

## 3-2 虛擬主機類型

上個小節我們說明了使用虛擬主機的優點和考量等內容，接下來在這個小節中本書將介紹架構虛擬主機時可供選擇的不同類型網站，其中包含的類型有：IP-based 虛擬主機、Name-based 虛擬主機與 Port-based 虛擬主機等。

### ➤ IP-based 虛擬主機

所謂「IP-based 虛擬主機」是指使用同一個組態設定檔來管理多個網站（虛擬主機），這是和「獨立伺服器」最大的不同，而所有的虛擬主機雖然位於同一電腦中，但彼此間使用的 IP 位址都不同。不僅所有虛擬主機可共用組態內容，而且也能共享伺服器的「程序集區」(Process Pool)，所以更易於管理，同時也能提供用戶端較快的回應時間。

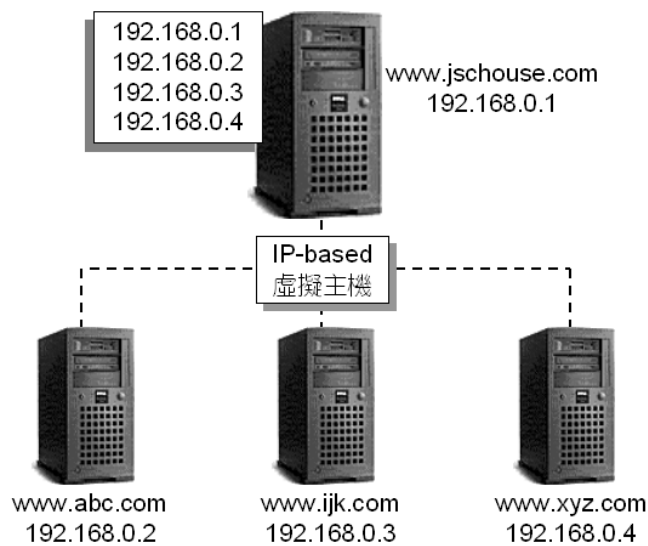


圖 3-2 IP-based 虛擬主機

但此架構的缺點是需要較多的 IP 位址(因為每個虛擬主都使用不同的 IP 位址)、較多的網路卡，或是將多個 IP 位址「連結」(Binding)在同一張網路卡上(這就是所謂的 IP Alias)，所幸目前較新的作業系統，例如 Linux，都可支援這些功能。

### ➤ Name-based 虛擬主機

為了解決 IP-based 虛擬主機消耗過多 IP 位址的缺點，因此在 Apache 伺服器中支援「Name-based 虛擬主機」的功能，此類型的虛擬主機允許多個網站共用一個 IP 位址，因此可大大降低對 IP 位址的需求

但是 Name-based 虛擬主機的方式只支援使用 HTTP/1.1 瀏覽器的用戶端，因為它需要利用 HTTP/1.1 通訊協定中的「Host」表頭功能，以使得 Apache 伺服器可識別用戶端要求的目標主機，所幸目前只有不到 1%的用戶端使用 HTTP/1.1 之前版本的瀏覽器，因此在支援度的部份應該不成問題。

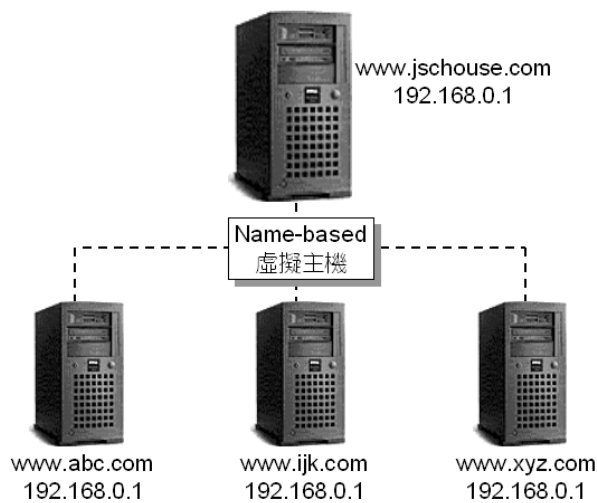


圖 3-3 Name-based 虛擬主機

Name-based 虛擬主機的設計原理很簡單，只要在 DNS 伺服器中，將每個虛擬主機名稱都對應到同一個 IP 位址，然後再到 Apache 伺服器上的 httpd.conf 檔案中，設定所有的 Name-based 虛擬主機都使用同一個 IP 位址，如此便完成所有的設定。

由以上的說明可以了解，使用 Name-based 虛擬主機的方式可以解決建立 IP-based 虛擬主機時的問題，但是在以下的情形時，建議仍然使用 IP-based 虛擬主機為宜：

- ◆ 用戶端瀏覽器若是使用較早期的 HTTP 標準，例如 HTTP/1.0，則無法連接 Name-based 虛擬主機，因為瀏覽器在連接至 Name-based 虛擬主機時，必須傳送 HTTP 主機表頭，這就必須使用 HTTP/1.1 的標準。
- ◆ Name-based 虛擬主機無法使用 SSL 功能，這是由於 SSL 通訊協定本身的限制。
- ◆ 有些作業系統及網路設備使用特殊的頻寬管理技術，因而無法在同一個 IP 位址上區分不同的主機。

### ➤ Port-based 虛擬主機

所謂的 Port-based 虛擬主機是指，在相同的 IP 位址或網域名稱上，由不同連接埠所建立的虛擬主機，因此利用此方式可快速的建立多個虛擬主機。以下是一些 Port-based 虛擬主機的可能範例：

- ◆ 192.168.0.1
- ◆ 192.168.0.1 : 81
- ◆ 192.168.0.1 : 800
- ◆ 192.168.0.1 : 800

**NOTE** 若沒有在 IP 位址或網域名稱後指定連接埠號碼，則表示使用 HTTP 的預設 80 連接埠。

但通常這種方式的虛擬主機都是屬於私人或實驗性質的用途居多，因為每一個位址可允許使用的連接埠有 65536 個（0 ~ 65535），而 HTTP 預設的連接埠為 80，所以若是任意變更 Apache 使用的連接埠號碼，則一般的用戶端並無法成功進行連接（除非得知確定的連接埠號碼）。

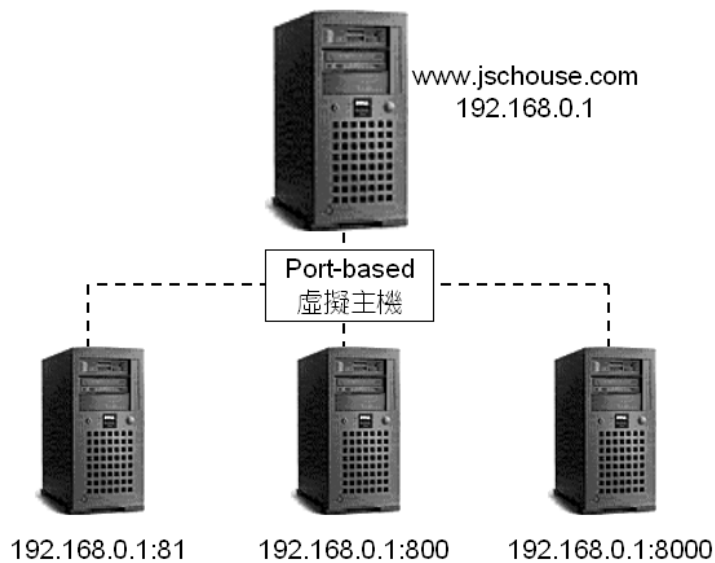


圖 3-4 Port-based 虛擬主機

### 3-3 虛擬主機內容選項

其實在設定虛擬主機的組態內容時，可用選項並不多，其中包含的有：DocumentRoot、NameVirtualHost、ServerAlias、ServerName、ServerPath 和 VirtualHost 等。因為有些選項已在先前的小節中介紹過，所以在此我們將說明其他選項的設定方法，以便在下個小節架設虛擬主機之用。

#### ➤ NameVirtualHost

「NameVirtualHost」是設定 Name-based 虛擬主機的必要選項，因為它可用來指定虛擬主機使用的 IP 位址，所以不建議在此直接以主機名稱替代 IP 位址。以下是一個指定虛擬主機 IP 位址的範例：

```
NameVirtualHost 192.168.0.1
```

在某些情形下，您可能希望虛擬主機使用特定的連接埠號碼，例如尚在測試中或專供特定人員使用的 Web 伺服器等，此時也可以使用「NameVirtualHost」選項來支援連接埠的指定。以下是一個指定虛擬主機 IP 位址及連接埠號碼的範例：

```
NameVirtualHost 192.168.0.1 : 8080
```

在 1.3.13 版之後的 Apache 伺服器，允許使用萬用字元「\*」來取代個別 IP 位址的輸入，它表示連接主機上所有的可用 IP 位址，這可節省逐一輸入的時間。以下是一個使用萬用字元取代 IP 位址的範例：

```
NameVirtualHost *
```

## ➤ ServerAlias

「ServerAlias」也是設定 Name-based 虛擬主機的必要選項，但它只適用於 1.1 版之後的 Apache 伺服器。此選項可以為虛擬主機新增其他的別名，例如 2 個同時使用 192.168.0.1 位址所建立的 Name-based 虛擬主機，可以分別使用 `www.jschouse.com` 和 `www.second.com` 的別名，但需注意一點，除了在此處指定伺服器的別名外，這 2 個別名也必須於 DNS 伺服器中登錄（A 或 CNAME 資源記錄）。以下是一個新增虛擬主機別名的範例：

```
ServerAlias w2.jschouse.com
```

## ➤ VirtualHost

`<VirtualHost>`和`</VirtualHost>`區塊是用來包含多個在虛擬主機上使用的選項，當伺服器收到用戶端對於特定虛擬主機上的文件要求時，它就會使用包含在`<VirtualHost>`和`</VirtualHost>`之間的組態內容。以下是使用的語法及說明：

```
<VirtualHost addr[:port] [addr[:port]] ...> ... </VirtualHost>
```

在語法中的「addr」可以使用以下二種方法來表示：

- ◆ 虛擬主機的 IP 位址
- ◆ 虛擬主機的主機名稱

以下的範例表示虛擬主機只使用單一 IP 位址：

```
<VirtualHost 192.168.0.1>  
ServerAdmin webmaster@jschouse.com  
DocumentRoot /www/docs/jschouse  
ServerName www.jschouse.com  
ErrorLog logs/jschouse-error_log  
</VirtualHost>
```

以下的範例表示虛擬主機同時使用多個 IP 位址：

```
<VirtualHost 192.168.0.1 192.168.0.2>
DocumentRoot /www/docs/jschouse
ServerName www.jschouse.com
ServerAlias sales
</VirtualHost>
```

### 3-4 虛擬主機架設

在前幾個小節中，本書介紹許多與虛擬伺服器有關的內容，接下來在這個小節我們將進行虛擬主機的實際架設，本書在此將以 3 種不同的範例來說明各類虛擬主機的架設方法，這些包括：

- ◆ IP-based 虛擬主機架設
- ◆ Name-based 虛擬主機架設
- ◆ Port-based 虛擬主機架設

因為以下的範例都是使用同一張乙太網路卡，所以爲了方便說明起見，我們先在此網路卡上新增其他的 IP 位址（IP Alias），在新增 IP 位址時可以使用下面的指令：

```
[root@ns1 root]# ifconfig eth0 : 1 192.168.0.121 netmask 255.255.255.0 up
[root@ns1 root]# ifconfig eth0 : 2 192.168.0.122 netmask 255.255.255.0 up
[root@ns1 root]# ifconfig eth0 : 3 192.168.0.123 netmask 255.255.255.0 u p
```

在新增並啓用 IP Alias 後，最好能使用「ping」指令以確定設定內容是否無誤：

```
[root@ns1 root]# ping -c 2 192.168.0.123
PING 192.168.0.123 ( 192.168.0.123 ) from 192.168.0.123 : 56 ( 84 ) bytes of data.
64 bytes from 192.168.0.123 : icmp_seq=1 ttl=64 time=0.087 ms
64 bytes from 192.168.0.123 : icmp_seq=2 ttl=64 time=0.066 ms

--- 192.168.0.123 ping statistics ---
2 packets transmitted, 2 received, 0% loss, time 1000ms
rtt min/avg/max/mdev = 0.066/0.076/0.087/0.013 ms
```

**NOTE** 有關 IP Alias 的內容請參考「1-4-1 設定網路卡組態 – ifconfig 指令」小節說明。

## ➤ IP-based 虛擬主機架設

以下本書將以三種不同的範例，來說明架設 IP-based 虛擬主機時可能出現不同的情況：

### ● 範例一

主機上有二個可用 IP 位址：192.168.0.1 和 192.168.0.121，它們對應的主機名稱分別為 `www.jschouse.com`（主要伺服器）與 `www.virtual.com`（虛擬主機）。以下是符合此情況的範例：

```
[root@ns1 root]# vi /etc/httpd/conf/httpd.conf

DocumentRoot "/var/www/html"
ServerName www.jschouse.com

<VirtualHost 192.168.0.121>
    DocumentRoot "/var/www/virtual"
    ServerName www.virtual.com
</VirtualHost>
```



在上例中，使用 `www.virtual.com` 的名稱只能連接到 `192.168.0.121` 的虛擬主機，而使用 `www.jschouse.com` 來連接，也只能存取主要伺服器（`192.168.0.1`）上的資源。

**NOTE** 若要使用以上二個主機名稱來進行連接，則必須先在 DNS 伺服器中新增這二個主機名稱的 A 資源記錄，若是沒有新增此記錄，仍可使用 IP 位址的方式連接。

## ● 範例二

主機上有二個可用 IP 位址：`192.168.0.1` 和 `192.168.0.121`，它們對應的主機名稱分別為 `www.jschouse.com` 與 `www.virtual.com`，但是沒有指定何者為主要伺服器（非虛擬主機）。以下是符合此情況的範例：

```
ServerName www.jschouse.com

<VirtualHost 192.168.0.1>
    DocumentRoot "/var/www/jschouse"
    ServerName www.jschouse.com
</VirtualHost>

<VirtualHost 192.168.0.121>
    DocumentRoot "/var/www/virtual"
    ServerName www.jschouse.com
</VirtualHost>
```

在以上的範例中，因為所有的 IP 位址都被指定為虛擬伺服器所使用，因此並沒有主要伺服器。

### ● 範例三

主機上有二個可用 IP 位址：192.168.0.1 和 192.168.0.121，它們對應的主機名稱分別為 `www.jschouse.com` 與 `www-cache.virtual.com`，但是沒有指定何者為主要伺服器（非虛擬主機）。前者監聽連接埠 80 的要求，而 `www-cache.virtual.com` 上啟用代理伺服器快取（Proxy Cache）的功能，並且監聽連接埠 8080 的交通。以下是符合此情況的範例內容：

```
<VirtualHost 192.168.0.1 : 80>
    DocumentRoot /var/www/jschouse
    ServerName www.jschouse.com
</VirtualHost>

<VirtualHost 192.168.0.121 : 8080>
    DocumentRoot /var/www/cache
    ServerName www-cache.virtual.com

    <Directory proxy : >
        Order Deny , Allow
        Deny from all
        Allow from 192.168.0.0/25
    </Directory>
</VirtualHost>
```

在以上的範例中，所有的 IP 位址都被指定為虛擬伺服器所使用，因此沒有主要伺服器。用戶端只能使用連接埠 80 來連接至 Web 伺服器，而代理伺服器則使用連接埠 8080。

## ➤ Name-based 虛擬主機架設

假設 Apache 伺服器的主機名稱為 `www.jschouse.com`，同時它也擁有二個別名（Aliases）－`sales.jschouse.com` 與 `backup.jschouse.com`，而這些別名都使用同一個 IP 位址（`192.168.0.1`）。以下是符合此情況的設定範例：

```
ServerName www.jschouse.com

<VirtualHost 192.168.0.1>
    DocumentRoot /var/www/sales
    ServerName sales.jschouse.com
</VirtualHost>

<VirtualHost 192.168.0.1>
    DocumentRoot /var/www/backup
    ServerName backup.jschouse.com
</VirtualHost>
```

在此範例中，雖然三個 Web 網站使用的 IP 位址都相同，但是透過 Name-based 虛擬主機的架設，就可得到不同的網頁內容，其中最重要的部份就在 DNS 伺服器中的設定。因為我們會在第十一章詳細說明 DNS 伺服器的使用，所以本書在此僅列出設定的步驟，其方法如下：

**Step 1** 建立正解區域，這個部份需要在 DNS 中的 `/etc/named.conf` 檔案新增以下的內容：

```
zone "jschouse.com" IN {
    type master;
    file "jschouse.hosts";
/* 正解區域設定檔「jschouse.hosts」是筆者自訂的區域設定檔，它位於
/var/named 目錄中。*/
}
```

**Step 2** 自行建立正解區域設定檔－jschouse.hosts（此為筆者自訂的檔案名稱），以下是本書提供的範例：

```
[root@ns1 root]# vi /var/named/jschouse.hosts

@                IN SOA    jschouse.com dns.jschouse.com (
                                2002080201    ; serial
                                86400           ; refresh
                                3600            ; retry
                                3600000        ; expiry
                                1D             ; minimum

@ IN NS jschouse.com

localhost        IN A 127.0.0.1
```

**Step 3** 新增資源記錄，以下是本書提供的範例：

```
www              IN  A       192.168.0.1
sales           IN  CNAME  www
backup          IN  CNAME  www
```

**Step 4** 重新啟動 BIND 伺服器，您可使用以下的方法：

```
[root@ns1 root]# /etc/rc.d/init.d/named restart
停止 named： 確定 ]
```