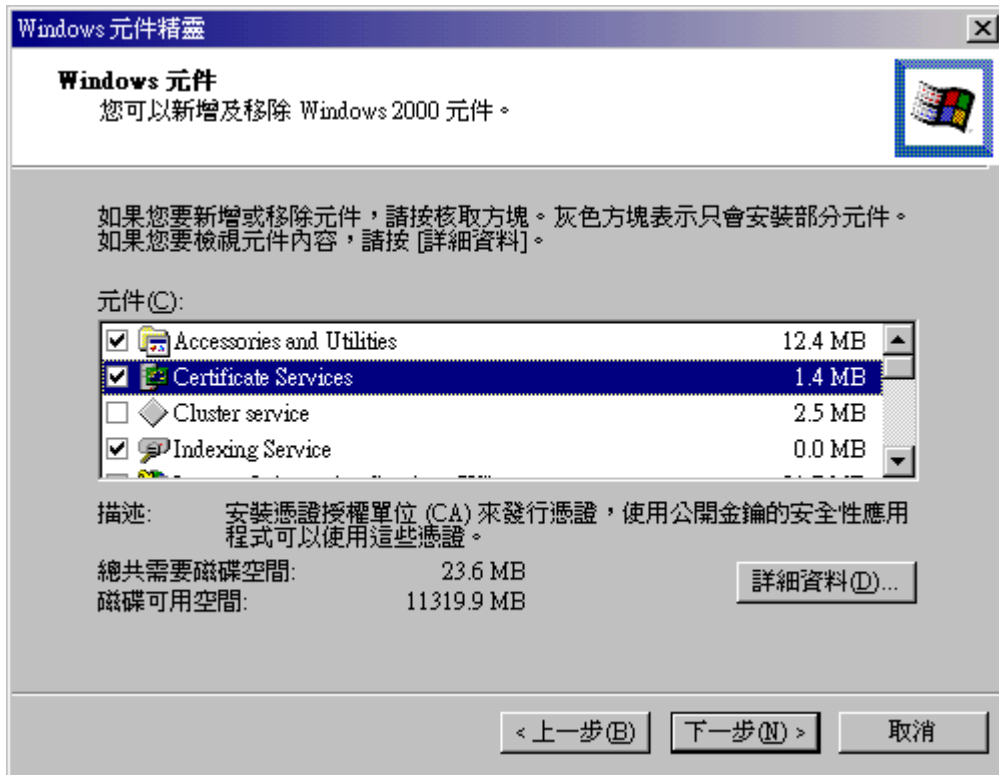
3.9.1.4 SSL：

由網景 (Netscape) 開發的 SSL 是 Secure Socket Layer 的縮寫，是一種網際網路上最普遍使用的安全通訊協定，保障網站伺服器及瀏覽器之間的數據資料傳輸的安全性。透過使用這個協定，網路上的數據傳輸會按照認證的種類(40 位元、128 位元) 進行不同程度的加密，更會檢查資料的完整性。除此以外，透過所謂『金鑰匙』的加密技術及嚴謹的 SSL 認證註冊的程序，SSL 可以驗證伺服

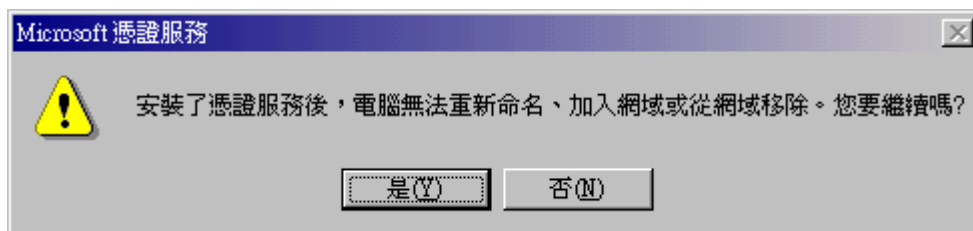
器的身分而達到網站瀏覽者向網站身分作出檢查的目的。

3.9.2 憑證伺服器的安裝：

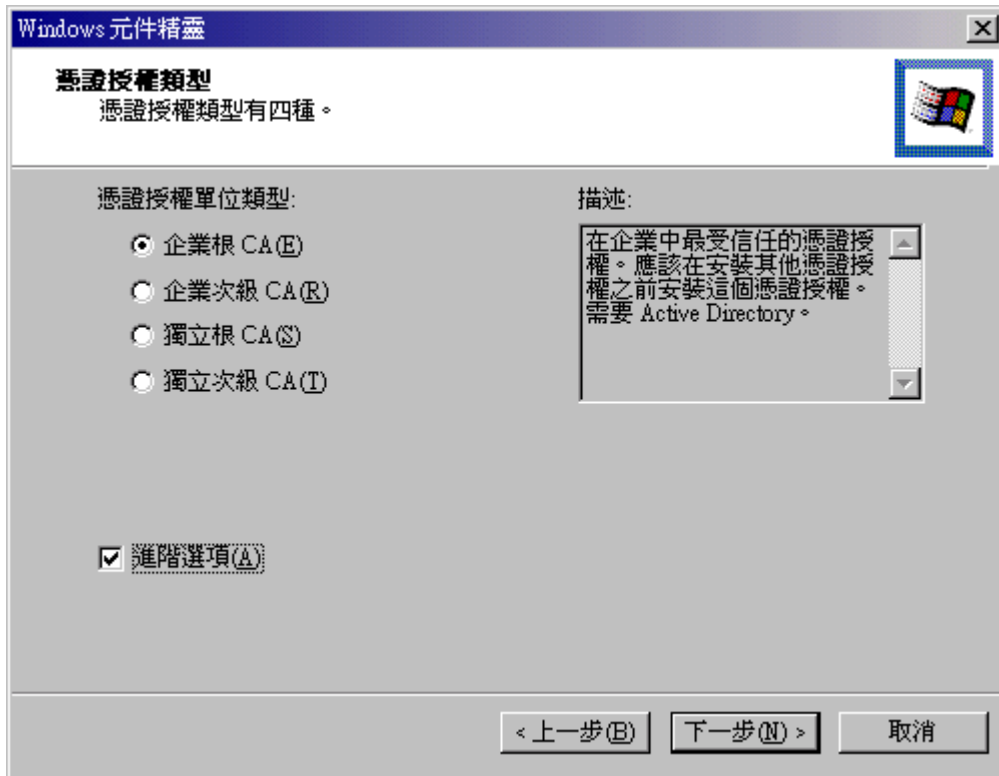
步驟一：從「控制台」→「新增或移除程式」→「新增/移除 Windows 元件」，勾選憑證服務 (Certificate Services)。



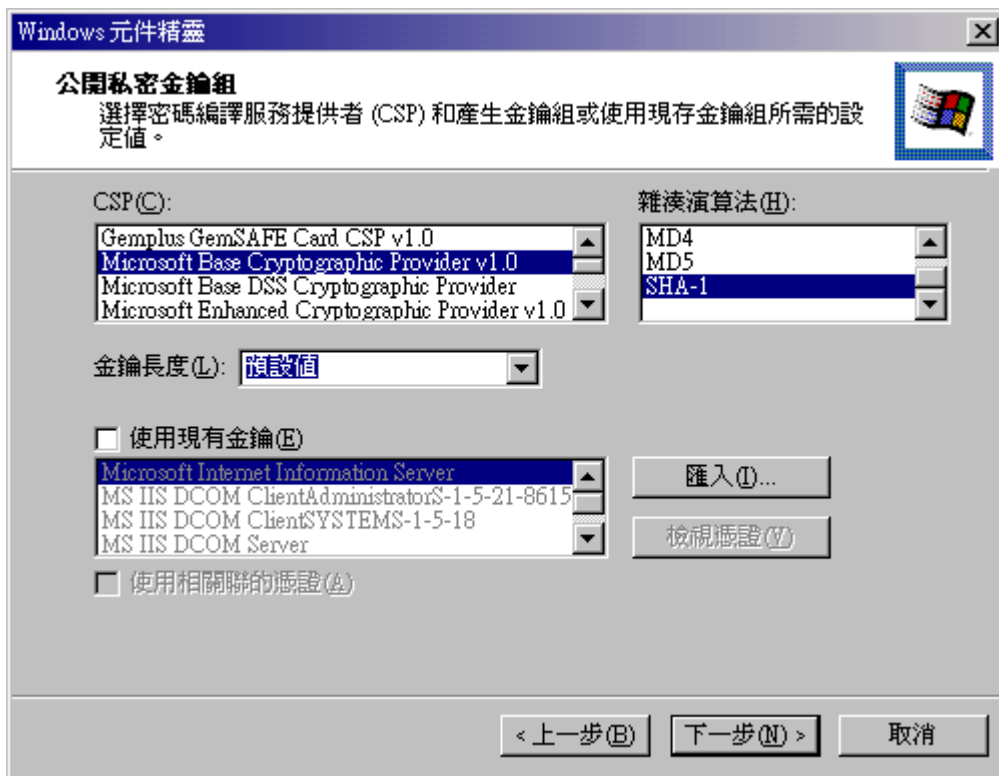
步驟二：因為安裝憑證服務後，此台電腦將無法重新命名、加入網域或從網域移除，所以要求使用者確認。



步驟三：選擇要安裝的憑證授權單位類型，下方「進階選項」是讓使用者可以自行作公開私密鑰的設定。



步驟四：選擇密碼編譯服務的提供者和其他設定值，金鑰長度越長，加密解密的時間也就越長，但密文相對的也就越安全。



步驟五：輸入 CA 的相關資訊，當 CA 的有效期限過期時，管理人員就必須重新更新一次所有的信任關係，考量到安全性及系統管理的負擔，根憑證授權單位的有效期間必須合理，預設為兩年。

Windows 元件精靈

CA 識別資訊
請輸入資訊來識別這個 CA

CA 名稱(A): IDV Home CA

組織(O): IDV

組織單位(U): Home

縣市(I): TPE

省份(S): TWN 國家(地區)(C): TW

電子郵件(E): abc@def

CA 描述(D): My Testing Home CA

有效期間(Y): 2 年 到期日: 2004/9/2 下午 01:50

< 上一步(B) 下一步(N) > 取消

步驟六：指定憑證資料庫儲存的位置。

Windows 元件精靈

資料存放位置
指定設定資料，資料庫及記錄檔的存放位置

憑證資料庫(C): C:\WINNT\System32\CertLog 瀏覽(O)...

憑證資料庫記錄檔(L): C:\WINNT\System32\CertLog 瀏覽(W)...

將設定資訊儲存在共用的資料夾(S)
共用資料夾(H): 瀏覽(B)...

保留現存的憑證資料庫(B)

< 上一步(B) 下一步(N) > 取消

步驟七：因為憑證服務同時也支援 IIS 伺服器的運作，所以若 IIS 伺服器已經處於啟動的狀態，安裝精靈會要求先將 IIS 伺服器停止。

接下來，便開始安裝憑證伺服器，安裝完成後可以在「系統管理工具」中啟動憑證授權的管理工具。



3.9.3 取得伺服器憑證的兩種方法：

- 1、使用 Microsoft 2000 Server 的憑證伺服器自己發出憑證，及自己扮演 CA。
- 2、從可信任的 CA 取得憑證。

3.9.4 建立 Web 伺服器的 SSL 安全機制步驟：

- 1、產生一對金鑰—申請者使用「Web 伺服器憑證精靈」來建立憑證要求（準備一個申請伺服器憑證的檔案），即產生一對金鑰（就是 Windows NT 的識別碼），即公開金鑰和私密金鑰。
- 2、向 CA 申請核發一個伺服器憑證—將憑證要求傳送至 CA，若使用 Microsoft 2000 Server 的憑證伺服器扮演 CA 時為使用 IE 瀏覽器中連線到 Windows 2000 憑證服務員網頁（<http://servername/certsrv>）。
- 3、CA 核發伺服器憑證—CA 處理要求後會傳送憑證給申請者。
- 4、安裝伺服器憑證—申請者使用「Web 伺服器憑證精靈」來安裝憑證。
- 5、在 Internet 服務管理員中啟用站台之 SSL 功能。
- 6、安裝憑證路徑後，使用者的電腦便可以新任從這個 CA 所發出的憑證。

3.10 索引伺服器

3.10.1 索引服務的基本概念：

索引服務會從一組文件中取出一些資訊，並使用 Windows 2000 的搜尋功能或 Web 瀏覽器快速存取該資訊來組織它。此資訊可包含文件（其內容）中的文字、文件（其屬性）的特性以及參數（如作者名稱）。只要建立好索引之號，就可以搜尋內含的關鍵字、片語或屬性之文件的索引。

透過索引的處理程序，索引服務會使用文件篩選器掃描文件，並取出相關的文字與屬性以傳遞給索引者。索引服務會自動儲存 System 或 Web 類別目錄中的所有索引資訊（這兩個是預設類別目錄），而索引服務可以為下列類型的文件作索引：

HTML (.htm / .html)

文字檔 (.txt)

Microsoft Office 95 及以上版本

Internet 郵件及新聞

任何可使用文件篩選器的其他文件

索引服務的設計是為了進行持續性的執行，而且只需要花費極少的人力維護。因為再設定好定義之後，所有操作均會自動進行，包括索引建立、索引更新以及損毀後的復原。系統管理人員與應用程式開發人員都可為其他種類的文件開發自訂篩選器，而開發自訂篩選器的相關資訊請參考 Microsoft Platform Software Development Kit。

索引服務的索引程序與搜尋引擎的效能取決於索引文件的數目與大小、提交查詢的速率以及查詢的複雜性。當然，電腦本身的設備也會嚴重影響執行效能。如果目前查詢的數量不多，則具備 Windows 2000 Server 最小硬體需求的電腦就可順利處理查詢。下表是依據要索引的文件數目而建議的記憶體設定：

要索引的文件數目	最少記憶體 (MB)	建議的記憶體 (MB)
少於 100000	64	64
100000-250000	64	64-128
250000-500000	64	128-256
500000 以上	128	256 或更多

由上表我們可以知道，如果所要索引的文件數目很多，則記憶體的不足會嚴重影響執行效能。如果執行索引服務實效能降低，則可以嘗試使用修改「索引服務效能」程序來調整執行效能。當然較快速的 CPU 也可以改進索引的效能與處理查詢的速度。如果執行 Index Server 時發現速度緩慢，則可用效能監視器確定引起效能降低的原因。

在索引服務的儲存空間方面，索引的文件大小與所使用的檔案系統類型都會影響儲存索引服務所需的磁碟空間。在 FAT 檔案系統上，類別目錄與暫存工作空間所

需的空間大約是索引文字總數的 30%。而在 NTFS 檔案系統上，所需磁碟空間大約是索引文字總數的 15%。

索引服務有些需要注意的事項說明分述如下：

索引服務完全符合 NTFS 安全性，如果類別目錄是在 NTFS 磁碟機上，而使用者沒有查看文件的使用權限，則無法查看搜尋結果清單的文件。

如果將類別目錄放在 FAT 磁碟機上，則使用者皆可查看此類別目錄。

索引服務需要具有正確的使用權限才能存取所要索引的文件。在你的本機電腦上，索引服務會使用 System 帳戶來操作，因此，如果拒絕該帳戶存取文件或目錄的權力，則索引服務就無法索引這些文件。

索引服務永遠不會為加密文件建立索引，如果當文件在索引之後才予以加密，則索引服務會將它從類別目錄中移除。

請勿將類別目錄存放在網站上，因為將類別目錄檔案放在 Web 根目錄或虛擬目錄之下，則 IIS 有時會鎖定類別目錄檔案並防止更新，這樣的結果再所飲食可能會導致無窮的索引迴圈。

如果使用的是多重類別目錄，則請在你的磁碟機上製作類別目錄資料夾，並在此資料夾的子資料夾中存放每個類別目錄。

在執行索引服務時不要執行掃毒或備份軟體，因為有時這類程式會鎖定檔案，而導致索引服務在建立索引時逾時。

3.10.2 索引伺服器簡介：

索引伺服器（Index Server）能對 Web 站台的內容建立索引，讓使用者可以在非常短的時間內找到所需的內容，在 Windows 2000 中就這個元件合併在作業系統中，索引服務會為磁碟上的文件與文件內容編製索引，並將資訊存放在類別目錄中，因此我們可以使用「索引服務」搜尋文件，也可以透過「開始」→「搜尋」功能或 Web 瀏覽器進行搜尋。

3.10.3 索引伺服器安裝：

由於 Windows 2000 Server 會根據伺服器的角色來安裝各種元件，如果機器尚未安裝 Indexing Service，則請進入「新增/移除 Windows 元件」中，然後勾選「Indexing Service」選項即可。

安裝完成後 Indexing Service 會被啟動並開始為 IIS 伺服器所設定的虛擬目錄中的檔案編排索引。在編排索引之後，如果對 IIS 伺服器所設定的虛擬目錄增加檔案時，Indexing Service 都會繼續為新增的檔案編排索引，即使 WWW 服務並未執行也不會影響 Indexing Service 對新增檔案的編排索引。

3.11 ASP

3.11.1 ASP 介紹：

ASP(Active Server Pages)是一種類似 HTML(超文字語言)、Script(VB Script、JavaScript、JScript)與 CGI(通用閘道介面)的結合體，他的用途、寫作觀念，和 CGI 的語言類似；但其執行效率卻比 CGI 好、程式寫作也比 HTML 更便利有彈性，安全及保密性也遠比 Script 佳，學習 ASP 也較為簡單。

ASP 是屬於微軟 Internet Information Server，他能夠讓您構築一個動態、互動性、高效能的網站；也屬於伺服器端(server-side scripting)動態網頁(一種在伺服器端執行的手稿語言)。

ASP 程式只要直接將控制語法加入 HTML 中即可，完全不需要重新編譯成執行檔就可以直接執行，其內建的 ADO(ActiveX Data Object)元件，允許使用者透過客戶端瀏覽器存取各式各樣的資料庫。

ASP 與 CGI 最大的不同在於物件導向及元件再使用，ASP 除了內建的 Request 物件、Response 物件、Server 物件、Session 物件、Application 物件及ObjectContext 物件等基本物件外，另外允許使用者以外掛的方式使用 ActiveX 元件；這些元件包括廣告迴轉元件、檔案存取元件、文件連結元件及資料庫存取元件等。

3.11.2 ASP 用法簡介

1：ASP 語言

ASP 是由靜態的 HTML 和動態的 ASP 代碼組成，採用 ASP 的一個很大的優點是，可以在客戶端中包含你的服務器端手稿的運行結果。為了能夠區分代碼的起始位置和終止位置，在 ASP 中是用 `<% ASP Code %>`來標識，其中`<%`是 ASP 代碼的起始的標記；`%>`是 ASP 終止的標記；ASP Code 就是任意的 ASP 代碼。(源代碼)

2：手稿語言的選擇

同時，你也可以在 ASP 中用`<Script>`標記，但是你必須設定你的 RunAt 參數，使它的值設定為你的服務器，否則你的代碼將會被發送到客戶瀏覽器端進行處理。在默認的情況下，IIS 會認為你的 ASP 代碼是 VBScript 編寫的。當然你也可以使用 JScript，只需要在頁面的頂部用處理命令 `<%@ LANGUAGE=VBSCRIPT%>`進行聲明就可以了，注意，這個聲明必須在這個頁面的第一行，且在 LANGUAGE 和@之間要有空格。當然你也可以在一頁中指定不同的手稿語言，例如：將主手稿語言設置為 VBScript 而期間再用 JavaScript 語言，格式如下：

```
<% @ LANGUAGE=VBScript RUNAT=Server%>
```

```
<html>
```

```
.....
```

```

<body>
    .....
    <Script language="JavaScript" runat= "Server">
        .....
    </Script>

    .....
</body>
</html>

```

3：代碼的使用

在 VBScript 中注釋是以單引號(')來表示的，同時 VBScript 中的變量不區分大小寫，ASP 內建對象和函數以及 VBScript 中的函數和結構都是不區分大小寫的。在 VBScript 中變量必須是以字母打頭，且長度不超過 255 個字符，它可以包含字母和數字，但是不能包含句點和其他特殊符號。在 VBScript 中不必事先聲明變量，可以直接使用。當然你可以事先聲明，用 Option Explicit 語句就是強制必須事先聲明所有變量。VBScript 聲明的變量是種變體 (Variant) 數據類型，當你為變量賦值時才決定以何種方式存儲這個變量。注意，當我們用一個變體來保存對象時，要使用 Set 語句來賦值。

在 VBScript 中變量的使用期限（也就是作用域）是依賴於在什麼地方聲明了這個變量，如果要在網頁中聲明變量，那麼它的使用期限是一直到這個頁面停止處理時才結束它的使用期限；如果在過程中聲明變量，那麼這個變量的使用期限也就是從各個過程開始到這個過程結束。

函數和過程一樣都是命名了的代碼塊，但它們卻有很大的區別，過程完成程序任務，函數則返回值。舉個例子，當你想獲取當前的系統日期，你只要 Date 函數，就可以得到系統的當前日期，再舉個代表性例子，當你想獲取某個數的平方根，你只要將該數傳給 VBScript 的 Sqr() 函數，此函數會立即返回該數的平方根。如：

A=sqr(9) ，則 A=3。

4：操作符

VBScript 進行數學運算時有+、-、*、和 / 等這些基本的數學符號進行操作，同時還可以用 \ 進行整數除法，這種除法只返回結果的整數部分。字符的連接是通過 & 來進項

了解編程語言的朋友一定知道在程序中控制程序流程的語句主要可以分為條件語句和循環語句，在 VBScript 中可使用 If...Then...Else 語句和 Select Case 語句來作為條件語句。

If...Then...Else 語句用於計算條件是否為 True 或 False，並且根據計

算結果指定要運行的語句。通常，條件是使用比較運算符對值或變量進行比較的表達式，If...Then...Else 語句可以按照需要進行嵌套。Select 語句是根據變量的數值來走那個分支。（源代碼）

常用循環語句有以下四個：

Do...Loop: 當（或直到）條件為“真”時循環。

While...Wend: 當條件為“真”時循環。

For...Next: 指定循環次數，使用計數器重複運行語句。

For Each...Next: 對於集合中的每項或數組中的每個元素，重複執行一組語句

5、認識 ASP 的物件

ASP 其實有 7 大物件，這 7 大物件各個都有強大的功能：

h Request 物件

h Response 物件

h Application 物件

h Session 物件

h Server 物件

h Connection 物件

h Command 物件

以下簡介此七大物件：

Request 物件--當使用端向伺服器端要求資源，就會有一種互動。Request 是一種擷取程序，Request 可以從表單、伺服器、Cookie 等等擷取程式處理時需要的數值。

Response 物件--Response 是一種伺服器的反應程序，可以在某種情況下或是某個特定的地方讓伺服器對使用者做出動作。如 Response.write 是回應一段文字，Response.redirect 是跳到另外一個網頁。

Application 物件--在程式執行時，有一些需要跨網頁讓多個使用者分享的數值（如線上人數），這時候 Application 物件就可以拍上用場。

Session 物件--當程式在執行的時候，有時候會用到一些常駐的變數（如會員登入），大部分這種功能都是用 Session 達成，Session 的好處是會在使用者開啟該 Session 物件後會一直存在直到使用者閒置太久或是生命週期結束，不像普通的變數，當程式終結的時候變數就消失了。

Server 物件-- Server 物件的主要用途就是讓伺服器在執行某個網頁的時候可以藉由 Server 物件來使用外部的資源或是元件。

Connection 物件-- Connection 物件可以讓一個 ASP 程式使用某種形式的資料庫或是資料儲存媒介。比較簡單的用法是配合 SQL 一起使用，當然，還有許許多多強大的功能。

Command 物件--Command 物件是一個進階的使用，他的功用是可以處理各種類型的指令。

6、基本 cookies

cookies 物件的功能主要是用於記錄瀏覽所輸入的資料，或將 ASP 傳遞/接收的變數暫存在 cookies 中以達到程式應用的功能。cookies 在運作上時是將 ASP 存入的變數存在 WINDOWS 目錄下 cookies 這個目錄下。一般是 c:\windows\cookies\ 以 IP 或主機名稱存為文字檔，下面是 cookies 的基本語法：

寫入 COOKIES:

```
response.cookies("變數名")=存入變數  
response.cookies("變數名").expires="2030/1/1"
```

讀出 COOKIES:

```
變數=request.cookies("變數名")
```

在寫入的程式中，運用到 ASP 的主要物件 response，response 是輸出用的物件，配合上.cookies("變數名") 就等於 response.cookies("變數名")

如此便可以將存入變數輸出到 cookies 儲存，而寫入的程式的第二行便是用設定此 cookies 變數的有限期限，若您沒有加上第二行則瀏覽器關掉後此 cookies 變數便消失不見了。

接下來是讀出的程式部份：

這裡只要把.form("欄位名稱") 改成 .cookies("變數名")

如此就可以讀出 cookies 變數了

接下來介紹常用的 cookies 語法：

以下是上面教的語法

```
response.cookies("變數名")=存入變數  
response.cookies("變數名").expires="2030/1/1"
```

現在，把語法改成下面這樣：

```
expiredate=dateadd("d",幾天內有效,date)  
response.cookies("變數名")=存入變數  
response.cookies("變數名").expires=formatdatetime(expiredate)
```

在第一行中，設定變數 expiredate 由 dateadd() 這個函數取得目前日期部份並計算幾天內有效 而語法中 dateadd("d", 幾天內有效, date) 的"幾天內有效"是我們要設定由目前算起幾天以內有效的變數。

例如：如果要設定一個月，就把"幾天內有效"

改成 30-->dateadd("d", 30, date) 這樣，這個 cookies 變數的有效期限就變成一個月了。

而第三行中，我們把.expires="2030/1/1" 改成了

```
expires=formatdatetime(expiredate)
```

如此在儲存 cookies 變數時便會以目前日期加??天的方式儲存其有效期限。
在使用的時候只需要更改 expiredate=dateadd("d",幾天內有效,date)中
設定自己的應用需求便可。

7、ASP 函式寫作：

基本寫法：

以下是函式的大概結構--

```
Sub 函式名稱()  
...  
函式內容  
...  
End Sub
```

實際的例子--

```
Sub arthur()  
response.write("這是函式寫出的字串")  
End Sub
```

sub 是宣告函式用的，此例子中你已經宣告了 arthur() 這個函式！

寫好了這個函式，必須要呼叫函式...

呼叫函式語法--

```
arthur()
```

如果沒有什麼錯誤的話應該可以看到網頁上出現這是函式寫出的字串這段文字。

參數：

在所有的程式設計函式中，所謂的 x 就是參數，參數可以讓執行函式的時候輸入一些數值。

```
Sub try(master)  
response.write("您輸入的字串是" & master)  
End Sub
```

在這個範例中可以看到 try()的括弧中間有一個 master，這個就是參數，當呼叫此函式的時候，可以用：

```
try("這是我輸入的參數")
```

來對應一個數值或是字串給 master 參數，當然，您也可以宣告多個參數，如：

```
try(master,secondary)
```

然後呼叫

```
try("這是我輸入的參數","這是第二段文字")
```

參數與參數之間記得用逗號分隔。

回傳數值的函式：

ASP 中有一個規定...那就是在同一個 ASP 網頁中不能夠有相同名稱的函式和變數，比如說，您寫了一個叫做 hello()函式，那在同一個 ASP 網頁中您就不能有名稱為 hello 的變數，原因是和函式名稱相同的變數代表的是該函式的回傳值(函式分成兩種:回傳數值與不回傳數值)有回傳數值的函式基本上寫法和之前的函式是一樣的，但是用法和用處有點不一樣。

由於回傳數值的函式除了執程式碼還會回傳數值，所以用法有點像是變數。

範例：

```
Sub arthur()  
arthur=123  
End Sub
```

呼叫範例：

```
response.write(arthur())
```

在這個範例中就是用 response.write()將 arthur()函式回傳的數值顯示出來。

附錄 A IIS 安全設計概要

IIS5 是 Windows2000 作業系統所帶的 Internet 服務套裝程式，它包含了 www、ftp、smtp 以及 index server 等等眾多實用功能。無論是創建一個 Internet 外部站點，還是構造一個 Intranet 內部應用，使用 IIS5 都是非常好的選擇。同時，如何保障 IIS5 安全穩定運行、內容發佈正確可靠，這是系統管理員必須高度重視的問題。下面就 IIS5 的安全策略設計進行概要分析，旨在讓大家瞭解系統管理員應在哪些方面執行必要的安全配置，然後根據實際環境再在各個環節分別擴展，最終建立一個安全的 IIS5 伺服器。

為虛擬目錄設置適當的訪問許可權

正確設置虛擬目錄的訪問許可權，將會很大程度地影響其中檔的安全性。建議考慮以下幾個方面的設置原則：

文件類型	建議的訪問許可權
CGI 程式 (.exe, .dll, .cmd, .pl)	Everyone (X) Administrators (Full Control) System (Full Control)
腳本文件 (.asp)	Everyone (X) Administrators (Full Control) System (Full Control)
包含檔 (.inc, .shtm, .shtml)	Everyone (X) Administrators (Full Control) System (Full Control)
靜態檔 (.txt, .gif, .jpg, .html)	Everyone (R) Administrators (Full Control) System (Full Control)

為不同類型的檔創建不同的目錄

如果一個虛擬目錄下面有多種類型的檔，按照上面的原則為每一種檔設置訪問許可權，無疑是非常煩瑣的事情。因此，我們可以採取為不同類型檔創建不同目錄的方式，然後再按照上面的原則為每一個目錄設置相應的訪問許可權。比如，我們可以創建如下的目錄結構：

1. C:\inetpub\wwwroot\myserver\static：存放靜態文字檔案(.html)

2. C:\inetpub\wwwroot\myserver\include : 存放包含檔(.inc)
3. C:\inetpub\wwwroot\myserver\script : 存放腳本文件(.asp)
4. C:\inetpub\wwwroot\myserver\executable : 存放可執行檔(.exe, .dll, .cmd, .pl)
5. C:\inetpub\wwwroot\myserver\images : 存放圖形檔(.gif, .jpeg)

另外，請注意以下 2 個特殊的目錄：

1. C:\inetpub\ftproot : FTP 服務所在目錄
2. C:\inetpub\mailroot : SMTP 服務所在目錄

這 2 個目錄的訪問許可權是 Everyone 完全控制(Full Control)，每個用戶都具有向其中添加資料的許可權。這樣，就有可能造成目錄所在磁碟的空間耗盡。因此，我們建議：

1. 將這 2 個目錄放置到另外的磁碟中，與其他的 IIS 服務程式分開。
2. 使用 Windows 2000 磁碟配置功能限制添加到這 2 個目錄的資料量。

為 IIS 日誌檔設置適合的訪問許可權

IIS 日誌檔記錄了所有訪問 IIS 服務程式的資訊，它對於系統管理員檢測故障非常重要。攻擊者為了銷毀他們的侵入痕跡，總是要想方設法刪除掉日誌檔。因此，我們建議對這些日誌檔進行重點保護，設置如下的訪問許可權：

1. Administrators (Full Control)
2. System (Full Control)
3. Everyone (RWC)

IIS 日誌檔一般位於如下路徑：`%systemroot%\system32\LogFiles`。

使用日誌檔

日誌檔對於檢查伺服器是否被攻擊是極為重要的。日誌檔有多種，我們建議使用"W3C 擴充日誌檔格式"，步驟如下：

1. 啟動"Internet 服務管理器"
2. 點擊滑鼠右鍵選擇要設置的站點，在從彈出功能表中選擇"屬性"
3. 點擊"Web 站點"選項卡

4. 點擊選中"啟用日誌記錄"核取方塊
5. 從"活動日誌格式"下拉選擇框中選擇"W3C 擴充日誌檔格式"
6. 點擊"屬性"
7. 點擊"擴充的屬性"選項卡，然後依次選中如下屬性：
 - 客戶 IP 地址
 - 用戶名
 - 方法
 - URI 資源
 - HTTP 狀態
 - Win32 狀態
 - 用戶代理
 - 伺服器 IP 位址
 - 伺服器埠

以上最後 2 個屬性只當一個電腦充當多個 Web 伺服器時有意義，也就是所謂的虛擬主機。屬性"Win32 狀態"對於調試非常有用，當它的數值等於 5 時，表示禁止訪問（access denied）。我們可以在命令行執行如下命令得到其他的"Win32 狀態碼"所表示的含義：

```
net helpmsg err
```

其中 err 表示 Win32 狀態碼。

禁止或者刪除所有的例副程式

默認安裝選項中，例副程式不會被安裝到機器中。對於一個正式應用的伺服器，我們不應該在其中安裝任何例副程式。如果已經安裝了某些例副程式，建議將它們完整刪除掉。為了查找方便，以下我們列出一些例副程式的默認安裝路徑：

例副程式類別	所在虛擬目錄	默認安裝路徑
IIS 例副程式	\IISamples	c:\inetpub\iissamples
IIS 文檔	\IISHelp	c:\winnt\help\iishelp
Data Access	\MSADC	c:\program files\common files\system\msadc

刪除虛擬目錄 IISADMPWD

這個虛擬目錄的作用是允許用戶重新設置他們在 Windows NT 或者 Windows 2000 作業系統上的帳號口令，應用環境主要針對企業內部網 Intranet。IIS5 的安裝中沒有包含這一項，但是如果是從 IIS4 升級到 IIS5，就會存在這個虛擬目錄。建議如果不是 Intranet 環境，就將之刪除。

刪除不使用的應用程式映射關聯

IIS 默認情況下可以解釋如 .asp、.htr、.shtm 等應用程式檔，當這些檔被 IIS 接收後，將交由一個 DLL 檔處理。如果使用不到其中某類檔，我們建議將刪除那個類型在 IIS 中的映射關係。步驟如下：

1. 啟動"Internet 服務管理器"
2. 點擊滑鼠右鍵選擇要處理的伺服器，再選擇"屬性"
3. 選擇"WWW 服務"，點擊"編輯"
4. 點擊"主目錄"
5. 點擊"配置"，顯示出當前"應用程式映射"列表

然後參照下表刪除相關類別：

如果不使用下列應用	就刪除掉以下項目
基於 Web 的口令修改	.htr
Internet 資料庫連接器 (注意:所有的 IIS5 Web 伺服器將使用 ADO 等相似技術代替資料庫連接器)	.idc
伺服器端包含檔 (Server-side Includes)	.stm, .shtm, and .shtml
Internet 列印	.printer
索引服務 (Index Server)	.htw, .ida and .idq

關於 Internet 列印服務，我們再說明一點。Internet 列印可以通過 Internet 服務管理器進行配置，也可以通過組策略進行配置。當兩者的配置間發生衝突時，以組策略的配置資訊優先。如果在 Internet 服務管理器中刪除了 Internet 列印，請記住不要再在本地或者域組策略中啟動它。默認的組策略配置中，既不啟動也不禁止 Internet 列印。在組策略中配置 Internet 列印的方法是：在 MMC 的組策略管理單元中，點擊"電腦配置"，點擊"管理範本"，點擊"印表機"，最後點擊"基於 Web 的列印"：

設置 IP 位址或者 DNS 功能變數名稱位址的訪問限制

如果 Web 站點隻對特殊 IP 位址的客戶服務，我們建議對 Web 站點執行 IP 位址限制的設置。如果設置是 DNS 功能變數名稱資訊，請注意，這會導致 IIS 執行 DNS 搜索工作，耗費一定的時間。

確認可執行檔的安全性

在訪問網站內容時，IIS 經常要執行一些 .DLL 文件。DLL 檔屬於可執行檔，可能會讀寫硬碟檔內容，所以最好確保其安全性。但是，如何判斷 DLL 檔的安全性，卻不是個簡單的事情。這裏介紹一個叫做 DumpBin 程式，它可以判斷出可執行檔

是否調用了某個 API 函數。比如說，我們使用下面的命令判斷檔 MyISAPI.dll 是否調用了 RevertToSelf：

```
dumpbin /imports MyISAPI.dll | find "RevertToSelf"
```

命令執行後，如果螢幕上沒有任何輸出資訊，就表明 MyISAPI.dll 沒有直接調用 RevertToSelf。

更新 IIS 伺服器上的根權威認證機構發放的證書 (Root CA Certificates)

要完成更新工作，需要 2 個步驟：

1. 首先，添加一個我們信任的新根權威認證機構發放的證書，特別是通過 Microsoft 認證服務 2.0 創建的新根權威認證機構證書。
2. 然後，刪除所有不被我們信任的根權威認證機構發放的證書。一般情況下，不被信任的機構就是指我們還不認識的機構。

所有被 IIS 使用的根權威認證機構證書都存儲在電腦中，我們可以按照下列步驟訪問它們：

1. 在"開始/運行"中輸入 mmc，點擊"確定"，啟動 Microsoft 管理控制臺，即 MMC
2. 從"控制臺"功能表中選擇"添加/刪除管理單元"，點擊"添加"
3. 選擇"證書"，點擊"添加"
4. 點擊"電腦帳戶"選項
5. 點擊"下一步"
6. 選擇要訪問的機器
7. 點擊"完成"
8. 點擊"關閉"，點擊"確定"
9. 擴展"證書"節點
10. 擴展"受信任的根證書頒發機構"
11. 選擇"證書"

這時，在右邊視窗格中顯示出當前所有的受信任的根證書頒發機構發放證書的資料。我們可以根據實際情況，刪除不需要的證書。但是請格外注意，不要刪除 Microsoft 公司或 Verisign 公司的相關證書，因為它們被作業系統廣泛地使用。

禁止或刪除不需要的 COM 元件

對於大多數應用程式不需要的 COM 元件，我們就應該考慮將其刪除，比如說"檔系統物件" (File System Object) 元件。請注意，如果刪除了"檔系統物件"元

件，"目錄物件" (Dictionary object) 元件也將被刪除。但是提醒一點，有些程式可能需要禁止的元件，比如說，Site 伺服器 3.0 就使用到"檔系統物件"元件。刪除"檔系統物件"元件的命令是： `regsvr32 scrrun.dll /u`。

重點檢查 ASP 代碼中的<FORM>輸入內容和查詢輸入內容

許多站點都使用來自用戶的輸入資訊去調用其他代碼或者直接創建 SQL 命令。換言之，輸入內容被假設為格式正確和語法正確。但是我們千萬不能這麼掉以輕心！許多攻擊者就是在輸入內容中填寫非法代碼從而巧妙偷窺伺服器的內容，甚至造成資料的毀壞。因此，我們要特別重視檢查來自<FORM>的輸入內容或者查詢字串資訊，只有其符合安全要求，才能進一步傳遞給下一個程式進行處理。我們可以使用 Jscript 版本 5 和 VBScript 版本 5 的常規運算式功能檢查文本資訊的合法性。來看看幾個例子。

下面的代碼是除去字串中所有非字母、非數位以及非_的字元：

```
Set reg = New RegExp
reg.Pattern = "\W+"
strUnTainted = reg.Replace(strTainted, "")
下面的代碼是除去|操作符後的所有文本：
Set reg = New RegExp
reg.Pattern = "^(.+)\|(.)"Anycharacter from the start of
' the string to a | character.
strUnTainted = reg.Replace(strTainted, "$1")
```

另外，當使用"檔系統物件"元件打開或創建檔時，如果檔案名取自用戶的輸入內容，那麼很可能被攻擊者利用去試圖打開一個電腦的串口或者印表機設備。為了防止這個問題發生，我們可以使用下面的 Jscript 代碼除去非法檔案名：

```
var strOut = strIn.replace(/(AUX|PRN|NUL|COM\d|LPT\d)+\s*$/i, "");
```

從以上幾個簡單的實例，我們可以看到腳本引擎版本 5 處理字串的強大功能。微軟站點有關於腳本應用的詳細文檔與常式，位址如下：

文檔：<http://msdn.microsoft.com/scripting/default.htm>

常式：

<http://msdn.microsoft.com/workshop/languages/clinic/scripting051099.asp>

禁止父路徑 (Parent Paths) 表達法

默認情況下，可以在函數中使用父路徑".."。但是為了安全考慮，我們應該禁止這個功能，步驟如下：

1. 在 "Internet 服務管理器" 中右鍵點擊要處理伺服器的 "默認 Web 站點" , 從彈出功能表中選擇 "屬性 "
2. 點擊 "主目錄" 選項卡
3. 點擊 "配置"
4. 點擊 "應用程式選項" 選項卡
5. 去除 "啟用父路徑" 核取方塊

禁止在頁面檔的 Content-Location 中包含 IP 位址資訊

頁面檔的 Content-Location 檔頭可能會暴露內部位址資訊，這些資訊通常是隱藏或偽裝形式的，位於 NAT 防火牆後面或者代理伺服器後面。所以，建議禁止在這個位置存放 IP 位址資訊。關於禁止的詳細方法，請參閱 Microsoft 網站下列地址：<http://support.microsoft.com/support/kb/articles/Q218/1/80.ASP>

附錄 B：保護你的 Windows NT/2000 機器避免網路攻擊

經過時間的洗禮，Windows NT/2000 對於 TCP/IP 處理能力是相當不錯的，其預設值多半可以阻擋大部分的網路攻擊。如果想要降低網路攻擊帶來的傷害並減少網路漏洞的話，下列幾個修改 TCP/IP 登錄設定的方法則可以加以參考。

- 設定路由器(routers)只允許相關伺服器使用的必須通訊埠(port)
- 停止不需要的網路服務
- 啟用 TCP/IP 安全過濾設定，限制可以存取的 port (請參考 Microsoft Knowledge Base 編號 Q150543 文章，該文章列出 Window 網路服務所使用的 port)
- 除非有必要請停用 NetBIOS over TCP/IP。 .
- 使用固定 IP。
- 如果你的機器執行 IIS 伺服器的話請參閱 IIS security checklist
- 經常性閱讀 Microsoft 安全網站或是其他與網路安全相關的網站。

此外藉由修改登錄檔 TCP/IP 相關設定以達到最大保護能力，避免來自網路的攻擊

以下列出的登錄檔設定值將有效的增加 Windows NT 或 Windows 2000 對於網路上拒絕服務(denial of service)防禦能力。

登錄檔相關位置：

Tcpip\Parameters

數值名稱 SynAttackProtect

數值型態 REG_DWORD

數值範圍 0, 1, 2

預設值 0

建議值 2

說明 這項保護設定包含減少 SYN-ACKS 信號重新傳輸的數量，也可以減少系統資源持續鎖定的時間

登錄檔相關位置：

Tcpip\Parameters

數值名稱 TcpMaxHalfOpen

數值型態 REG_DWORD-number

數值範圍 100 - 0xFFFF

預設值 100 (Professional, Server), 500 (advanced server)

建議值 同上

說明 適用於 Windows 2000, 我尚未在 Windows NT 測試

登錄檔相關位置：

Tcpip\Parameters

數值名稱 TcpMaxHalfOpenRetried

數值型態 REG_DWORD

數值範圍 80-0xFFFF

預設值 80 (Professional, Server), 400 (Advanced Server)

建議值 同上

說明 適用於 Windows 2000, 尚未在 Windows NT 測試

登錄檔相關位置：

Tcpip\Parameters

數值名稱 EnablePMTUDiscovery

數值型態 REG_DWORD—Boolean

數值範圍 0, 1(False, True)

預設值 1

建議值 0

說明 設定為 1 的時候，TCP 將嘗試尋找遠端機器連線路徑中最適合 MTU(Maximum Transmission Unit)大小，藉由尋找 MTU 與限制 TCP 節區(segments)的大小，TCP 可以減少封包通過連線路徑上具有不同 MTU 大小的路由器所產生的斷碎(fragmentation)情形。封包斷碎將會嚴重影響 TCP 傳輸能力。

設定為 0 將會使 Windows NT/2000 處理所有網路連線時候均使用 576bytes 大小的 MTU(不包含同一網路遮罩的區域網路)

登錄檔相關位置：

Netbt\Parameters

數值名稱 NoNameReleaseOnDemand

數值型態 REG_DWORD—Boolean

數值範圍 0, 1 (False, True)

預設值 0

建議值 1

說明 這項參數可以控制：當接收倒來自網路上的 name-release 要求，可以決定是否釋放出電腦的 NetBIOS 名稱。對於 Administrator 來說本設定應該被加入在登錄檔中，以保護電腦避免 malicious name-release 攻擊。

登錄檔相關位置：

Tcpip\Parameters

數值名稱 EnableDeadGWDetect

數值型態 REG_DWORD—Boolean

數值範圍 0, 1 (False, True)

預設值 1

建議值 0

登錄檔相關位置：

Tcpip\Parameters

數值名稱 KeepAliveTime

數值型態 REG_DWORD—Time in milliseconds

數值範圍 1 – 0xFFFFFFFF

預設值 7,200,000 (two hours)

建議值 300,000

登錄檔相關位置：

Tcpip\Parameters\Interfaces\

數值名稱 PerformRouterDiscovery

數值型態 REG_DWORD

數值範圍 0,1,2

預設值 2

建議值 0

登錄檔相關位置：

Tcpip\Parameters

數值名稱 EnableICMPRedirects

數值型態 REG_DWORD

數值範圍 0, 1 (False, True)

預設值 1

建議值 0

附錄 C ASP 常用函式

Array()

FUNCTION: 傳回一個數組

SYNTAX: Array(list)

ARGUMENTS: 字元, 數字均可

EXAMPLE: <%

```
Dim myArray()  
For i = 1 to 7  
    Redim Preserve myArray(i)  
    myArray(i) = WeekdayName(i)  
Next  
>%
```

RESULT: 建立了一個包含 7 個元素的數組 myArray
myArray("Sunday", "Monday", "Saturday")

CInt()

FUNCTION: 將一個表達式轉化為數字類型

SYNTAX: CInt(expression)

ARGUMENTS: 任何有效的字元均可

EXAMPLE: <%

```
f = "234"  
response.write cINT(f) + 2  
>%
```

RESULT: 236

轉化字元 "234" 為數字 "234", 如果字元串為空, 則傳回 0 值

CreateObject()

FUNCTION: 建立和傳回一個已註冊的 ACTIVEX 元件的實例。

SYNTAX: CreateObject(objName)

ARGUMENTS: objName 是任何一個有效、已註冊的 ACTIVEX 元件的名字。

EXAMPLE: <%

```
Set con = Server.CreateObject("ADODB.Connection")  
>%
```

RESULT:

CStr()

FUNCTION: 轉化一個表達式為字元串。

SYNTAX: CStr(expression)

ARGUMENTS: expression 是任何有效的表達式。

EXAMPLE: <%

```
s = 3 + 2
```

```
response.write "The result is: " & cStr(s)
```

```
%>
```

RESULT: 轉化數字「5」為字元「5」。

Date()

FUNCTION: 傳回目前系統日期。

SYNTAX: Date()

ARGUMENTS: None.

EXAMPLE: <%=Date%>

RESULT: 8/4/99

DateAdd()

FUNCTION: 傳回一個被改變了的日期。

SYNTAX: DateAdd(timeinterval,number,date)

ARGUMENTS: timeinterval is the time interval to add; number is amount of time intervals to add; and date is the starting date.

EXAMPLE: <%

```
currentDate = #8/4/99#
```

```
newDate = DateAdd("m",3,currentDate)
```

```
response.write newDate
```

```
%>
```

```
<%
```

```
currentDate = #12:34:45 PM#
```

```
newDate = DateAdd("h",3,currentDate)
```

```
response.write newDate
```

```
%>
```

RESULT: 11/4/99

3:34:45 PM

"m" = "month";

"d" = "day";

If currentDate is in time format then,

"h" = "hour";

"s" = "second";

DateDiff()

FUNCTION: 傳回兩個日期之間的差值。

SYNTAX: DateDiff(timeinterval,date1,date2 [, firstdayofweek][, firstweekofyear]])

ARGUMENTS: timeinterval 表示相隔時間的類型，如「M」表示「月」。

EXAMPLE: <%

```
fromDate = #8/4/99#
```

```
toDate = #1/1/2000#
```

```
response.write "There are " & _  
    DateDiff("d",fromDate,toDate) & _  
    " days to millenium from 8/4/99."
```

%>

RESULT: 從 8/4/99 到 2000 年還有 150 天。

Day()

FUNCTION: 傳回一個月的第幾日。

SYNTAX: Day(date)

ARGUMENTS: date 是任何有效的日期。

EXAMPLE: <%=Day(#8/4/99#)%>

RESULT: 4

FormatCurrency()

FUNCTION: 傳回表達式，此表達式已被格式化為貨幣值

SYNTAX: FormatCurrency(Expression [, Digit][, LeadingDigit][, Paren][, GroupDigit]]])

ARGUMENTS: Digit 指示小數點右側顯示位數的數值。預設值為 -1，指示使用的是電腦的區域設定；LeadingDigit 三態常數，指示是否顯示小數值小數點前面的零。

EXAMPLE: <%=FormatCurrency(34.3456)%>

RESULT: \$34.35

FormatDateTime()

FUNCTION: 傳回表達式，此表達式已被格式化為日期或時間

SYNTAX: FormatDateTime(Date, [, NamedFormat])

ARGUMENTS: NamedFormat 指示所使用的日期/時間格式的數值，如果省略，則

使用

vbGeneralDate.

EXAMPLE: <%=FormatDateTime("08/4/99", vbLongDate)%>

RESULT: Wednesday, August 04, 1999

FormatNumber()

FUNCTION: 傳回表達式，此表達式已被格式化為數值。

SYNTAX: FormatNumber(Expression [, Digit][, LeadingDigit][, Paren][, GroupDigit]]]]])

ARGUMENTS: Digit 指示小數點右側顯示位數的數值。預設值為 -1，指示使用的是

電腦的區域設定。； LeadingDigit i 指示小數點右側顯示位數的數值。預設值為 -

1，指示使用的是電腦的區域設定。； Paren 指示小數點右側顯示位數的數值。

預設

值為 -1，指示使用的是電腦的區域設定。； GroupDigit i 指示小數點右側顯示位數

的數值。預設值為 -1，指示使用的是電腦的區域設定。。

EXAMPLE: <%=FormatNumber(45.324567, 3)%>

RESULT: 45.325

FormatPercent()

FUNCTION: 傳回表達式，此表達式已被格式化為尾隨有 % 符號的百分比(乘以 100)。 (%)

SYNTAX: FormatPercent(Expression [, Digit][, LeadingDigit][, Paren][, GroupDigit]]]]])

ARGUMENTS: 同上。

EXAMPLE: <%=FormatPercent(0.45267, 3)%>

RESULT: 45.267%

Hour()

FUNCTION: 以 24 時傳回小時數。

SYNTAX: Hour(time)

ARGUMENTS:

EXAMPLE: <%=Hour("#4:45:34 PM#)%>

RESULT: 16

(Hour has been converted to 24-hour system)

Instr()

FUNCTION: 傳回字元或字元串在另一個字元串中第一次出現的位置。

SYNTAX: Instr([start,] strToBeSearched, strSearchFor [, compare])

ARGUMENTS: Start 為搜尋的起始值, strToBeSearched 接受搜尋的字元串
strSearchFor 要搜尋的字元.compare 比較方式 (詳細見 ASP 常數)

EXAMPLE: <%

```
strText = "This is a test!!"
```

```
pos = Instr(strText, "a")
```

```
response.write pos
```

%>

RESULT: 9

InstrRev()

FUNCTION: 同上, 只是從字元串的最後一個搜尋起

SYNTAX: InstrRev([start,] strToBeSearched, strSearchFor [, compare])

ARGUMENTS: 同上.

EXAMPLE: <%

```
strText = "This is a test!!"
```

```
pos = InstrRev(strText, "s")
```

```
response.write pos
```

%>

RESULT: 13

Int()

FUNCTION: 傳回數值類型, 不四捨五入, 注意取值是不大於它的整數。

SYNTAX: Int(number)

ARGUMENTS:

EXAMPLE: <%=INT(32.89)%> <%=int(-3.33)%>

RESULT: 32 -4

IsArray()

FUNCTION: 判斷一對象是否為數組, 傳回布爾值 .

SYNTAX: IsArray(name)

ARGUMENTS:

EXAMPLE: <%

```
strTest = "Test!"
```

```
response.write IsArray(strTest)
```

%>

RESULT: False

IsDate()

FUNCTION: 判斷一對象是否為日期，傳回布爾值

SYNTAX: IsDate(expression)

ARGUMENTS: expression is any valid expression.

EXAMPLE: <%

strTest = "8/4/99"

response.write IsDate(strTest)

%>

RESULT: True

IsEmpty()

FUNCTION: 判斷一對象是否初始化，傳回布爾值。

SYNTAX: IsEmpty(expression)

ARGUMENTS:

EXAMPLE: <%

Dim i

response.write IsEmpty(i)

%>

RESULT: True

IsNull()

FUNCTION: 判斷一對象是否為空，傳回布爾值。

SYNTAX: IsNull(expression)

ARGUMENTS:

EXAMPLE: <%

Dim i

response.write IsNull(i)

%>

RESULT: False

IsNumeric()

FUNCTION: 判斷一對象是否為數字，傳回布爾值。

SYNTAX: IsNumeric(expression)

ARGUMENTS:

EXAMPLE: <%

```
i = "345"
response.write IsNumeric(i)
%>
RESULT: True
就算數字加了引號，ASP 還是認為它是數字。
```

```
IsObject()
FUNCTION: 判斷一對象是否為對象，傳回布爾值。
SYNTAX: IsObject(expression)
ARGUMENTS:
EXAMPLE: <%
Set con = Server.CreateObject("ADODB.Connection")
response.write IsObject(con)
%>
RESULT: True
```

```
LBound()
FUNCTION: 傳回指定數組維的最小可用下標。
SYNTAX: Lbound(arrayname [, dimension])
ARGUMENTS: ; dimension 指明要傳回哪一維下界的整數。使用 1 表示第一維，
2
表示第二維，以此類推。如果省略 dimension 參數，預設值為 1。
EXAMPLE: <%
i = Array("Monday", "Tuesday", "Wednesday")
response.write LBound(i)
%>
RESULT: 0
```

```
LCase()
FUNCTION: 傳回字元串的小寫形式
SYNTAX: Lcase(string)
ARGUMENTS: string is any valid string expression.
EXAMPLE: <%
strTest = "This is a test!"
response.write LCase(strTest)
%>
RESULT: this is a test!
```

Left()

FUNCTION: 傳回字元串左邊第 length 個字元以前的字元 (含第 length 個字元).

SYNTAX: Left(string, length)

ARGUMENTS:

EXAMPLE: <%

```
strTest = "This is a test!"  
response.write Left(strTest, 3)  
%>
```

RESULT: Thi

Len()

FUNCTION: 傳回字元串的長度.

SYNTAX: Len(string | varName)

ARGUMENTS:

EXAMPLE: <%

```
strTest = "This is a test!"  
response.write Len(strTest)  
%>
```

RESULT: 15

LTrim()

FUNCTION: 去掉字元串左邊的空格.

SYNTAX: LTrim(string)

ARGUMENTS:

EXAMPLE: <%

```
strTest = " This is a test!"  
response.write LTrim(strTest)  
%>
```

RESULT: This is a test!

Mid()

FUNCTION: 傳回特定長度的字元串(從 start 開始,長度為 length).

SYNTAX: Mid(string, start [, length])

ARGUMENTS:

EXAMPLE: <%

```
strTest = "This is a test! Today is Monday."  
response.write Mid(strTest, 17, 5)
```


%>

RESULT: Today

Minute()

FUNCTION: 傳回時間的分釧.

SYNTAX: Minute(time)

ARGUMENTS:

EXAMPLE: <%=Minute(#12:45:32 PM#)%>

RESULT: 45

Month()

FUNCTION: 傳回日期.

SYNTAX: Month(date)

ARGUMENTS: date is any valid date expression.

EXAMPLE: <%=Month(#08/04/99#)%>

RESULT: 8

MonthName()

FUNCTION: Returns a string identifying the specified month.

SYNTAX: MonthName(month, [, Abb])

ARGUMENTS: month is the numeric representation for a given month; Abb (optional) is a boolean value used to display month abbreviation. True will display the abbreviated month name and False (default) will not show the abbreviation.

EXAMPLE: <%=MonthName(Month(#08/04/99#))%>

RESULT: August

Now()

FUNCTION: Returns the current system date and time.

SYNTAX: Now()

ARGUMENTS: None

EXAMPLE: <%=Now%>

RESULT: 8/4/99 9:30:16 AM

Replace()

FUNCTION: Returns a string in which a specified sub-string has been replaced with another substring a specified number of times.

SYNTAX: Replace(strToBeSearched, strSearchFor, strReplaceWith [,

start

```
][, count ][, compare]]])
```

ARGUMENTS: strToBeSearched is a string expression containing a sub-string to be replaced; strSearchFor is the string expression to search for

within strToBeSearched; strReplaceWith is the string expression to replace

sub-string strSearchFor; start (optional) is the numeric character position to begin search; count (optional) is a value indicating the comparison constant.

EXAMPLE: <%

```
strTest = "This is an apple!"
```

```
response.write Replace(strTest, "apple", "orange")
```

```
%>
```

RESULT: This is an orange!

Right()

FUNCTION: 傳回字元串右邊第 length 個字元以前的字元 (含第 length 個字元).

SYNTAX: Right(string, length)

ARGUMENTS: .

EXAMPLE: <%

```
strTest = "This is an test!"
```

```
response.write Right(strTest, 3)
```

```
%>
```

RESULT: st!

Rnd()

FUNCTION: 產生一個隨機數.

SYNTAX: Rnd [(number)]

ARGUMENTS:

EXAMPLE: <%

Randomize()

```
response.write RND()
```

```
%>
```

RESULT: 任何一個在 0 到 1 之間的數

Round()

FUNCTION: 傳回按指定位數進行四捨五入的數值。

SYNTAX: Round(expression [, numRight])

ARGUMENTS: numRight 數字表明小數點右邊有多少位進行四捨五入。如果省略，則

Round 函數傳回整數。

EXAMPLE: <%

i = 32.45678

response.write Round(i)

%>

RESULT: 32

Rtrim()

FUNCTION: 去掉字元串右邊的字元串。

SYNTAX: Rtrim(string)

ARGUMENTS:

EXAMPLE: <%

strTest = "This is a test!! "

response.write RTrim(strTest)

%>

RESULT: This is a test!!

Second()

FUNCTION: 傳回秒。

SYNTAX: Second(time)

ARGUMENTS: .

EXAMPLE: <%=Second(#12:34:28 PM#)%>

RESULT: 28

StrReverse()

FUNCTION: 反排一字元串

SYNTAX: StrReverse(string)

ARGUMENTS:

EXAMPLE: <%

strTest = "This is a test!!"

response.write StrReverse(strTest)

%>

RESULT: !!tset a si sihT

Time()

FUNCTION: 傳回系統時間.

SYNTAX: Time()

ARGUMENTS: .

EXAMPLE: <%=Time%>

RESULT: 9:58:28 AM

Trim()

FUNCTION: 去掉字元串左右的空格.

SYNTAX: Trim(string)

ARGUMENTS: string is any valid string expression.

EXAMPLE: <%

```
strTest = " This is a test!! "
```

```
response.write Trim(strTest)
```

```
%>
```

RESULT: This is a test!!

UBound()

FUNCTION: 傳回指定數組維數的最大可用下標.

SYNTAX: Ubound(arrayname [, dimension])

ARGUMENTS: ; dimension (optional) 指定傳回哪一維上界的整數。1 表示第一

維，2 表示第二維，以此類推。如果省略 dimension 參數，則預設值為 1.

EXAMPLE: <%

```
i = Array("Monday", "Tuesday", "Wednesday")
```

```
response.write UBound(i)
```

```
%>
```

RESULT: 2

UCase()

FUNCTION: 傳回字元串的大寫形式.

SYNTAX: UCase(string)

ARGUMENTS:

EXAMPLE: <%

```
strTest = "This is a test!!"
```

```
response.write UCase(strTest)
```

```
%>
```

RESULT: THIS IS A TEST!!

VarType()

FUNCTION: 傳回指示變數子類型的值

SYNTAX: VarType(varName)

ARGUMENTS:

EXAMPLE: <%

```
i = 3
```

```
response.write varType(i)
```

```
%>
```

RESULT: 2(數字)詳見"asp 常數"

WeekDay()

FUNCTION: 傳回在一周的第幾天.

SYNTAX: WeekDay(date [, firstdayofweek])

ARGUMENTS: .

EXAMPLE: <%

```
d = #8/4/99#
```

```
response.write Weekday(d)
```

```
%>
```

RESULT: 4(星期三)

WeekDayName()

FUNCTION: 傳回一周第幾天的名字.

SYNTAX: WeekDayName(weekday [, Abb][, firstdayofweek])

ARGUMENTS: Abb 可選。Boolean 值, 指明是否縮寫表示星期各天的名稱。如果省

略, 預設值為 False, 即不縮寫星期各天的名稱。firstdayofweek 指明星期第一天的

數值

EXAMPLE: <%

```
d = #8/4/99#
```

```
response.write WeekdayName(Weekday(d))
```

```
%>
```

RESULT: Wednesday

Year()

FUNCTION: 傳回目前的年份.

SYNTAX: Year(date)

ARGUMENTS:

EXAMPLE: <%=Year(#8/4/99#)%>

RESULT: 1999