第 2 篇
主機的簡易資安防護措施

有很多團體做過許多作業系統安全性偵測的研究，他們發現一部沒有經過更新與保護的 Linux/Windows 主機(不論是一般個人電腦還是伺服器)，只要一接上 Internet 幾乎可以在數小時以內就被入侵或被當成跳板！您瞧瞧，這是啥世界啊～所以說，要好好的保護好您自己的伺服器主機才行喔！那應該要如何保護你的伺服器主機呢？基本上，你最要知道的是你的伺服器開了多少網路服務，而這些服務會啟動什麼埠口？根據這層關係來關閉一些不必要的網路服務。再者，利用線上更新系統讓你的 Linux 隨時保持在最新的軟體的狀態，這個小動作可以預防絕大部分的入侵攻擊，可以說是最重要的一步了！最後才是架設基礎防火牆。

因為 Linux 的功能太強了，如果你不好好的保護好你的主機，要是被入侵並且被當成跳板，這可能會讓您吃上官司的！不要小看這層動作喔！雖然被入侵後只要將舊系統移除並且重灌後，你的伺服器主機就能夠『短暫』的恢復正常，不過如果您的一些操作習慣不改的話，呵呵，並不是重灌就能夠讓你的伺服器主機活的好好的喔！所以囉，我們在架站之前，基本的網路防備措施還是得來瞭解一下，免得三不五時要重灌、重灌、重灌....

7

網路安全與主機基本防護：限制埠口、網路升級與 SELinux

通過第一篇的鍛鍊之後，現在你應該已經利用 Linux 連上 Internet 了。但是你的 Linux 現在恐怕還是不怎麼安全的。因此，在開始伺服器設定之前，我們必須要讓你的系統強壯些！以避免被惡意的 cracker 所攻擊啊！在這一章當中，我們會介紹封包的流向，然後根據該流向來制訂系統強化的流程！包括線上自動升級、服務管控以及 SELinux 等等。現在就來瞭解瞭解囉！

7.1 網路封包連線進入主機的流程

在這一章當中，我們要討論的是，當來自一個網路上的連線要求想進入我們的主機時，這個網路封包在進入主機實際取得資料的整個流程是如何？瞭解了整個流程之後，你才會發現：**原來系統操作的基本概念是如此的重要**！而你也才會瞭解要如何保護你的主機安全的！閒話少說，咱們趕緊來瞧一瞧先。

7.1.1 封包進入主機的流程

在第一章我們就談過網路連線的流程，當時舉的例子是希望你可以理解為啥架設伺服器需要瞭解作業系統的基本觀念。在這一章當中，我們要將該流程更細緻化說明，因為，透過這個流程分析，你會知道為啥我們的主機需要進行過一些防護之後，系統才能夠比較強壯。此外，透過第二章的網路概念解釋後，你也瞭解了網路是雙向的，伺服器與用戶端都得要有IP:port才能夠讓彼此的軟體互相溝通。那麼現在，假設你的主機是WWW伺服器，透過下方的圖示，網路封包如何進入你的主機呢？

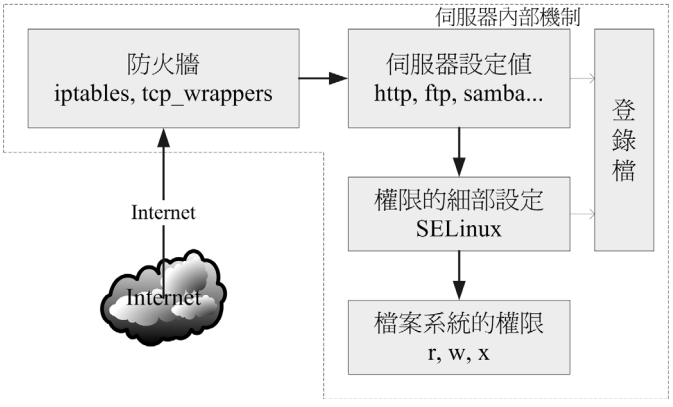

圖 7.1-1 網路封包進入主機的流程

1. **經過防火牆的分析：**

   Linux系統有內建的防火牆機制，因此你的連線能不能成功，得要先看防火牆的臉色才行。預設的Linux防火牆就有兩個機制，這兩個機制都是獨立存在的，因此我們預設就有兩層防火牆喔。第一層是封包過濾式的netfilter防火牆，另一個則是透過軟體控管的TCP Wrappers防火牆。

- **封包過濾防火牆：IP Filtering 或 Net Filter**

  要進入 Linux 主機的封包都會先通過 Linux 核心的預設防火牆，就是稱為 netfilter 的東東，簡單的說，就是 **iptables** 這個軟體所提供的防火牆功能。為何稱為封包過濾呢？因為他主要是分析 TCP/IP 的封包表頭來進行過濾的機制，主要分析的是 OSI 的第二、三、四層，主要控制的就是 MAC, IP, ICMP, TCP 與 UDP 的埠口與狀態 (SYN, ACK...) 等。詳細的資料我們會在第九章防火牆介紹。

- **第二層防火牆：TCP Wrappers**

  通過 netfilter 之後，網路封包會開始接受 Super daemons 及 TCP_Wrappers 的檢驗，那個是什麼呢？說穿了 **就是 /etc/hosts.allow 與 /etc/hosts.deny 的設定檔功能囉**。這個功能也是針對 TCP 的 Header 進行第二次的分析，同樣你可以設定一些機制來控制某些 IP 或 Port，好讓來源端的封包被丟棄或通過檢驗；

透過防火牆的管控，我們可以將大部分來自網際網路的垃圾連線丟棄，只允許目前開放的服務的連線進入主機而已，可以達到最基礎的安全防護。

2. **服務 (daemon) 的基本功能：**

   預設的防火牆是 Linux 的內建功能，但防火牆主要管理的是 MAC, IP, Port 等封包表頭方面的資訊，如果想要控管某些目錄可以進入，某些目錄則無法使用的功能，那就得要透過權限以及伺服器軟體提供的相關功能了。舉例來說，你可以在 httpd.conf 這個設定檔之內規範某些 IP 來源不能使用 httpd 這個服務來取得主機的資料，那麼即使該 IP 通過前面兩層的過濾，他依舊無法取得主機的資源喔！但要注意的是，**如果 httpd 這支程式本來就有問題的話，那麼 client 端將可直接利用 httpd 軟體的漏洞來入侵主機，而不需要取得主機內 root 的密碼！**因此，要小心這些啟動在網際網路上面的軟體喔！

3. **SELinux 對網路服務的細部權限控制：**

   為了避免前面一個步驟的權限誤用，或者是程式有問題所造成的資安狀況，因此 Security Enhanced Linux (安全強化 Linux) 就來發揮它的功能啦！簡單的說，SELinux 可以針對網路服務的權限來設定一些規則 (policy)，讓程式能夠進行的功能有限，因此即使使用者的檔案權限設定錯誤，以及程式有問題時，該程式能夠進行的動作還是被限制的，即使該程式使用的是 root 的權限也一樣。舉例來說，前一個步驟的 httpd 真的被 cracker 攻擊而讓對方取得 root 的使用權，由於 httpd 已經被 SELinux 控制在 /var/www/html 裡面，且能夠進行的功能已經被規範住了，因此 cracker 就無法使用該程式來進行系統的進一步破壞囉。現在這個 SELinux 一定要開啟喔！

4. **使用主機的檔案系統資源：**

   想一想，你使用瀏覽器連接到 WWW 主機最主要的目的是什麼？當然就是讀取主機的 WWW 資料啦！那 WWW 資料是誰？就是檔案啊！^_^！所以，最終網路封包其實是要向主機要求檔案系統的資料啦。我們這裡假設你要使用 httpd 這支程式來取得系統的檔

案資料，但 httpd 預設是由一個系統帳號名稱為 httpd 來啟動的，所以：**你的網頁資料的權限當然就是要讓 httpd 這支程式可以讀取**才行啊！如果你前面三關的設定都 OK，最終權限設定錯誤，使用者依舊無法瀏覽你的網頁資料的。

在這些步驟之外，我們的 Linux 以及相關的軟體都可能還會支援登錄檔記錄的功能，為了記錄系統程式在未來的錯誤查詢與入侵偵測，良好的分析登錄檔的習慣是一定要建立的，尤其是 /var/log/messages 與 /var/log/secure 這些個檔案！雖然各大主要 Linux distribution 大多有推出適合他們自己的登錄檔分析軟體，例如 CentOS 的 logwatch，不過由於該軟體並不見得適合所有的 distributions，所以鳥哥嘗試自己寫了一個 logfile.sh 的 shell script，你可以在下面的網址下載該程式：

- http://linux.vbird.org/download/index.php?action=detail&fileid=60

好了，那麼根據這些流程，你覺得 cracker 這些個壞傢伙能夠怎樣的攻擊我們的系統呢？得要先到對方想要怎麼破壞，我們才能夠想辦法來補強系統嘛！下一小節讓鳥哥來談談幾個常見的攻擊手法囉。

7.1.2 常見的攻擊手法與相關保護

我們由圖 7.1-1 瞭解到資料傳送到本機時所需要經過的幾道防線後，那個權限是最後的關鍵啦！現在你應該比較清楚為何我們常常在書本裡面一直談到**設定正確的權限可以保護你的主機**了吧？那麼 cracker 是如何透過上述的流程還能夠攻擊你的系統啊？下一小節就讓我們來分析分析。

- **取得帳號資訊後猜密碼**

由於很多人喜歡用自己的名字來作為帳號資訊，因此帳號的取得是很容易的！舉例來說，如果你的朋友將你的 email address 不小心洩漏出去，例如：dmtsai@your.host.name 之類的樣式，那麼人家就會知道你有一部主機，名稱為 your.host.name，且在這部主機上面會有一個使用者帳號，帳號名稱為 dmtsai，之後這個壞傢伙便利用某些特殊軟體例如 nmap 來進行你主機的 port scan 之後，嘿嘿！他就可以開始透過你主機有啟動的軟體功能來猜你這個帳號的密碼了！

另外，如果你常常觀察你的主機登錄檔，那你也會發現如果你的主機有啟動 Mail server 的服務時，你的登錄檔就會常常出現有些怪傢伙嘗試以一些奇怪的常見帳號在試圖猜測你的密碼，舉例來說像：admin, administrator, webmaster .... 之類的帳號，嘗試來竊取你的私人信件。如果你的主機真的有這類的帳號，而且這類的帳號還沒有良好的密碼規劃，那就容易『中標』！哎！真是麻煩！所以我們常講，**系統帳號千萬不能給予密碼，容易被猜密碼啊**！

這種猜密碼的攻擊方式算是最早期的入侵模式之一了，攻擊者知道你的帳號，或者是可以猜出來你的系統有哪些帳號，欠缺的就只是密碼而已，因此他會『很努力的』去猜你的密碼，此時，你的密碼規劃如果不好的話，很容易就被攻擊了！主機也很容易被綁架啊！所以，**良好的密碼設置習慣是很重要的。**

不過這種攻擊方式比較費時，因為目前很多軟體都有密碼輸入次數的限制，如果連續輸入三次密碼還不能成功的登入，那該次連線就會被斷線！所以，這種攻擊方式日益減少，目前偶而還會看到就是了！這也是初級 cracker 會使用的方式之一。那我們要如何保護呢？基本方式是這樣的：

- **減少資訊的曝光機會**：例如不要將 Email Address 隨意散佈到 Internet 上頭；
- **建立較嚴格的密碼設定規則**：包括 /etc/shadow, /etc/login.defs 等檔案的設定，建議你可以參考基礎篇內的帳號管理那一章來規範你的使用者密碼變更時間等等，如果主機夠穩定且不會持續加入某些帳號時，也可以考慮使用 chattr 來限制帳號 (/etc/passwd, /etc/shadow) 的更改；
- **完善的權限設定**：由於這類的攻擊方式會取得你的某個使用者帳號的登入權限，所以如果你的系統權限設定得宜的話，那麼攻擊者也僅能取得一般使用者的權限而已，對於主機的傷害比較有限啦！所以說，權限設定是重要的；

◆ **利用系統的程式漏洞『主動』攻擊**

由圖 7.1-1 裡面的第二個步驟中，我們知道如果你的主機有開放網路服務時，就必須有啟動某個網路軟體啊！我們也知道由於軟體可能撰寫方式的問題，可能存在一些會被 cracker 亂用的臭蟲程式碼，而這些臭蟲程式碼由於存在問題的大小，有分為 bug (臭蟲，可能會造成系統的不穩定或當機) 與 Security (安全問題，程式碼撰寫方式會導致系統的使用權限被惡意者所掌握) 等問題。

當程式的問題被公布後，某些較高階的 cracker 會嘗試撰寫一些針對這個漏洞的攻擊程式碼，並且將這個程式碼放置到 cracker 常去的網站上面，藉以拋銷自己的『功力』..... 鳥哥要提醒的是，這種程式碼**『是很容易被取得的』**。常見多『駭客資訊站 (台語，閒閒沒事幹之意)』取得這些程式碼後，他可能會想要『試一試這個攻擊程式的威力』，所以就拿來『掃射』一番，如果你八字比較輕，或者當天是學家說你比較倒楣時，可能就會被不小心的攻擊到...

這種攻擊模式是目前最常見的，因為攻擊者只要拿到攻擊程式就可以進行攻擊了，**『而且由攻擊開始到取得你系統的 root 權限不需要猜密碼，不需要兩分鐘，就能夠立刻入侵成功』**，所以『駭客資訊站』們最愛的就是這個時候了。但這個玩意兒本身是靠『你主機的程式漏洞』來攻擊的，所以，如果你的主機隨時保持在即時更新的階段，或者是關閉大部分不需要的程式，那就可以降避過這個問題。因此，你應該要這樣做：

- **關閉不需要的網路服務**：開的 port 越少，可以被入侵的管道越少，一部主機負責的服務越單純，越容易找出問題點。
- **隨時保持更新**：這個沒話講！一定要進行的！
- **關閉不需要的軟體功能**：舉例來說，後面會提到的遠端登入伺服器 SSH 可以提供 root 的遠端登入，那麼危險的事情當然要給他取消啊！^_^

◆ **利用社交工程作欺騙**

社交工程 (Social Engineering) 指的其實很簡單，就是透過人與人的互動來達到『入侵』的目的！@_@！人與人的互動可以入侵你的主機？鳥哥在呼嚨你嗎？當然不是。

近日在台灣的社會你不是常看到某些人會以『退稅、中獎、花小錢買貴重物品』等名義來欺騙你的個人資料，讓你的姓名與錢包掏出口袋裡的金錢給那些可惡的金光黨嗎？社交工程也是類似的方法。在大公司裡面，或許你可能會接到這樣的電話：『我是人事部門的經理，我的帳號為何突然間不能登入了？你給我看一看，恩？乾脆直接幫我另建一個帳號，我告訴你我要的密碼是 ....』。如果你一時不查給他帳號密碼的話，你的主機可能就這樣被綁走了～

社交工程的欺騙方法多的是，包括使用『好心的 email 通知』、『警告信函』、『中獎單』等等，在在都是要欺騙你的帳號密碼，有的則利用釣魚方式來欺騙你在某些惡意網站上面輸入你的帳號密碼，很討厭的啦！舉例來說，我們昆山計中的 email 常常會收到系統維護的信件，要我們將帳號密碼提交給系統管理員統一控管，這當然是假的！計中根本不會寄出這樣的信件啊！傷腦筋啦！所以要注意啊！那要如何防範呢？

- **追蹤對談者**：不要一味的相信對方，你心須要有信心的向上司報告，不要一時心慌就中了計！
- **不要隨意透露帳號/密碼等資訊**：最好不要隨意在 Internet 上面填寫這些資料，真的很危險的！因為在 Internet 上面，你永遠不知道對方螢幕前面坐著的是誰？

◆ **利用程式功能的『被動』攻擊**

啥？除了主動攻擊之外，還有所謂的被動攻擊喔？沒錯啊，『系統入』！那如何作被動攻擊呢？那就得要由『惡意網站』講起了。如果你喜歡上網隨意瀏覽的話，那麼有的時候可能會連上一些就告很多，或者是一些釣出式視窗的網站，這些網站有時還會很好心的『提供你很多好用的軟體自動下載與安裝』的功能，如果該網站是你所信任的，例如 Red Hat, CentOS, Windows 官網的話，那還好，如果是一個你也不清楚他是幹嘛的網站，那你是否要隨意下載安裝該軟體？

如果你常常在注意一些網路主機處理的相關新聞時，常會發現 Windows 的瀏覽器 (IE) 有問題，有時則是全部的瀏覽器 (Firefox, Netscap, IE...) 都會出現問題。那你會不會覺得奇怪啊，怎麼『瀏覽器也會有問題？』這是因為很多瀏覽器會主動的答應對方 WWW 主機

所提供的各項程式功能，或者是自動安裝來自對方主機的軟體，有時瀏覽器還可能由於程式發生安全問題，讓對方 WWW 瀏覽器得以傳送惡意程式碼給你的主機來執行，嘿嘿！中標！

那你大會想啊，那我幹嘛瀏覽那樣的惡意網站？因因！總是會有些粗心大意的時候啊！如果你今天不小心收到一個 email，裡面告訴你你的銀行帳號有問題，希望你趕緊連上某個網頁去看看你的帳號是否在有問題的行列中，你會不會去？如果今天有個網路消息，說某某網頁在提供大特價商品，那你會不會去碰碰運氣？都是可能的啊！不過，這也就很容易被對方攻擊到了。

那如何防備啊？當然建立良好的習慣最重要了：

- **隨時更新主機上的所有軟體**：如果你的瀏覽器是沒有問題的，那對方傳送惡意程式碼時，你的瀏覽器就不會執行，那自然安全的多啊！
- **較小化軟體的功能**：舉例來說，讓你的收信軟體不要主動的下載檔案，讓你的瀏覽器在安裝某些軟體時，要通過你的確認後才安裝，這樣就比較容易免除一些小麻煩；
- **不要連接到不明的主機**：到這爲止可說爲這個下最難！因爲很多時候我們都用 google 在搜尋問題的解決之道啊，那你如何知道對方是不是駭人的？所以，前面所點防備還是很重要的！不要以爲沒有連接上惡意網站就不會有問題啊！

◆ **蠕蟲或木馬的 rootkit**

rootkit 意思是說可以取得 root 權限的一群工具組 (kit)，就如同前面主動攻擊程式漏洞的方法一樣，rootkit 主要也是透過主機的程式漏洞。不過，rootkit 也會透過社交工程讓使用者下載、安裝 rootkit 軟體，結果讓 cracker 得以鳩巢的綁架對方主機啊！

rootkit 除了可以透過上述的方法來進行入侵之外，rootkit 還會偽裝或者是進行自我複製，舉例來說，很多的 rootkit 本身就是蠕蟲或者是木馬間諜程式。蠕蟲會讓你的主機一直發送封包向外攻擊，結果會讓你的網路頻寬被吃光光，例如 2001-2003 年間的 Nimda, Code Red 等等；至於木馬程式 (Trojan Horse) 則會對你的主機進行開啟後門 (開一個 port 來讓 cracker 主動的入侵)，結果就是 ....綁架、綁架、綁架！

rootkit 到這挺不好追蹤的，因爲很多時候他會主動的去修改系統觀察的指令，包括 ls, top, netstat, ps, who, w, last, find 等等，讓你看不到某些有問題的程式，如此一來，你的 Linux 主機就很容易被當成是跳板了！有夠危險！那如何防備呢？

- **不要隨意安裝不明來源的檔案或者是不明網站的檔案資料；**
- **不要讓系統有太多危險的指令**：例如 SUID/SGID 的程式，這些程式很可能會造成使用者不當的使用，而使得木馬程式有機可趁！
- **可以定時以 rkhunter 之類的軟體來追查**：有個網站提供 rootkit 程式的檢查，你可以前往下載與分析你的主機：http://www.rootkit.nl/projects/rootkit_hunter.html

- **DDoS 攻擊法 (Distributed Denial of Service)**

這類型的攻擊中文翻譯成『分散式阻斷服務攻擊』，從字面上的意義來看，它就是透過分散在各地的區域電腦進行攻擊，讓你的系統所提供的服務被阻斷而無法順利的提供服務給其他用戶的方式。這種攻擊法也很多命，而且方法有很多，最常見的就屬 SYN Flood 攻擊法了！還記得我們在網路封包裡面提到的，當主機接收了一個帶有 SYN 的 TCP 封包之後，就會啟用對方要求的 port 來等待連線，並且發送出回應封包 (帶有 SYN/ACK 旗標的 TCP 封包)，並等待 Client 端的再次回應。

好了，在這個步驟當中我們來想一想，如果 cient 端在發送出 SYN 的封包後，卻將來自 Server 端的確認封包丟棄，那麼你的 Server 端就會一直空等，而且 Client 端可以透過軟體功能，在短短的時間內持續發送出這樣的 SYN 封包，那麼你的 Server 就會持續不斷的發送確認封包，並且開啟大量的 port 在空等～呵呵！等到全部主機的 port 都啟用完畢，那麼.....系統就掛了！

更可怕的是，通常攻擊主機的一方不會只有一部！他會透過 Internet 上面的區域網路 (已經成為跳板，但網站主卻沒有發現的主機) 發動全體攻擊，讓你的主機在短時間內就立刻掛點。這種 DDoS 的攻擊手法比較類似『玉石俱焚』的手段，**他不是入侵你的系統，而是要讓你的系統無法正常提供服務！**最常被用來作為阻斷式服務的網路服務就是 WWW 了，因為 WWW 通常得對整個 Internet 開放服務。

這種攻擊方法也是最難處理的，因為要嘛就得要系統核心有支援自動抵擋 DDoS 攻擊的機制，要嘛你就得要自行撰寫偵測軟體來判斷！真是麻煩啊～而除非你的網站非常大，並且『得罪不少人』，否則應該不會被 DDoS 攻擊啦！^_^

- **其他**

上面提到的都是比較常見的攻擊方法，是還有一些高等的攻擊法啦，不過那些攻擊法都需要有比較高的技術水準，例如 IP 欺騙。他可以欺騙你主機去知該封包來源是來自信任網域，而且透過封包傳送的機制，自攻擊的一方持續的主動發送出確認封包與工作指令。如此一來，你的主機可能就會誤判該封包確實有回應，而且是來自內部的主機。

不過我們知道網際網路是有路由的，而每部主機在每一個時段的 ACK 確認碼都不相同，所以這個方式要達成可以說 V，會比較麻煩，所以說，不太容易發生在我們這些小型主機上面啦！不過你還是得要注意一下說：

- **設定規則完善的防火牆**：利用 Linux 內建的防火牆軟體 iptables 建立較為完善的防火牆，可以防範部分的攻擊行為；
- **核心功能**：這部份比較複雜，你必須要對系統核心有很深入的瞭解，才有辦法設定好你的核心網路功能。

- **登錄檔與系統監控**：你可以透過分析登錄檔來瞭解系統的狀況，另外也可以透過類似 MRTG 之類的監控軟體來即時瞭解到系統是否有異常，這些工作都是很好的努力方向！

◆ **小結語**

要讓你的系統更安全，沒有『三秒三』是沒辦法『上梁山』的！我們也一直鼓吹，『維護網站比架設網站還要重要』的觀念！因為『一人得道雞犬升天』，同樣的道理：『一人中標全員掛點』，不要以為你的主機沒有啥重要資料，被入侵或被植入木馬也沒有關係，因為我們的伺服器通常會對內部來源的主機規範的較為寬鬆，如果你的主機在公司內部，但是不小心被入侵的話，那麼位於公司的伺服器是否就會暴露在危險的環境當中了？

另外，在蠕蟲很『發達』的今代，我們也會發現只要區域網路裡面有一部主機中標，整個區域網路就會無法使用網路了，因為頻寬已經被蠕蟲塞爆！如果老闆發現他今天沒有辦法收信了，但無法收信的原因並非伺服器掛點，而是因為內部人員的某部個人電腦中了蠕蟲，而那部主機中蠕蟲的原因只是因為該使用者不小心去看了一下色情網站，你覺得老闆會高興的跟該員工一起看色情網站還是 fire 掉該人員？

所以啊，主機防護還是很重要的！不要小看了！提供幾個方向給大家思考看看吧：

1. 建立完善的登入密碼規則限制；
2. 完善的主機權限設定；
3. 設定自動升級與修補軟體漏洞、及移除危險軟體；
4. 在每項系統服務的設定當中，強化安全設定的項目；
5. 利用 iptables, TCP_Wrappers 強化網路防火牆；
6. 利用主機監控軟體如 MRTG 與 logwatch 來分析主機狀況與登錄檔；

7.1.3 主機能作的保護：軟體更新、減少網路服務、啟動 SELinux

根據本章前面的分析，現在你知道封包的流向以及主機該有需要進行的防護了。不過你或許還是有疑慮，那就是，既然我都已經有了防火牆，那麼權限的控管啦、密碼的嚴密性啦、伺服器軟體的更新啦、SELinux 啦等等的，是否就沒有這麼重要呢？舉例它是封包進入的第一關卡！這關把關嚴格，後續可以稍微寬鬆嗎？其實...你錯了！對於開放某些服務的伺服器來說，你的防火牆『根本跟此一樣，是沒有用的！』怎麼說呢？

◆ **軟體更新的重要性**

讓我們瞧一瞧圖 7.1-1 的流程好了，假設你需要對全世界開放 WWW，那麼提供 WWW 服務的 httpd 這支程式就得要執行，並且，你的防火牆得要打開 port 80 讓全世界都可以連接到你的 port 80，這樣才是一部合理的 WWW 伺服器嘛！問題來啦，如果 httpd 這支程

式有資安方面的問題時，請問防火牆有沒有效用？當然沒有！因為防火牆原本就得要開放 port 80 啊！此時防火牆對你的 WWW 一點防護也沒有。那怎麼辦？

沒啥好說的，就是軟體持續更新到最新就對了！因為自由軟體就是有這個好處，當你的程式有問題時，開發者會在最短的時間內取得志工提供的修補程式 (patch)，並將該程式碼補充到軟體更新資料中，讓一般用戶可以直接透過網路來自動更新。因此，要克服這個伺服器軟體的問題，更新系統軟體就對了。

但是你得要注意，你的系統能否更新軟體與系統的版本有關！舉例來說，2003 年方右發佈的 Red Hat 9 目前已經沒有支援了，如果你還是執意要安裝 Red Hat 9 這款系統，那麼很抱歉，你得要手動將系統內的軟體透過 make 動作來重新編譯到最新版，因此，很麻煩～同樣的，Fedora 最新版雖然有提供網路自動更新，但是 Fedora 每一個版本的維護期間較短，你可能需要常常大幅度的變更你的版本，這對伺服器的設定也不妥當。此時一個企業版本的 Linux distributions 就很重要啦！舉例來說，鳥站的主機截至目前為止 (2011/07) 還是使用 CentOS 4.x，因為這個版本目前還是持續維護中。這對伺服器來說，是相當重要的！穩定與安全比什麼都重要！

想要瞭解軟體的安全通報，可以參考如下的網站資料喔！

- 台灣電腦網路危機處理小組(TWCERT)：http://www.cert.org.tw/
- Red Hat 的官方說明：https://www.redhat.com/support/

◆ 認識系統服務的重要性

再回到圖 7.1-1 當中，用時間來看一下第二章網路基礎裡面談到的網路連線是雙向這件事，我們會得到一個答案，那就是在圖 7.1-1 內的第二個步驟中，如果能夠減少伺服器上面的監聽埠口，此時因為伺服器端沒有可供連線的埠口，用戶端當然也就無法連線到伺服器端啦！那麼如何限制伺服器開啟的埠口呢？第二章就談到過了，關閉埠口的方式是透過關閉網路服務。沒錯啊！所以囉，此時能夠減少網路服務就減少，可以避免很多不必要的麻煩。

◆ 權限與 SELinux 的輔助

根據網路上面多年來的觀察，很多朋友在發生權限不足方面的問題後，都會直接將某個目錄直接修訂成為 chmod -R 777 /some/path/。如果這部主機只是測試用的沒有上網提供服務，那還好。如果有上網提供某些服務時，那可就傷腦筋了！因為目錄的 wx 權限設定一起後，代表該身份可以進行新增與刪除的動作。偶偶你入給 777 (rwxrwxrwx)，代表所有的人都可以在該目錄下進行新增與刪除！萬一不小心某支程式被攻擊而被取得操作權，想想看，你的系統不就可能被寫入某些可怕的東西了嗎？所以不要隨便設定權限啊！

那麼如果由於當初規劃的帳號身份與群組設定的太雜亂，導致無法使用單純的三種身份的三種權限來設定你的系統時，那該如何是好？沒關係的，可以透過 ACL 這個好用的東西！ACL 可以針對單一帳號或單一群組進行特定的權限設定，相當好用喔！他可以輔助傳統 Unix 的權限設定方面的不便哩。詳情請參考基礎篇的內容囉！

那麼如何避免使用者亂用系統，亂設定權限呢？這個時候就得要透過 SELinux 來控制了。SELinux 可以在程序與檔案之間再加入一道細部的權限控制，因此，即使程序與檔案的權限符合了操作動作，但如果程序與檔案的 SELinux 類型 (type) 不吻合時，那麼該程序就無法讀取該檔案喔！此外，我們的 CentOS 也針對了某些常用的網路服務制訂了許多的檔案使用規則 (rule)，如果這些規則沒有啟用，那麼即使權限、SELinux 類型都對了，該網路服務的功能還是無法順利的運作喔！

根據這樣的分析，我們可以知道，**隨時更新系統軟體、限制連線埠口以及透過啟動 SELinux 來限制網路服務的權限**，經過這三個簡單的步驟，你的系統將可以獲得相當大的保護！當然啦，後續的防火牆以及系統登錄檔分析工作仍是需要進行的。不過後續將依據這三點來深入介紹。

7.2 網路自動升級軟體

在現在的網際網路上面，cracker 實在是太多了！這些個人會利用已經存在的系統漏洞，來進行偵測、入侵你的主機。因此，除了未來架設防火牆之外，**最重要的 Linux 日常管理工作，莫過於軟體的升級**了！不過，如果使用者還得要自己每天觀察網路安全通報，並主動去查詢各大 distribution 針對這些漏洞來提供升級軟體包，那真是太不人性化了！因此，目前就有很多線上直接更新的機制出現了！有了這些線上直接更新軟體的手段與方法，我們系統管理員在管理主機系統上面，可就輕鬆的多囉！

7.2.1 如何進行軟體升級

通常鳥哥安裝好 Linux 之後，會先開啟系統預設的防火牆機制，然後第一件事情就是進行全系統更新啦！不論是哪一套 Linux 鳥哥都是這樣做的，因為要避免軟體資安的問題嘛！好了，那麼 Linux 上面的軟體該如何進行更新與升級呢？還記得你是如何安裝軟體的嗎？不就是 rpm, tarball 與 dpkg 嗎？所以囉，你的軟體如果想要升級，那就得依據當時你安裝該軟體的方式來進行升級啊！而每種方式都有其通用性：

- RPM：

  這是目前最常見於 Linux distribution 當中的軟體管理方式，包括 CentOS / Fedora / SuSE / Red Hat / Mandriva 等等，都是使用這個方式來管理的；

- **Tarball：**

  利用軟體的官方網站所釋出的原始碼在您的系統上面編譯與安裝，一般來說，由於軟體是直接在自己的機器上面編譯的，所以效能會比較好一些。不過，升級的時候就比較麻煩，因為人得要下載新的原始碼並且重新編譯一次。這種安裝模式常見於某些特殊軟體 (沒有包含在 distribution 當中)，或者是 Gentoo 這個強調效能的 distribution；

- **dpkg：**

  是 debian 這個 distribution 所使用的軟體管理方式，與 RPM 很類似，都是透過預先編譯的處理，可以讓 end user 直接使用來升級與安裝。

舉例來說，如果你的系統是 CentOS，我們知道他使用的是 RPM 類型的軟體管理模式，那如果你想要安裝 B2D 的軟體怎辦？要注意，B2D 是使用 debian 的 dpkg 來管理軟體的，兩者並不相同啊！要互相安裝大難了！所以說，要升級的話，得先瞭解到你系統上的軟體安裝與管理的方法才行。

不過，有個特殊案例，那就是舊版本的 Linux (例如 Red Hat 9) 的軟體升級該如何是好？由於舊版本的軟體支援要本來就比較差，原廠公司或者是社群也沒有這麼多心力放在舊版本的支援上，所以，你這個時候可以選擇：**(1)升級到較新的版本，例如 CentOS 6.x，或者是 (2)利用 Tarball 來自行升級核心與軟體。**不過，比較建議升級到新版本啦，因為要自行以手動方式由 Tarball 安裝到最新的版本，實在是很費時費力，而且還得要常常查閱官方網站所拋出的最新消息，漏過一則都可能發生無法預期的狀況。

我們都曉得在 Windows 的環境下，他有提供一個 Live update 的項目可以自動的線上升級，甚至很多的防毒軟體與防木馬軟體也都有拋出即時的線上更新，如此一來可以讓您的軟體維持在最新版的狀況，真是好啊！咦！那我們的 Linux 是否有這樣的功能？如果有的話，那麼系統自動進行軟體升級，不就可以輕鬆又快樂了？沒錯！確實是這樣的！所以就讓我們來談一談 Linux 的線上升級機制吧！

在 Linux 最常見的軟體安裝方式：RPM / Tarball / dpkg 當中，Tarball 由於取得的是原始碼，所以要用 Tarball 來作線上自動更新是不太可能進行的，所以僅能用 RPM 或 dpkg 這兩種軟體管理的方式來進行線上更新了。

但 RPM 與 dpkg 不是有所謂的相依屬性嗎？這倒不需要擔心啦！因為我們的 RPM 與 dpkg 軟體檔案都有一些軟體的標頭資訊，並且時記錄了軟體的相依屬性 (記得使用 rpm -q 的查詢嗎)，所以當分析這些標頭資訊並使用一些機制將這些相依資訊記錄下來後，再透過一些額外的網路功能，就能夠自動的分析你的系統與修補軟體之間的差異，並可進一步幫你分析所需要升級與相依屬性的軟體，就可達成自動升級的理想啦！

由於各家 distributions 在管理系統上都有自己獨特的想法，所以在分析 RPM 或 dpkg 軟體與方式上面就有所不同，也就有了這些不同的線上升級機制啦：

- yum：

  CentOS 與 Fedora 所常用的自動升級機制，透過 FTP 或 WWW 來進行線上升級以及線上直接安裝軟體；

- apt：

  最早由 debian 這個 distribution 所發展，現在 B2D 也是使用 apt，同樣由於 apt 的可移植性，所以只要你的 RPM 可以使用 apt 來管理的話，就可以自行建立 apt 伺服器來提供其他使用者進行線上安裝與升級。

- you：

  所謂的 Yast Online Update (YOU) 是由 SuSE 所自行開發出來的線上安裝升級方式，經過註冊取得一組帳號密碼後，就能夠使用 you 的機制來進行線上升級。不過如果是免費的版本，則僅有 60 天的試用期！

- urpmi：

  這個則是 Mandriva 所提供的線上升級機制！

講了這些升級機制並且與 distribution 作了對應，你就該瞭解到：『**每個 distribution 可以使用的線上升級機制都不相同**』的啊！所以請參考你的 distribution 所提供的文件來進行線上升級的設定喔！否則就得要自行手動下載安裝了！@_@

鳥哥這裡都是使用 CentOS 這個 Red Hat 相容的 distributions 來介紹的，因此，底下僅介紹了 yum 而已。不過，yum 已經能夠適用於 CentOS, Red Hat Enterprise Linux, Fedora 等等，也很該是挺夠用的了！另外，基礎篇裡面已經談過 rpm 與 yum 的用法，所以在這裡僅是加強介紹與更新有關的用法而已喔！

7.2.2 CentOS 的 yum 軟體更新、映射站使用的原理

我們曾經在基礎篇裡面談過 yum 了，基本上他的原理是，我們的 CentOS 會跑到 yum 伺服器上頭，下載了官方網站釋出的 RPM 表頭清單資料，該資料除了記載每個 RPM 軟體的相依性之外，也說明了 RPM 檔案所放置的容器 (repository) 所在。因此透過分析這些資料，我們的 CentOS 就能夠直接使用 yum 去下載與安裝所需要的軟體了！詳細圖示與流程有點像這樣：

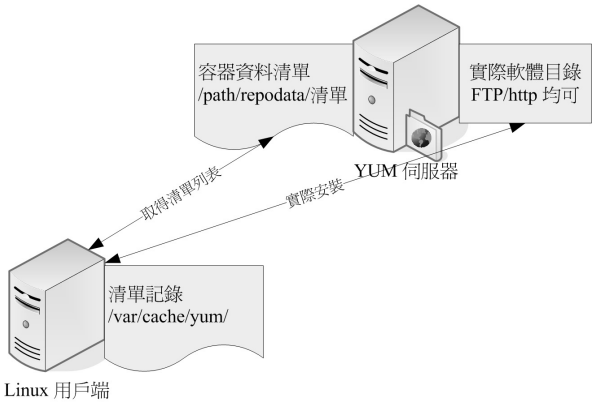

圖 7.2-1　使用 yum 下載清單表頭與取得容器相關資料示意圖

1. 先由設定檔判斷 yum server 所在 IP 位址；
2. 連接到 yum server 後，先下載新的 RPM 檔案的表頭資料；
3. 分析比較使用者所欲安裝/升級的檔案，並提供使用者確認；
4. 下載使用者選擇的檔案到系統中的 /var/cache/yum ，並進行實際安裝；

由於你所下載的清單當中已經含有所有官方網站所釋出的 RPM 檔案的表頭相依屬性的關係，所以如果你想要安裝的軟體包含某些尚未安裝的相依軟體時，我們的 yum 會順便幫你下載所需要的其他軟體，預先安裝後，再安裝你所實際需要的軟體！從分析、下載到安裝，全部一口氣搞定！很簡單的啦！

不過，恐怕還是有問題。如果全世界使用 CentOS 的朋友通通連線到同一部 Yum 伺服器去下載所需要的 RPM 檔案，哇！那頻寬不就很容易被塞爆！那怎麼辦？沒關係，有所謂的映射站啊！CentOS 在世界各地都有映射站，這些映射站會將官網的 yum 伺服器的資料複製一份，同時在映射站上面也提供同樣的 yum 功能，因此，你可以在任何一部 yum 伺服器的映射站上面下載與安裝軟體。舉例來說 CentOS 官網上面列出的亞洲地區映射站一覽表：

- http://www.centos.org/modules/tinycontent/index.php?id=32

現在的 yum 大很聰明，它會自動的去分析離你的主機最近的那部映射站，然後直接使用該部映射主機作為你的 yum 來源，因此，『理論上』你下載要更動任何設定，在台灣，你的 CentOS 就會使用台灣地區的 yum 伺服器囉！就這麼簡單！所以，接下來就讓我們直接來談談怎麼使用 yum 吧！

yum 的原理與相關使用，我們在基礎篇裡面已經分門別類的介紹過了，因此底下僅就比較重要的部分介紹一下囉！

7.2.3 yum 的使用：安裝, 軟體群組, 全系統更新

yum 可不止能夠線上自動升級而已，他還可以作查詢、軟體群組的安裝、整體版本的升級等等，好用的哩！先來談論一下 yum 這個指令的用法吧：

```
[root@www ~]# yum [option] [查詢的工作項目] [相關參數]
選項與參數：
option：主要的參數，包括有：
   -y ：當 yum 詢問使用者的意見時，主動回答 yes 而不需要由鍵盤輸入；

[查詢的工作項目]：由於不同的使用條件，而有一些選擇的項目，包括：
   install ：指定安裝的軟體名稱，所以後面需接『 軟體名稱 』
   update  ：進行整體升級的行爲；當然也可以接某個軟體，僅升級一個軟體；
   remove  ：移除某個軟體，後面需接軟體名稱；
   search  ：搜尋某個軟體或者是重要關鍵字；
   list    ：列出目前 yum 所管理的所有的軟體名稱與版本，有點類似 rpm -qa；
   info    ：同上，不過有點類似 rpm -qai 的執行結果；
   clean   ：下載的檔案被放到 /var/cache/yum ，可使用 clean 將他移除，
             可清除的項目：packages | headers | metadata | cache 等；

在[查詢的工作項目]部分還可以具有整個群組軟體的安裝方式，如下所示：
   grouplist    ：列出所有可使用的『軟體群組』，例如 Development Tools 之類的；
   groupinfo    ：後面接 group_name，則可瞭解該 group 內含的所有軟體名；
   groupinstall：這個好用！可以安裝一整組的軟體群組，相當的不錯用！
                 更常與 --installroot=/some/path 共用來安裝新系統
   groupremove ：移除某個軟體群組；

# 範例一：搜尋 CentOS 官網提供的軟體名稱是否有與 RAID 有關的？
[root@www ~]# yum search raid
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile  <==這裡就是在測試最快的映射站
 * base: ftp.isu.edu.tw                     <==共有四個容器內容
 * extras: ftp.isu.edu.tw                   <==每個容器都在 ftp.isu.edu.tw 上
 * updates: ftp.isu.edu.tw
base                          | 3.7 kB     00:00  <==下載軟體的表頭清單中
extras                        |  951 B     00:00
```

```
updates                          | 3.5 kB     00:00
=================== Matched: raid =================<==找到的結果如下
dmraid.i686 : dmraid (Device-mapper RAID tool and library)
....(中間省略)....
mdadm.x86_64 : The mdadm program controls Linux md devices (software RAID
....(底下省略)....

# 範例二：上述輸出結果中，mdadm 的功能爲何？
[root@www ~]# yum info mdadm
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: ftp.twaren.net
 * extras: ftp.twaren.net
 * updates: ftp.twaren.net
Installed Packages  <==這裡說明這是已經安裝的軟體！
Name        : mdadm
Arch        : x86_64
Version     : 3.1.3
Release     : 1.el6
Size        : 667 k
Repo        : installed
From repo   : anaconda-CentOS-201106060106.x86_64
Summary     : The mdadm program controls Linux md devices (software RAID
URL         : http://www.kernel.org/pub/linux/utils/raid/mdadm/
License     : GPLv2+
Description: The mdadm program is used to create, manage, and monitor
....(底下省略)....
# 由上述底線的 Summary 關鍵字，知道這軟體在達成軟體磁碟陣列功能！！
```

yum 真是個很好用的東西，它可以直接查詢是否有某些特殊的軟體名稱。舉例來說，你可以利用底下的兩個方式取得軟體名稱：

- yum search **"一些關鍵字"**
- yum list **(可列出所有的軟體檔名)**

然後再以規表示法取得關鍵字，或者是『yum info **"軟體名稱"**』就能夠知道該軟體的用途，最後再決定要不要安裝啊！上面的範例一就是在找出磁碟陣列的管理軟體。如果確定要安裝時，那就參考參考底下的流程吧！

◆ **利用 yum 進行安裝**

```
# 範例三：安裝某個軟體吧！以 mdadm 這個軟體名爲例：
[root@www ~]# yum install mdadm
....(前面省略)....
Setting up Install Process
Package mdadm-3.1.3-1.el6.x86_64 already installed and latest version
Nothing to do

[root@www ~]# yum install mdadma
Setting up Install Process
No package mdadma available.
Nothing to do
```

仔細的看上述的兩個指令，第二個指令爲了故意寫錯字，讓軟體名稱由 mdadm 變成 mdadma 了！模擬同學如果打錯字時所輸出的訊息。由上述的訊息，你可以知道，同樣結果是『Nothing to do』，但是 yum 會告訴你該軟體是『已安裝 (installed and lastest version)』還是『沒有該軟體 (No package mdadma avaliable)』。作這個範例是希望朋友們能夠仔細的看輸出的訊息啦！好啦！我們還是來安裝一個不曾安裝過的，就拿 javacc 這套軟體來裝看看好了！

```
[root@www ~]# yum list javacc*
Available Packages
javacc.x86_64            4.1-0.5.el6      base
javacc-demo.x86_64       4.1-0.5.el6      base
javacc-manual.x86_64     4.1-0.5.el6      base
# 共有三套軟體，分別是 javacc, javacc-demo, javacc-manual ，版本爲 4.1-0.5.el6，
# 軟體是放置到名稱爲 base 的容器當中存放的。

[root@www ~]# yum install javacc
....(前面省略)....
Setting up Install Process
Resolving Dependencies
--> Running transaction check  <==開始檢查有沒有相依屬性的軟體問題
---> Package javacc.x86_64 0:4.1-0.5.el6 set to be updated
....(中間省略)....

================================================================================
 Package                    Arch     Version           Repository  Size
================================================================================
Installing:
 javacc                     x86_64   4.1-0.5.el6       base       895 k
```

```
Installing for dependencies:
 java-1.5.0-gcj              x86_64   1.5.0.0-29.1.el6  base       139 k
 java_cup                    x86_64   1:0.10k-5.el6     base       197 k
 sinjdoc                     x86_64   0.5-9.1.el6       base       705 k

Transaction Summary
================================================================================
Install       4 Package(s)  <==安裝軟體彙整，共安裝 4 個，升級 0 個軟體
Upgrade       0 Package(s)

Total download size: 1.9 M
Installed size: 5.6 M
Is this ok [y/N]: y  <==讓你確認要下載否！
Downloading Packages:
(1/4): java-1.5.0-gcj-1.5.0.0-29.1.el6.x86_64.rpm      | 139 kB     00:00
(2/4): java_cup-0.10k-5.el6.x86_64.rpm                 | 197 kB     00:00
(3/4): javacc-4.1-0.5.el6.x86_64.rpm                   | 895 kB     00:00
(4/4): sinjdoc-0.5-9.1.el6.x86_64.rpm                  | 705 kB     00:00
--------------------------------------------------------------------------------
Total                                          3.1 MB/s | 1.9 MB     00:00
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing     : java-1.5.0-gcj-1.5.0.0-29.1.el6.x86_64             1/4
  Installing     : 1:java_cup-0.10k-5.el6.x86_64                      2/4
  Installing     : sinjdoc-0.5-9.1.el6.x86_64                         3/4
  Installing     : javacc-4.1-0.5.el6.x86_64                          4/4

Installed:            <==主要需要安裝的
  javacc.x86_64 0:4.1-0.5.el6

Dependency Installed: <==爲解決相依性額外裝的
  java-1.5.0-gcj.x86_64 0:1.5.0.0-29.1.el6   java_cup.x86_64 1:0.10k-5.el6
  sinjdoc.x86_64 0:0.5-9.1.el6

Complete!
```

瞧！經過 yum 我們可以很輕鬆的就安裝好一個軟體，並且這個軟體已經自動的幫我們做好相依屬性的克服了，真是方便到爆！另外，CentOS 6.x 預設的情況下，yum 下載的套件除了每個容器的表頭清單檔案之外，**所有下載的 RPM 檔案都會在安裝完畢之後予以**

刪除！這樣你的系統就不會有容量被下載的資料塞爆的問題。但如果你想要下載的 RPM 檔案繼續保留在 /var/cache/yum 當中，就得要修改 /etc/yum.conf 設定檔了！

```
[root@www ~]# vim /etc/yum.conf    <==看看就好，不要真的作！
[main]
cachedir=/var/cache/yum/$basearch/$releasever
keepcache=1
debuglevel=2
logfile=/var/log/yum.log
exactarch=1
obsoletes=1
....(底下省略)....
```

上述的特殊字體地方將 0 改成 1，這樣就能夠讓你的 RPM 檔案保存下來。不過，除非你有好多部主機要更新，你想利用一台先 yum 升級且下載，然後將所有的 RPM 檔案收集起來給內網的機器升級 (rpm -Fvh *.rpm) 之外，**上面的 vim 修改動作不建議修改**！因為你的 /var 恐怕會被塞爆啊！再次提醒！

- **yum 安裝軟體群組**

什麼是『軟體群組』呢？由於 RPM 軟體將一個大型程式分成好幾個小計畫來執行，每個小計畫都可以獨立安裝，這樣的好處是可以讓使用者/與軟體發展者/安裝不同的環境！舉例來說，在桌面系統中 (Desktop)，一般用戶應該不會跑去發展軟體吧？所以針對桌上型電腦，軟體群組又分為 "Desktop Platform" 與開發者 "Desktop Platform Development" 兩部份，每個軟體群組內又含有多個不同的RPM 軟體檔案！這樣做的用途是方便使用者/安裝一整套的程式啦！

那麼系統有多少軟體群組呢？又該如何觀察某個軟體群組有哪有的 RPM 檔案呢？我們就利用 Desktop Platform 這個程式來說明一下囉：

```
# 範例四：查詢系統有的軟體群組有多少個？
[root@www ~]# LANG=C yum grouplist
Installed Groups:              <==這個是已安裝的軟體群組
   Additional Development
   Arabic Support
   Armenian Support
   Base
....(中間省略)....
Available Groups:              <==這個是尚可安裝的軟體群組
   Afrikaans Support
   Albanian Support
   Amazigh Support
```

```
....(中間省略)....
   Desktop Platform
   Desktop Platform Development
....(後面省略)....

# 範例五：那麼 Desktop Platform 內含多少個 RPM 軟體呢？
[root@www ~]# yum groupinfo "Desktop Platform"
Group: 桌面環境平台
 Description: 受支援的 CentOS Linux 桌面平台函式庫。
 Mandatory Packages: <==主要的會被安裝的軟體有這些
   atk
....(中間省略)....
 Optional Packages:  <==額外可選擇的軟體是這些
   qt-mysql
....(底下省略)....
# 如果你確定要安裝這個軟體群組的話，那就這樣做：

[root@www ~]# yum groupinstall "Desktop Platform"
# 因爲這裡在介紹伺服器的環境，所以上面的動作鳥哥是按下 n 來拒絕安裝的！
```

利用這個『yum groupinstall "軟體群組名"』可以讓你一口氣安裝很多的軟體，而不必擔心某個軟體忘記裝了！實在是很不錯啦～而且利用 groupinfo 的功能你也可以發現一些不錯的軟體資料，如此一來，你就可以很方便的管理你的 Linux 系統了，很不錯吧！

- **全系統更新**

我們都知道使用『yum update』就可以進行軟體的更新。不過你曉得嗎？yum update 也可以直接進行同一版本的升級喔！舉例來說，你可以從 6.0 升級到 6.1 版本哩！而且中間過程完全無痛的！就跟一般軟體升級而已，並沒有不同的！夠愉快吧！

不過，如果你是想要從較舊版的 CentOS 5.x 升級到 6.x 的話，那麼可能就得要多費些功夫了。爲啥不要重灌比較快呢？因爲你可能已經有些資料設定好，所以不想變更嘛！但引導說，不同版本 (ex> 5.x --> 6.x) 間的升級最好還是不要嘗試啦！重新安裝可能是最好的狀況。底下列出酷學園的前輩提供的升級方式，以及 CentOS 官網直接提供的升級方式給你參考參考：

- 酷學園 TWU2 所提供的 Red Hat 9 升級到 CentOS 3.x 的方法：
  http://phorum.study-area.org/index.php/topic,28648.html
- CentOS 官網提供的 CentOS 4.x 升級到 5.x 的方法：
  http://lists.centos.org/pipermail/centos-announce/2007-April/013660.html

- CentOS 維護團隊提供的 CentOS 4.4 升級到 5.1 的方法：
  http://wiki.centos.org/HowTos/MigrationGuide/ServerCD_4.4_to_5

例題

請設定一個工作排程，讓你的 CentOS 可以每天自動更新系統

答：可以使用『crontab -e』來動作，也可以編輯『vim /etc/crontab』來動作，由於這個更新是系統方面的，所以鳥哥習慣使用 vim /etc/crontab 來進行指令的說明。其實內容很簡單：

```
40 5 * * * root yum -y update && yum clean packages
```

這樣就可以自動更新了，時間訂在每天的凌晨 5:40。

7.2.4 挑選特定的映射站：修改 yum 設定檔與清除 yum 快取

雖然 yum 是你的主機能夠連線上 Internet 就可以直接使用的，不過，由於 CentOS 的映射站台可能會選錯，舉例來說，我們在台灣，但是 CentOS 的映射站台卻選擇到了大陸北京或者是日本去，有沒有可能發生啊！有啊！鳥哥教學方面就常常發生這樣的問題，要知道，我們連線到大陸或日本的速度是非常慢的呢！那怎辦？當然就是手動的修改一下 yum 的設定檔就好囉！

在台灣，鳥哥熟悉的 CentOS 映射站台主要有崑山科大、高速網路中心與義守大學。在學術網路之外，鳥哥近來比較偏好高速網路中心，似乎更新的速度比較快，而且連接台灣學術網路也非常快速哩！因此，鳥哥建議台灣的朋友使用高速網路中心的 ftp 主機資源來作為 yum 伺服器來源喔！不過，因為鳥哥的機器很多都在崑山科大，所以在學術網路上，使用的不而是崑山科大的 FTP 囉。目前高速網路中心對於 CentOS 所提供的相關網址如下：

- http://ftp.twaren.net/Linux/CentOS/6/

如果你連接到上述的網址後，就會發現裡面有一堆連結，那些連結就是這個 yum 伺服器所提供的容器了！所以高速網路中心也提供了 addons, centosplus, extras, fasttrack, os, updates 等容器，最好認的容器就是 os (系統預設的軟體) 與 updates (軟體升級版本) 囉！由於鳥哥在我的測試用主機是利用 x86_64 的版本，因此那個 os 再點進去，就會得到如下的可提供安裝的網址：

- http://ftp.twaren.net/Linux/CentOS/6/os/x86_64/

為什麼在上述的網址內呢？有什麼特色！**最重要的特色就是那個『repodata』的目錄！該目錄就是分析 RPM 軟體後所產生的軟體屬性相依資料放置處！**因此，當你要找容器所在網

址時，最重要的就是該網址下一定要有個名為 repodata 的目錄存在！那就是容器的網址了！其他的容器也是網址，就請各位看倌自行尋找一下喔！現在讓我們修改設定檔吧！

```
[root@www ~]# vim /etc/yum.repos.d/CentOS-Base.repo
[base]
name=CentOS-$releasever - Base
mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=os
#baseurl=http://mirror.centos.org/centos/$releasever/os/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6
```

如上所示，鳥哥僅列出 base 這個容器的原始內容而已，其他的容器內容請自行查閱囉！上面的資料需要注意的是：

- **[base]：**

  代表容器的名字！中括號一定要存在，裡面的名稱則可以隨意取。但是不能有兩個相同的容器名稱，否則 yum 會不曉得該到哪裡去找容器相關軟體清單檔案。

- **name：**

  只是說明一下這個容器的意義而已，重要性不高！

- **mirrorlist=：**

  列出這個容器可以使用的映射站台，如果不想使用，可以註解到這行。由於等一下我們是直接設定映射站，因此這行待會兒確實是需要註解掉的喔！

- **baseurl=：**

  這個最重要，因為後面接的就是容器的實際網址！mirrorlist 是由 yum 程式自行去捉映射站台，baseurl 則是指定固定的一個容器網址！我們剛剛找到的網址放到這裡來啦！

- **enable=1：**

  就是讓這個容器被啟動。如果不想啟動可以使用 enable=0 喔！

- **gpgcheck=1：**

  還記得 RPM 的數位簽章嗎？這就是指定是否需要查閱 RPM 檔案內的數位簽章！

- **gpgkey=：**

  就是數位簽章的公鑰檔所在位置！使用預設值即可

瞭解這個設定檔之後，接下來讓我們修改整個檔案的內容，讓我們這部主機可以直接使用高速網路中心的資源吧！修改的方式為刪除列出 base 這個容器項目而已，其他的項目請您自行依照上述的作法來處理即可！

```
[root@www ~]# vim /etc/yum.repos.d/CentOS-Base.repo
[base]
name=CentOS-$releasever - Base
baseurl=http://ftp.twaren.net/Linux/CentOS/6/os/x86_64/    <==就屬它最重要！
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6
# 底下其他的容器項目，請自行到高速網路中心去查詢後自己處理！

[root@www ~]# yum clean all  <==改過設定檔，最好清除既有清單
```

接下來當然就是給他測試一下囉！如何測試呢？再次使用 yum 即可啊！

```
# 範例：列出目前 yum server 所使用的容器有哪些？
[root@www ~]# yum repolist all
repo id         repo name                 status
base            CentOS-6 - Base           enabled: 6,019
c6-media        CentOS-6 - Media          disabled
centosplus      CentOS-6 - Plus           disabled
contrib         CentOS-6 - Contrib        disabled
debug           CentOS-6 - Debuginfo      disabled
extras          CentOS-6 - Extras         enabled:     0
updates         CentOS-6 - Updates        enabled: 1,042
repolist: 7,061
# 在 status 上寫 enabled 才是有啓動的！由於 /etc/yum.repos.d/
# 有多個設定檔，所以你會發現還有其他的容器存在。
```

- **修改容器產生的問題與解決之道**

由於我們是修改系統預設的設定檔，事實上，我們應該要在 /etc/yum.repos.d/ 底下新建一個檔案，該副檔名必須是 .repo 才行！但因為我們使用的是指定特定的映射站台，而不是其他軟體開發商提供的容器，因此才修改系統預設設定檔。但是可能由於使用的容器版本有新舊之分，你得要知道，yum 會先下載容器的清單到本機的 /var/cache/yum 裡面去！那我們修改了網址卻沒有修改容器名稱 (中括號內的文字)，可能就會造成本機的清單與 yum 伺服器的清單不同步，此時就會出現無法更新的問題了！

那怎麼辦啊？很簡單，就清除掉本機上面的舊資料即可！需要手動處理嗎？不需要的，透過 yum 的 clean 項目來處理即可！