

## 高雄市政府教育局 105 年度第 2 梯次網管人員應用研習課程表

第 3 場次-(Linux 基礎班) (共 1 場次 2 天)

日期	時間	課程內容	講師
10 月 14 日 (五)	09:00-10:00	Linux 概述 & 安裝	漢民國小 張宏明
	10:00-11:00	Linux 概述 & 安裝	
	11:00-12:00	圖形介面基本操作	
	12:00-13:30	休息時間	
	13:30-14:30	Unix 使用者常用指令	
	14:30-15:30	Unix 使用者常用指令	
	15:30-16:30	系統管理常用指令集	
10 月 15 日 (六)	09:00-10:00	常用文書編輯器介紹	
	10:00-11:00	檔案權限	
	11:00-12:00	檔案存取權限	
	12:00-13:30	休息時間	
	13:30-14:30	系統管理相關檔案介紹 (passwd, service, inetd, conf)	
	14:30-15:30	系統管理相關檔案介紹 (crontab, fstab, profile)	
	15:30-16:30	系統管理實作	



# Linux 簡介

高雄市政府教育局  
資訊教育中心  
網路組 張宏明  
更新日期：2012/11/14

## 一、認識 **Linux**：

在資訊的領域中，一般人所接觸到的幾乎都 **WINDOWS 9X/2000/ME** 系統，而一般事務的處理也是使用 **WINDOWS** 的 **MS OFFICE** 或其他應用軟體，然而 **WINDOWS** 並不代表一切，**PC** 所能處理工作也不僅於此，在 **WINDOWS** 龐大身影的背後，有一群默默耕耘的程式發展人員，在自由軟體的精神下，集大眾之力，致力發展比視窗系統更穩定、安全的作業系統。

原本只在大型電腦系統上執行 **UNIX** 作業系統，其多工、穩定、快速的處理能力，被移植到 **PC** 上，即成為我們所熟知的 **UNIX LIKE** 作業系統，例如 **LINUX**、**FreeBSD**、**NetBSD** 等，這些作業系統不僅保有 **UNIX** 的強大功能，而且是免費的作業系統，遵循 **Open Source** 的精神，完全開放程式碼供大家修改散佈。而在其作業系統上所執行的應用軟體大都是免費使用。

近來，**linux** 對於包括中文在內的各國語言支援日益完善，不僅使用者大增（包括海峽對岸也大力推 **linux** 系統），連 **IBM**、**Oracle**、**HP** 等軟硬體大廠，也紛紛投入此領域，**Linux** 的發展將有非常大的寬廣空間。

## 二、**LINUX** 起源：

**Linux** 創始人為芬蘭籍 **Linus Torvalds**，在 1991 年 10 月 5 日於 News 初次正式發表，版本 0.02 的「第一版」**Linux** 系統，其目的為讓個人電腦也能享用同樣功能而且免費的 **UNIX** 系統。

當時 **Linus** 將這系統放在芬蘭最熱門的 FTP 站，並且建造一個名為「**Linux**」的目錄，意謂「**Linus** 的 **Unix** 系統」，正式將 **LINUX** 定名。後來大家從網路上流傳開來，許多程式設計師紛紛加入發展的計劃，經過不斷的修正與改進，「**Linux**」便成為炙手可熱而功能強大的作業系統。

1994 正式釋出 **Linux Kernel 1.0**

目前的版本為常用 2.4.x 以及最新的 2.6.x。一般人所認定的 **LINUX** 版本，事實上所指的是 **Linux Kernel**（核心程式）的版本，所謂的 **Kernel** 是作業系統的核心，在電腦啟動後，所最先載入的程式，其作用為負責硬體與軟體的溝通、控制以及資源分配。**kernel** 版本的編號，通常以小數點第一位為偶數表示是 **stable**（穩定）的版本，以奇數表示為測試版本，例如 2.4.x，2.6.x 為穩定版本，2.6.x 為測試版本。

### 三、**LINUX** 套件(Distribution)：

**Linux** 作業系統是自由軟體最成功的典型範例，因為有許多人投入開發各式各樣的應用軟體。由於這些自由軟體散落在各處，而且各有不同的語系，所以花許多時間找尋。於是拿 **Linux kernel** 搭配多種應用程式，配合自行設計的安裝介面，包裝而成一 **Linux 發行版(Linux Distribution)**。

**Linux Distribution = Linux Kernel + Linux 應用軟體 + 安裝方式**，因為任何人都可以拿 **Linux** 所公佈的 **Linux kernel** 搭配各種應用程式或語系而組成一種 **Linux Distribution**，所以造成 **distribution** 版本眾多，但是 **kernel** 只有 **linux** 所公布的一種，各類版本的差異在於架構、應用程式管理及安裝方式的不同。

#### 各樣 **LINUX Distribution** 簡介(列表)

版本	簡介
OpenLinux caldera	公行發行，圖形化的安裝設定介面，容易安裝以及設定簡單，附贈商業版軟體
Red Hat	目前銷售量最高，世界上最流行的發行版，有強大研發團隊。安裝簡便，管理容易， <b>RPM</b> 軟體安裝方式，適合初學者使用
Debian	<b>GNU</b> 計劃中的 <b>OS</b> ，系統設計安全嚴謹，發展人員、所包含的軟體種類最多，缺點是自由度高不易入門，適合進階人員。
Ubuntu	"Ubuntu" 是一個古非洲語，意思是 "人道待人" (對他人仁慈)。Ubuntu 也意謂著: "我們都是同源一體"。" Ubuntu Linux 將此 Ubuntu 的精神帶進軟體世界中。Ubuntu 是一個以 <b>Linux</b> 核心為基礎的作業系統，可自由地取得並擁有社群及專業的支援。它是 <b>Debian</b> 的一個支系。
Linux Mint	<b>Linux Mint</b> 是一種基於 <b>Ubuntu</b> 開發出的 <b>Linux</b> 作業系統，其特色為簡潔。現在它的最新版本為 <b>Linux Mint 13 "Maya"</b> (2012 年 05 月 23 日更新)，是目前 <b>Distrowatch</b> 排行榜上第一名的 <b>Linux</b> 發行版。他繼承了 <b>Ubuntu</b> 的眾多優點，同時也在 <b>Ubuntu</b> 的基礎上加入很多自己優秀的特性。其標的是提供一種更開箱即刻可用的使用者體驗，免去使用者要安裝系統後還要自己安裝一系列滿足基本使用需求的軟體。它包括提供瀏覽器外掛程式、多媒體編解碼器、對 <b>DVD</b> 播放的支援、 <b>Java</b> 和其他元件。它與 <b>Ubuntu</b> 軟體倉庫完全相容。
SuSE	歐洲最流行的 <b>distribution</b> ，支援各國語言，國際化最深。
TurboLinux	支援日文、中文簡體及繁體，安裝簡單，提供中文化的軟體
slackware	早期流行的 <b>distribution</b>
Mandrake	源於 <b>RedHat</b> ，與 <b>RedHat</b> 很像，但使用習慣與界面與 <b>Windows</b> 相當，是個很好的入門級 <b>DeskTop Linux</b> 產品，中文化最完整的套件
Trustix	也是源自於 <b>RedHat</b> ，以安全性為號召的產品，因其不具有 <b>X-windows</b> 套件，

	故只適於用來當 <b>Server</b> 之套件產品
<b>KNOPPIX</b>	<b>Linux Live CD/DVD</b> 之主流，在免安裝、圖形介面的前提下，讓使用者可以很方便地使用 <b>Linux</b> ，而不用考慮複雜的安裝程序。其整個 <b>OS</b> 還是源於 <b>Debian</b> 為基礎而實作出來的 <b>Linux</b> 系統。
<b>B2D</b>	國內目前專門針對 <b>Server</b> 而設定的 <b>Live Cd</b> 系統，也是源於 <b>Debian</b> ，而全中文化及簡單的操作介面，大大地降低中小學網管人員進入 <b>Linux</b> 的門檻。
<b>EZGO 系列</b>	由國內熱心人士組成的團隊，修改於 <b>ubuntu</b> 系列的 <b>OS</b> ，將之修改為適合學校裡教學、應用及電腦教室教學使用之 <b>OS</b> 套件，目前最新版為 <b>EZGO X</b> 。

而 **CLE** 是 **Chinese Language Extension** 的縮寫，稱為中文延伸套件，是一群中文相關軟體的組合，主要建構在 **Redhat Linux**、**Mandrake**、**SuSE** 等 **distribution** 上，提供使用者一個良好的中文環境。

#### 四、自由軟體基金會與 **GNU** 組織

**GNU( GNU's Not Unix)**: **Richard Stallman**，在 1993 年所提出的一個理想中的「Free Unix」作業系統，號召網路上的同好一起加入這個 **GNU Project**。後來 **Stallman** 在 **MIT** 附近成立自由軟體基金會(**Free software foundation,FSF**)，全力投入 **GNU** 的發展計劃。

**GPL(General Public License)**: **GNU** 的自由軟體精神，是標榜開放原始碼(**Open Source**)，**stallman** 認為程式原始碼應該自由流通，不應被軟體公司據為己有，唯有讓程式碼自由流通，才能不斷提高軟體的品質。於是 **GPL** 成為 **GNU** 推動自由軟體的規範。任何程式設計人員在取得原始碼，都可以自由使用傳播以及修改軟體的原始碼，但不能據為己有（謀利）。所以自由軟體的精神不是在於免費使用，而是原始程式碼可以自己的流通並且修改散佈。

#### 五、**Linux** 會取代 **Windows 9x** 嗎？

電腦的應用可以分為桌面應用（工作站／客戶端）和伺服器應用。為伺服器設計的性能可靠、穩定的 **UNIX** 作業系統，與為使用者設計人性化介面的 **Windows 9x**，一直是長期並存的主流作業系統。

**Linux** 是一種類 **UNIX** 作業系統，其設計上是對穩定性的考慮比易用性考慮得要多得多，它不是一種桌面系統，它也不是為了取代桌面系統而出現的。所以 **Linux** 不會取代 **Windows 9x**。

（本節主講按:不知您是否用過 **Windows NT workstation** 或 **2000 profational**? 它不是當 **server** 的版本，顧名思義就是工作用，也就是不是用來 "打電動" 用的，用起來感覺會比 **windows 9x** 穩，目前 **Linux** 的 **Desktop** 版本\*，以小弟的感覺已超越 **windows** 的好用）

#### 六、**Linux** 對軟體業有什麼影響？

最早，程式是程式設計師思想的結晶，是人類共同的文化。後來，微軟公司的創始人比爾·蓋茲先生建立了一個全新的軟體產業模式：程式是要花錢買的，原始碼是程式員的知識財產，是需要保護的。也因為如此，將比爾推到了世界首富的地位。如果說，微軟的 **Windows** 為世界做了很大的貢獻的話，那麼它也給軟體業的發展帶來了壟斷和銅臭。自由軟體基金會

FSF 就是想通過 GUN 計劃來打破這種狀況，再次讓所有人共享程式設計師思想的結晶——原始碼。

## 七、如何學習 **Linux** ？

- 多閱讀 **Linux** 書籍
- 敢於嘗試，經常使用 **Linux**，從實踐中學習 **Linux**，別怕「try again」
- 經常透過網路與 **Linux** 愛好者們互相交流，提昇功力。

## 八、**Linux** 的優點：

### 1. 可攜性：

可以將作業系從某個平台移植到另一個平台，而它依舊能正常地運作的能力。可攜性可以讓執行 **Linux** 的電腦平台，正確和有效地和其它沒有加裝特別、高價的通訊介面的其它平台溝通。現今其他的作業系統，沒有一個能做到這樣。

### 2. 應用程式：

大部分的人需要靠電腦來從事工作，今日 **Linux** 已有數以萬計的應用程式可以使用，包括了試算表、資料庫、文書處理、各種電腦語言的應用程式發展環境，和可以連上網路的通訊套裝軟體，還有許多遊戲。

### 3. **Hacker**（駭客）：

基本上，**Linux** 是由駭客所創造和使用的。**Hacker**（駭客）這稱呼在現今社會上有著負面的含意，但這個字並不意謂著電腦駭客是罪犯。駭客通常意謂著某個人更深地，並且有興趣地 **hacking** 系統。**Hacking** 通常意謂學習所有關於系統的事情，進而沉迷於系統，而且在系統出了問題後，有能力去修復這系統。他們視他們所沈迷的系統為快樂的工作，對於賺錢和尋求報負並不感到興趣；然而有些駭客超越了這個界線，則變成了駭客常被稱呼的另一種名稱，怪客（**cracker**）。如果您對 **Linux** 充滿好奇，並且想去學習更多相關的知識，那 **Linux** 是個有完整功能，而且免費的版本，您可以用最高權限做任何想做的事。

## 九、把 **Windows** 扔掉的理由：

- **Linux** 比較安全：**Linux** 雖然不是對所有的線上攻擊都免疫，但是具有更嚴格的安全性設定，所有 **Linux** 在網頁伺服器市場的高成長就不奇怪了。（本節主講按：它的另一個安全的主因是因為它的原始碼開放，全球隨時都有成千成萬的人在檢視它是否有何漏洞，bug。不開放源碼的軟體，就算你的公司有多大，也不會隨時有那麼多的人在 debug.）
- **Linux** 可以免費使用：**Linux** 基本上是免費的，您不需要付錢給任何人，這是 **Windows** 做不到的。**Webmin**：你可以利用簡易的網頁介面來管理你的電腦，或是你網路上的每一台電腦。**Webmin** 是一個網頁驅動的管理系統，可以組態 **Linux** 上所有設定。（基於網路安全上之考量，不建議使用這個功能）**Internet**：**Linux** 之所以能成為世上成長最快的作業系統，主要因素就是 **internet**。
- **Linux** 是自由的——不受約束的：想要使用 **Linux** 的人們可以開放的方式進行設計，沒有任何隱藏的進程。
- 速度：**Linux** 的自訂能力，可以輕易地把你的系統拆成單純的個體，使速度提昇。

- **Linux** 是開放原始碼系統：把原始碼公開，代表的是你不只可以擁有軟體，也可以擁有程式之所以為之的原始碼。因為你可以從中了解到軟體的行為，甚至依自己需求進行修改。這對一群人、一個公司或政府機關特別有用。
- 共同檢閱(**peer review**)：還記得幾年前，**Netscape** 宣稱 **Microsoft** 將 **IE** 分開的話，**Windows** 的其他部分不能獨立地正常運作。主要是原始碼是保密的，如果是可以共同檢閱的話，就不會有這種爭論了。
- 更新：要維持最新的版本是非常花錢的，**Linux** 核心和大部分支援的程式也都會定期更新，不過更新都是免費的。
- 社群導向：**Linux** 是個非常社群導向的系統，因為人們想要使用它而存在，而想要使用它的人可以決定它會變成的樣子。
- 低的經常費用：你曾在只有 **8MB** 記憶體 的 **486** 電腦上跑過 **Windows98** 嗎？呃，不用試了。不過，你可以在這種系統上執行 **Linux**（不過 **X Window** 還是跑不動）。
- 可自訂：**Linux** 軟體的自訂通常比較有彈性，因為它是在開放原始碼的環境中開發出來的。選擇的自由正是開放原始碼哲學的中心主旨。
- 公平的賽場：任何有能力的人，都可以為 **Linux** 發展軟體，核心中並沒有私屬的秘密。
- 新技術：**Linux** 都站在技術的最前線。
- 交互操作性：你曾在 **PC** 上讀過 **Mac** 的軟碟片嗎？當然利用軟體可以完成這項工作，但是在 **Linux** 上都有支援，相容不是問題。
- 穩定性：使用 **Windows** 系統時，可能常有必須重新開機或出現藍畫面的經驗，但 **Linux** 的程式中，很少能夠卡住整個系統的。

#### 十、使用 **Linux** 的缺點：

- 缺乏技術支援
- 安裝不易，並非在所有的硬體平台都能執行。不過安裝的方式已愈來愈人性化了。

#### 十一、**Linux** 檔案結構介紹：

如果您玩過 **DOS** 的話，相信這對您不是很難，但如果您一直以來所接觸的電腦都是 **Windows** 的話，不好意思，您可能需要認真操練操練才行。(這裡我要事先提醒一點的是：在 **Linux** 下面，字母的大小是有分別的，一定要留意哦！)

在 **Linux** 裡面，檔案系統都從一個稱為 **root** (寫作 **/**) 的根目錄開始，離開了這個 **/** 一切免談就是了。在 **/** 下面是一些檔案子目錄，而子目錄下面也有其自己的檔案和子目錄，再下面的子目錄也一樣。整個檔案結構就像樹狀一樣：一切從根開始，然後分幹、分枝、分杈、分頁、...，這樣一直延伸出去。我們稱這樣的結構為層階(**hierarchy**)式結構。一般的 **Linux** 檔案系統，其層階結構如下：

各目錄的名稱和安排，當然是有一定標準的，其功能主要如下：

目錄名稱	存放檔案類型
<b>/boot</b>	系統開機核心所需檔案
<b>/etc</b>	系統設定檔案

目錄名稱	存放檔案類型
/home	使用者家目錄所在
/dev	設備檔案
/lib	函式庫檔案
/bin	一般執行檔，系統管理員需要
/sbin	系統執行檔，系統管理員需要
/usr/bin	一般執行檔，普通用戶需要
/usr/sbin	系統執行檔，普通用戶需要
/tmp	臨時檔存放目錄
/var	系統資訊/設定檔

總而言之，檔案系統所使用到的任何東西都以‘檔案’的形式出現，包括目錄本身也是一個特殊的檔案。就連所有的硬體設備，也必須在檔案系統上面有相對的檔案，比方說，`/dev/fd0`就是系統的第一個軟碟機，諸如此類，應該找到許多例子。

#### 資料來源：

1. <http://www.study-area.net>
2. 台南縣莊士毅老師的 Linux 研習講義 <http://s7.tnc.edu.tw/~linux5/>
3. 台南縣謝遠敦老師的 Linux 研習講義 <http://www.dwhs.tnc.edu.tw/linux/>
4. 台南縣王勝忠主任的 Linux 研習講義 <http://s7.tnc.edu.tw/~linux6/>



# CentOS 6 的安裝

高雄市政府教育局  
資訊教育中心  
網路組 張宏明  
更新日期：2012/11/14

## 一、規劃：

和其他任何系統的安裝一樣，事先的規劃是非常重要的。我們在安裝 Linux 之前，不妨先問問自己：

- ✧ 想拿它做什麼用途？
- ✧ 網路範圍如何？要不要劃分子網路？各主機的名稱和 IP 是什麼？
- ✧ 硬體是否支援？(如跑 Xwindow 要確定顯示卡可以支援，網卡也應該找些比較普通型號的，等等) 關於 RedHat Linux 的硬體兼容清單可以從 <http://www.redhat.com/support/hardware/> 獲得。
- ✧ 另一個常要考慮的題目是：如何劃分硬碟？如果您要跑多重系統的話，這是非常重要的，因為硬碟一經劃好，以後要重新劃分是非常頭痛和冒險的。

當我們開始玩 Linux 的時候，先劃分單一分區給 Linux 好了。等以後有經驗了，可以多切幾個，分別為 /、/boot、/usr、/home、/var 等目錄劃分各自分區。這樣在安全和效能上都有所改善，如果系統要是意外掛掉，收復起來也容易些。假如您現在不知道我說什麼，儘可以不理，等將來您對 Linux 有心得了，提出的建議可能比我這裡的更好呢！

通常這方面的考量是多方面的，主要看主機的使用狀況：

- 無論如何 / 不要太小。
- 如果您怕遇到 1024 磁柱的問題，那將 /boot 掛在較前面，幾十 MB 就夠了。
- 如果您打算裝一大堆程式，那 /usr 要大點。(例如 RH71 全裝的話，給它 2~3G 不過份)
- 如果您的 user 很多，那 /home 也要大點(例如提供 imap 服務)。就算 user 不多，也建議獨立出來，萬一以後系統重灌，/home 裡面的東西都可以保留著。
- 如果您不習慣清理 log，又怕它某天體積太大把系統分區吃掉了，那就將 /var/log 獨立出來吧。大小看您的 log 有多少和您需要保留多久的 log 了。
- /tmp 也可以獨立出來，如果您怕太多錯誤程式會在利用它的話。
- 將 ftpd 匿名登錄空間和 httpd 服務的 DocumentRoot 獨立出來也是個不錯的主意。
- 將 /var/spool 獨立出來，或 mount 在大容量的硬碟上也可接受(尤其在提供大量 pop 服務的時候)
- 將 /var/spool/news 獨立出來，然後將 inode 和 block size 降低一點。因為 news 文章通常不大，但數量很多。
- 那些要進行 quota 的目錄，最好也獨立出來，例如 /home 和 /var/spool/mail 這些。
- 可以考量保留一部份空間閒著，看日後那些目錄用得比較厲害，可以再獨立掛出來，或將其中的子目錄掛出來。

還有很多很多，用久了自然有心得。(當初我也是將所有東西都裝在一個 `partition` 上，後來就慢慢會切了。)

如果您的硬碟很大的話，(如超過 8GB)，在規劃的時候要注意 1024 磁柱的問題，也就是說：不要把 Linux 的啟動區(/boot)裝在 1024 磁柱(大約 8.3G)之後，解決的辦法是可以在較前面切一個小小的分區(50MB 已經非常足夠了)，然後給 /boot 這個掛載點使用；不過，也可以在 lilo 裡面使用 `lba32` 參數。

## 二、選擇安裝的方式：

### 1. 光碟安裝：

請以 ftp 的方式連上 [ftp.kh.edu.tw](ftp://ftp.kh.edu.tw) 在 `/pub/Linux/Caosity/centos/6/isos/` 下可找到各平台的光碟 ISO 檔，若您使用 INTEL CPU，通常是選擇 i386 平台，若您使用 INTEL EM64T 或 AMD64(如 IBM x206m)，則請選擇 x86\_64 平台。

#### **i386 平台您可下載**

CentOS-6.3-i386-bin-DVD1.iso

CentOS-6.3-i386-bin-DVD2.iso

#### **x86\_64 平台您可下載**

CentOS-6.3-x86\_64-bin-DVD1.iso

CentOS-6.3-x86\_64-bin-DVD2.iso

### 2. 網路安裝：

因為光纖到校之後，各校對外的頻寬有 10-100Mbps 的速率，再加上 Linux 套件發展快速，每季均有新套件出版，為避免一天到晚都在燒 Linux 的光碟，可考慮使用網路安裝之方式進行系統之安裝。

#### **I386 平台請下載**

<http://ftp.kh.edu.tw/Linux/Caosity/centos/6/isos/i386/CentOS-6.3-i386-netinstall.iso>，

x86\_64 平台(INETL EM64T 或 AMD64 CPU)請下載

[http://ftp.kh.edu.tw/Linux/Caosity/centos/6/isos/x86\\_64/CentOS-6.3-x86\\_64-netinstall.iso](http://ftp.kh.edu.tw/Linux/Caosity/centos/6/isos/x86_64/CentOS-6.3-x86_64-netinstall.iso)

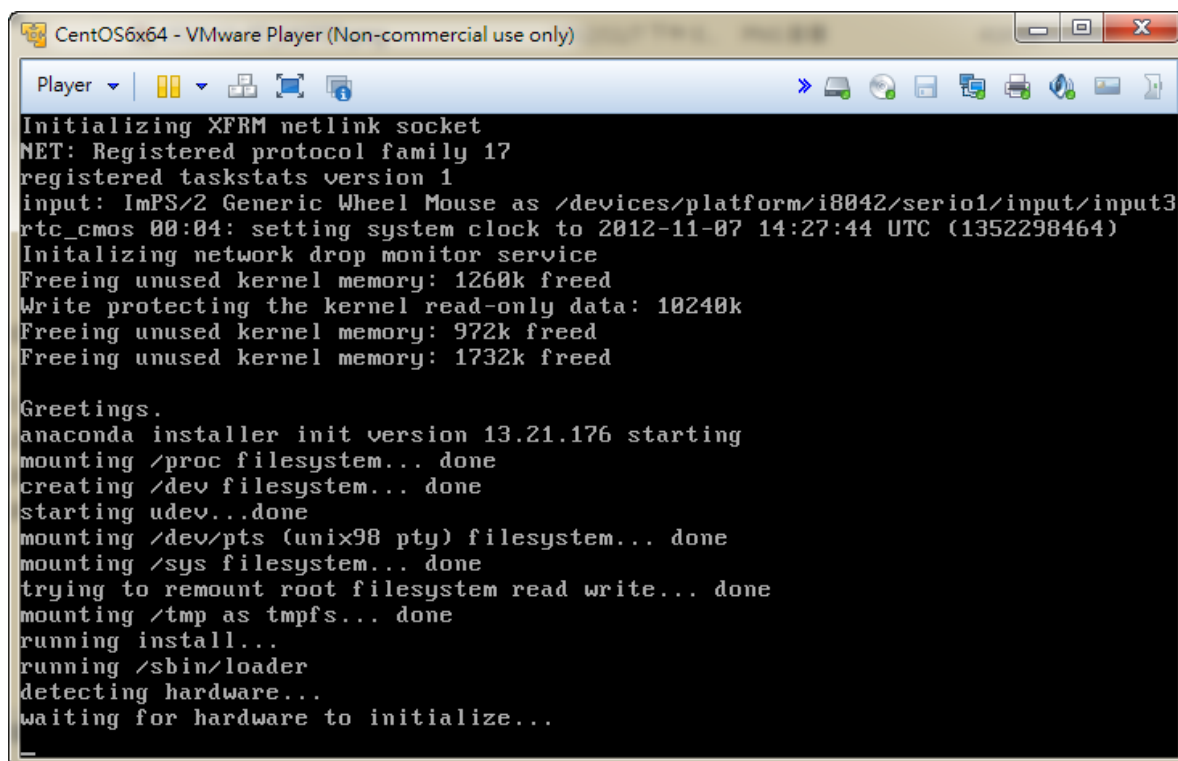
，並以 Nero 直接以『燒錄映像檔』燒錄出一張網路安裝光碟。

### 三、進行安裝：

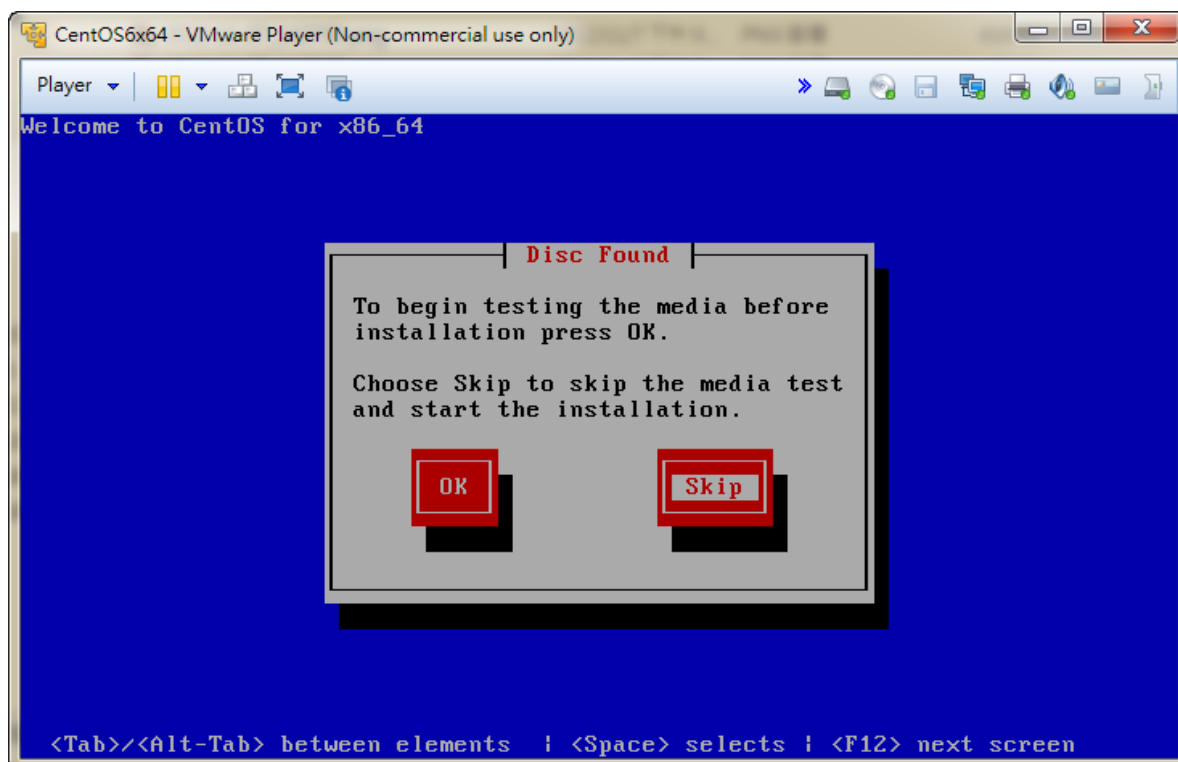
1. 因為 Centos 6 不支援 SAS RAID，若您的電腦是 IBM X206m，請在第一次開機時，進入 RAID 設定畫面，將 RAID 功能關閉，只單純使用 SCSI(SAS)功能。若您先前已經建立 RAID 的話，在 RAID 設定畫面中，先刪除 RAID 磁碟機後，再將 RAID 功能關閉。
2. 請先將電腦之 BIOS 設成由光碟開機。
3. 放入 Centos 6 第一片光碟片或網路安裝光碟片，然後將電腦重新開機由光碟開機，以便載入 CentOS 安裝程序。
4. 您看到的第一個畫面可以看到 CentOS，這裡您要選擇一種安裝模式。在預設情況下，系統直接停在第一個選項(**Install or upgrade an existing system**)，如果您的顯示卡可以被安裝程式正確抓到，按 Enter 會直接進入圖形化的安裝界面。但如果您使用預設的選項進入後，看不到設定畫面，就要改用第二個選項(**Install system with basic video driver**)。CentOS 從第 6 版開始，不再提供純文字安裝介面，這點要特別注意。
5. 在 x86\_64 平台安裝比較簡單，只要確認您的 CPU 是 INTEL 具 EM64T 技術或 ADM64 均可直接安裝，因整個平台均安裝成 64 位元的作業模式，才能夠完全發揮 64 位元 CPU 的效能。在這個安裝模式下，若您要安裝，就請直接按下 Enter 鍵即可。
6. 若您的機器不是上述屬於 64bit CPU 的話，記得改用 i386 的網路安裝光碟，用光碟開機後，按 Enter 進入安裝系統。



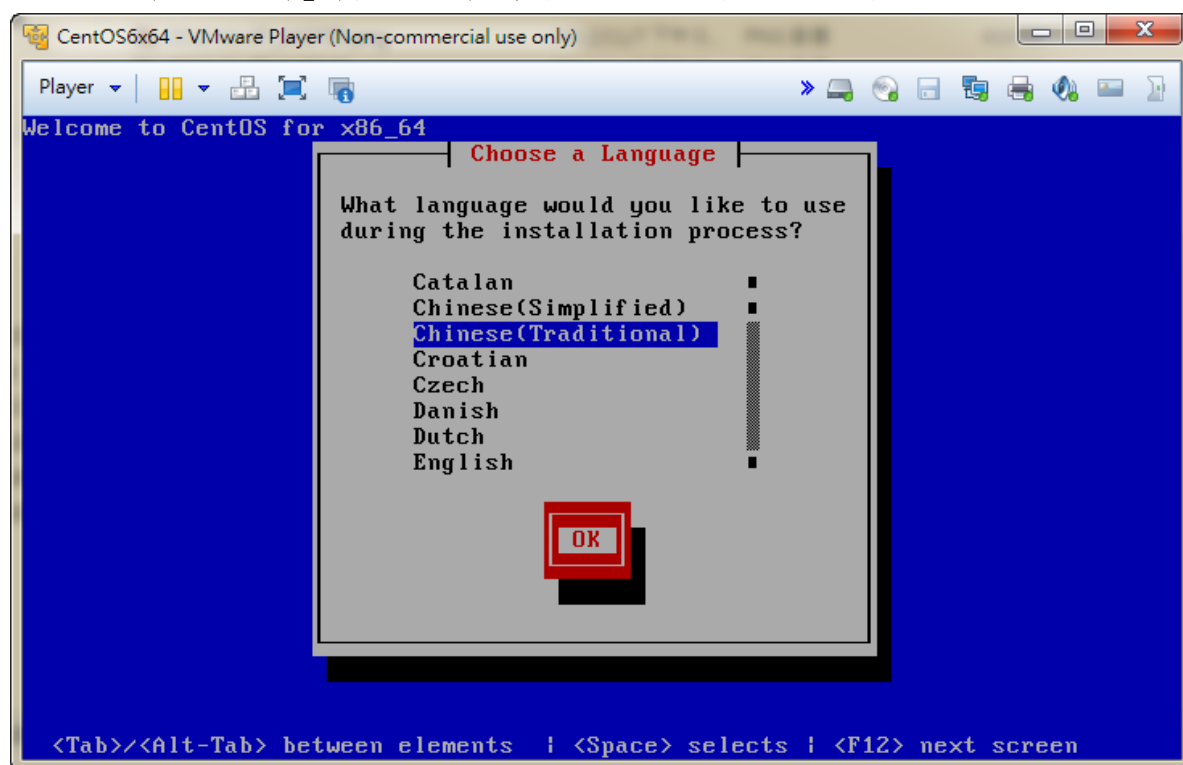
7. 進入安裝系統後，首先會先載入安裝系統，若硬體有 SCSI 介面的話，也會載入 SCSI 的 Driver，以便啟動 SCSI 介面來使用 SCSI 硬碟，若有 SAS 卡也同時會在此時啟動並找到 SAS 硬碟。



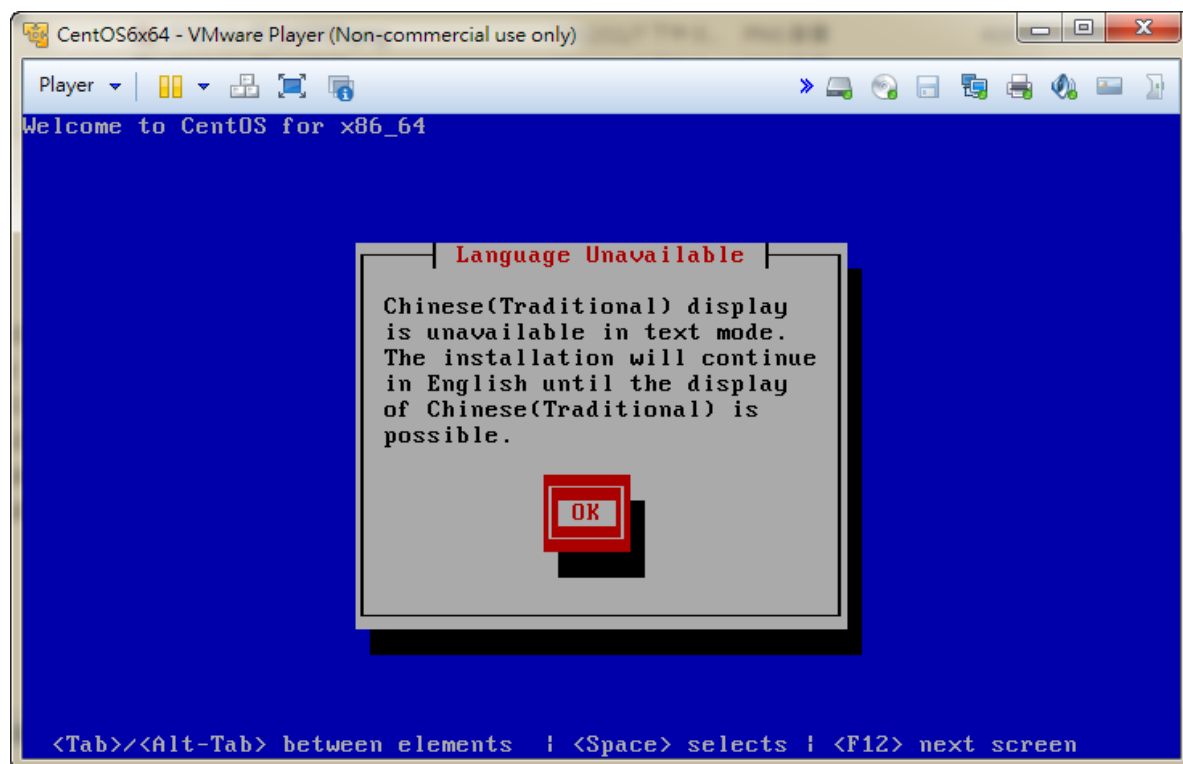
8. 接下來安裝系統會要求檢查光碟是否正確無損壞，因我們採用網路安裝，在此跳過光碟檢查。請您按「Tab」鍵將光棒選取「Skip」後按下Enter即可。



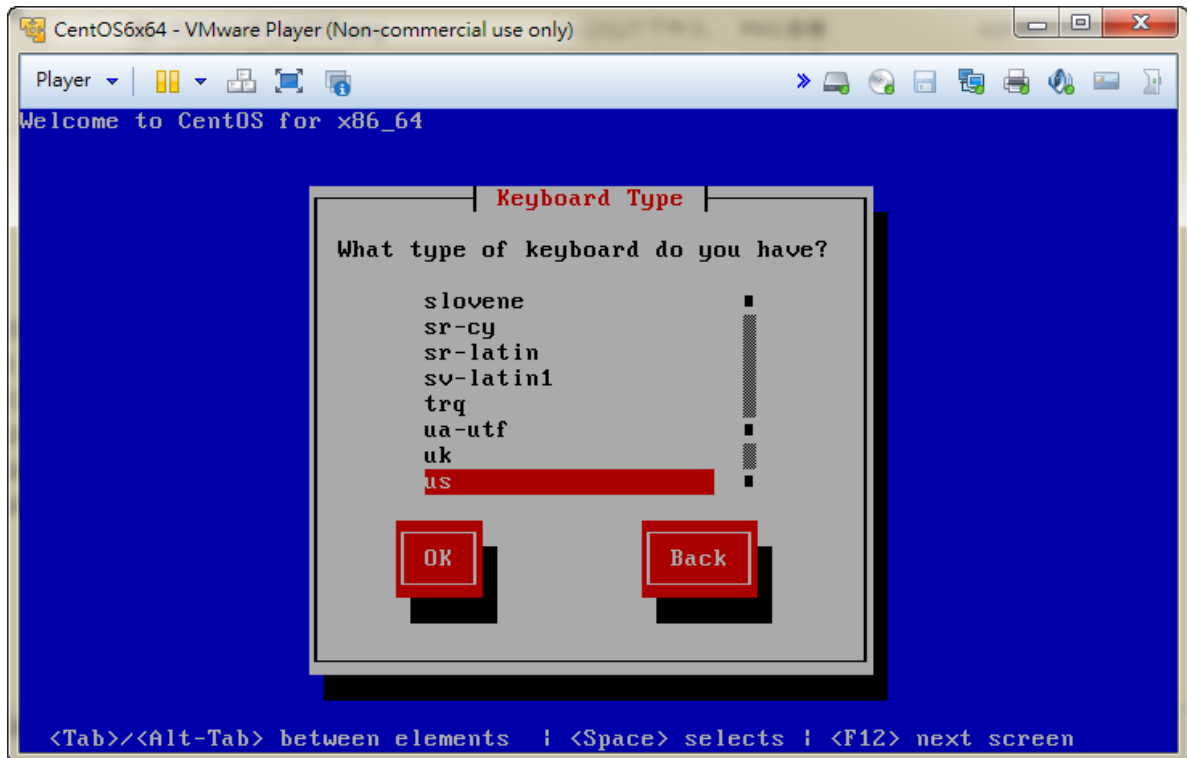
9. 接下來，您會看到第一個「Choose a Language」畫面，因為 CentOS 與 RHEL 完全相容，所以在文字模式安裝期間，是可以看到中文字的。為了方便起見，在此，請選擇「Chinese(traditional)」(繁體中文)以便後續的文字介面均能順利地以中文表示。



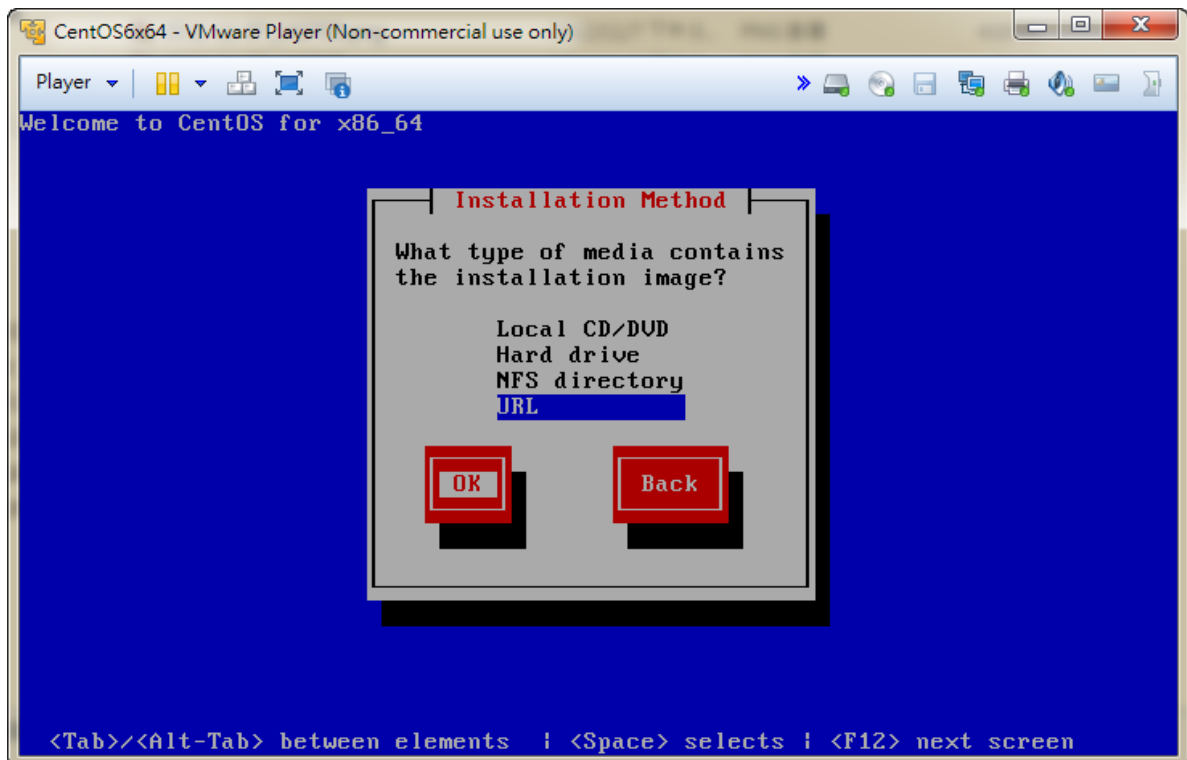
10. 因為剛剛選取正體中文語系，在安裝的程序中，是無法顯示中文的，所以系統會提示這個警告畫面。



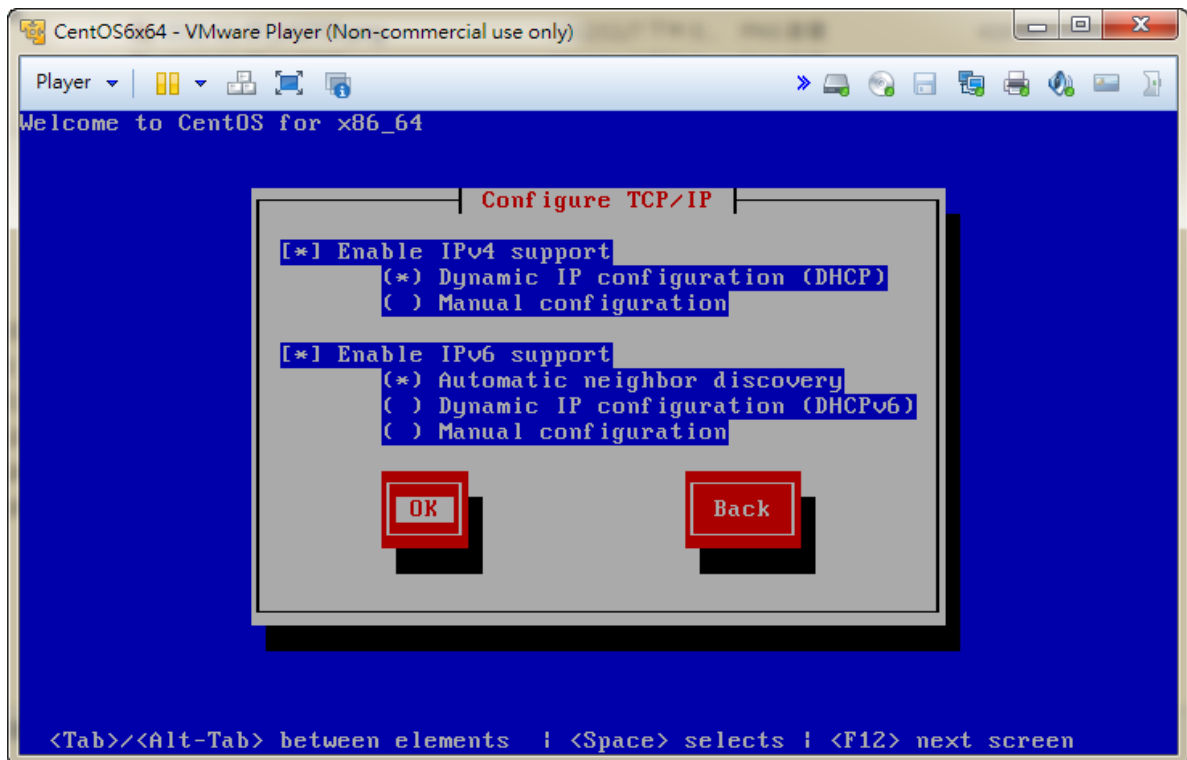
11. 接著，在“Keyboard Selection”畫面裡面，選 us 即可，按 Enter 繼續。



12. 再來選擇網路安裝的類型，資教中心這邊提供了 FTP 及 HTTP 兩種安裝服務，您可任選這兩種中任何一種來進行安裝。在此我們選擇以 HTTP 的方式進行網路安裝程序。

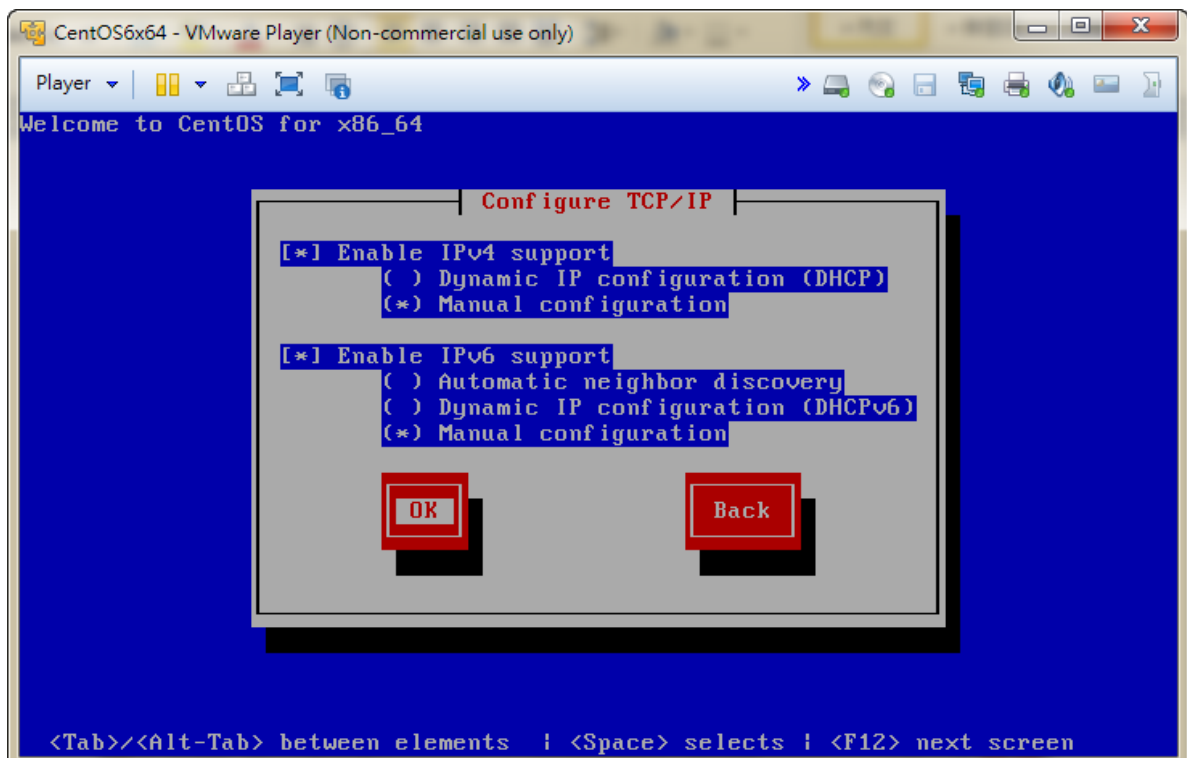


13. 因為要網路安裝，所以必須先設設網路卡的基本設定，以便能順利在網路中找到安裝的來源。若您是練習安裝，IPv4 可直接選取「Dynamic IP configuration(DHCP)」方式取得 IP、IPv6 可選取「Automatic neighbor discovery」方式取得 IPv6 IP 及其他網路設定值；



若您是要拿來當 Server 使用，請依您電腦的網路環境，設定相關的設定值。固定 IP 設定方式如下：

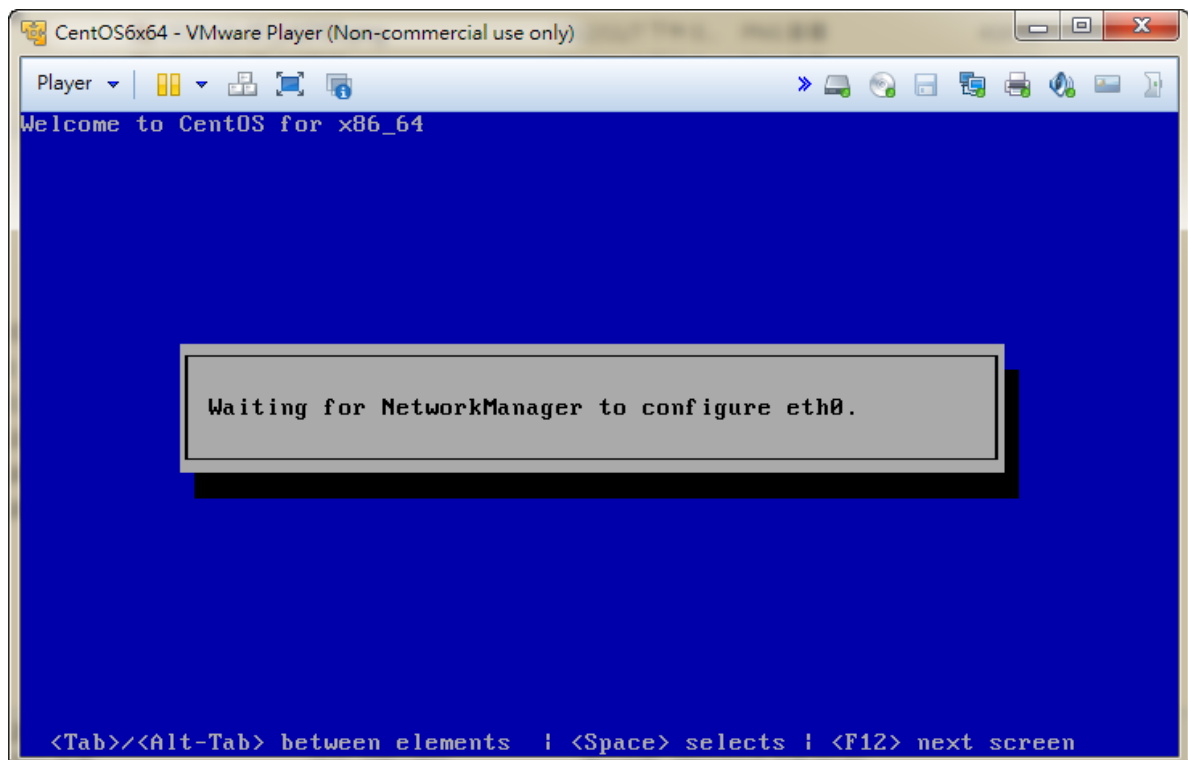
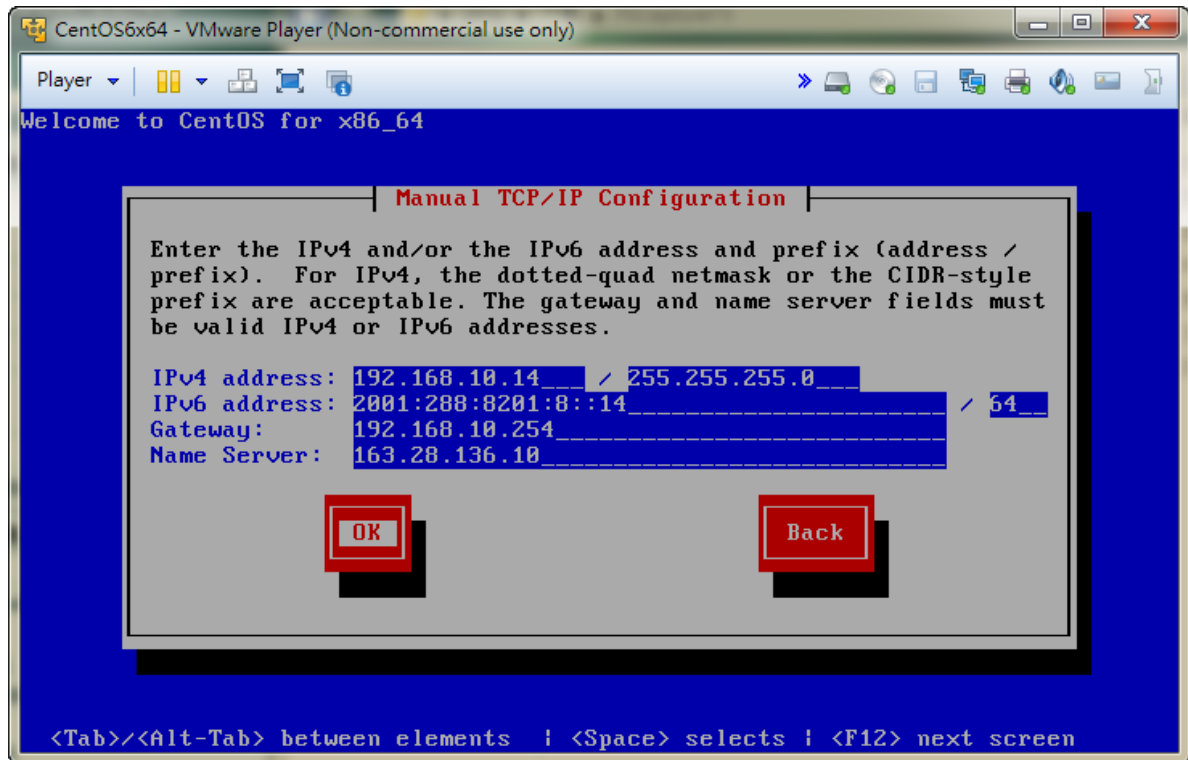
- (1)同時選取「Enable IPv4 support」及「Enable IPv6 support」的「Manual configuration」後按下「OK」。



(2)然後依序填入以下各欄資料:

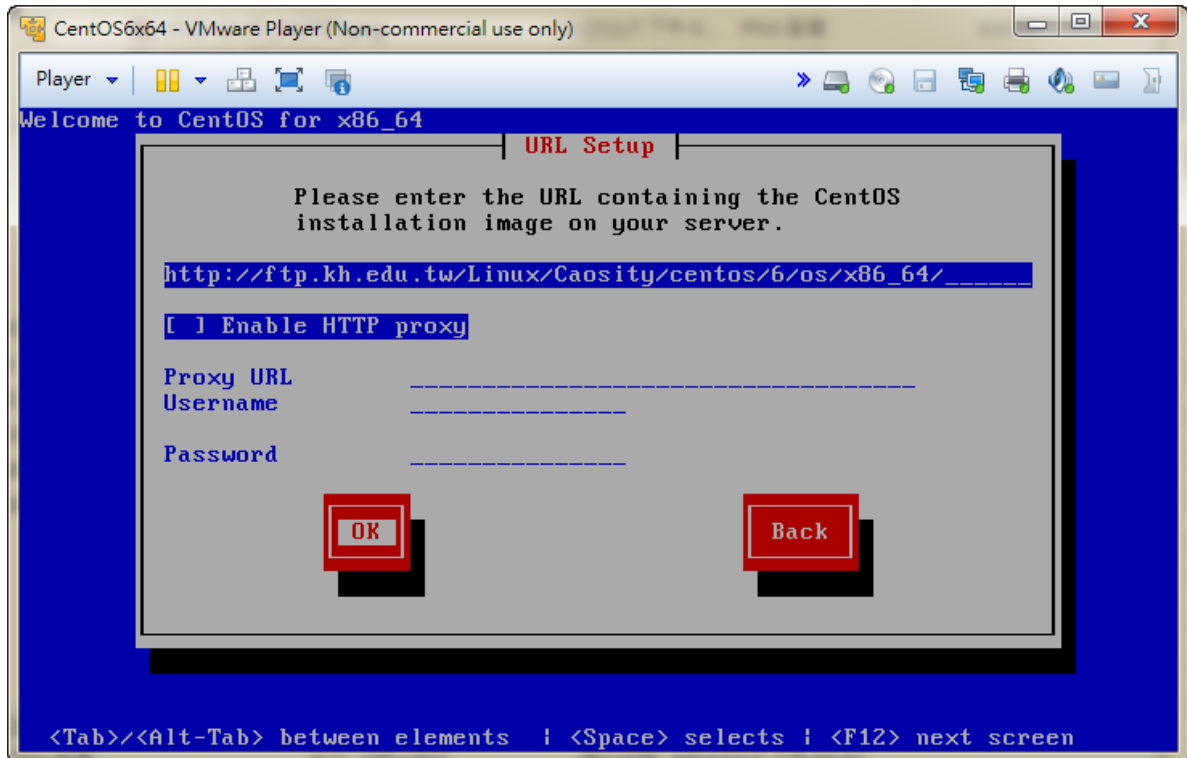
IPv4/IPv6 位址、子網路遮罩、預設閘道器的 IP 位址、第一個名稱伺服器位址。

完成後，請以 Tab 鍵選取「OK」鈕後，按下「Enter」鍵，以進行下一個程序。

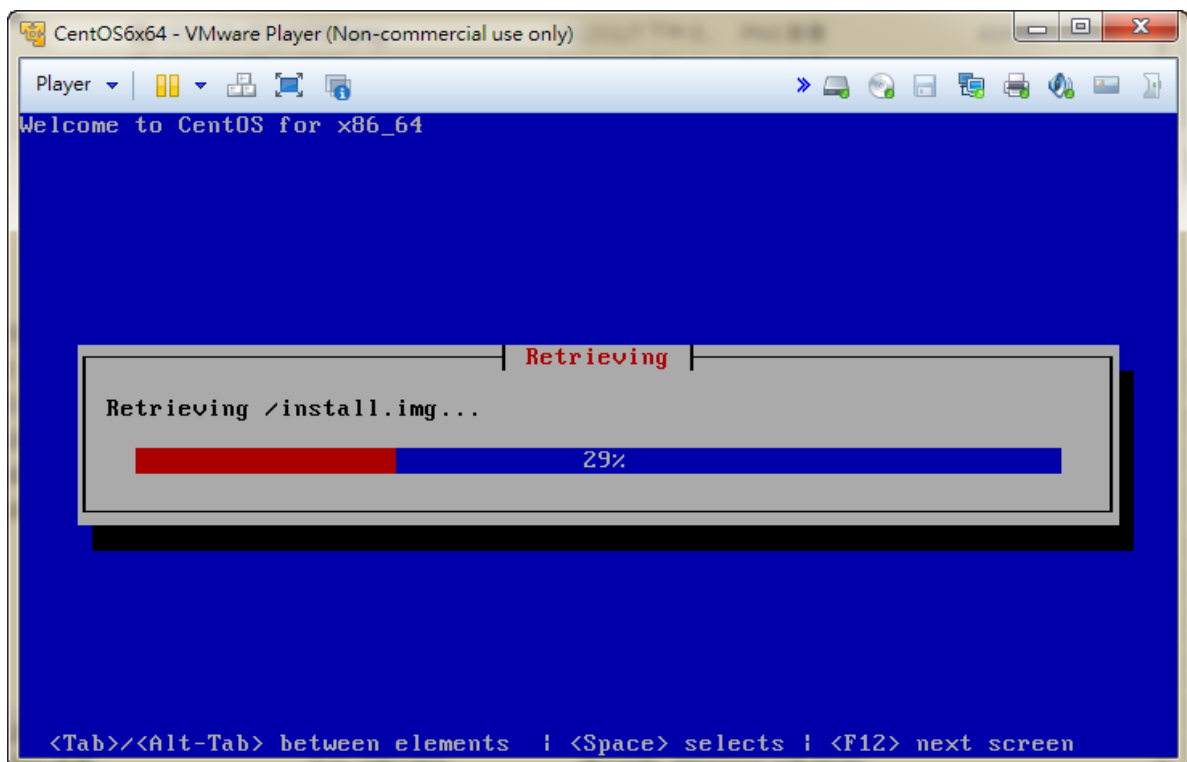




14. 網路設定完成後，再來是指定網路安裝的來源。您必須正確地指定安裝系統的檔案來源與確實路徑。若您使用市網提供的 HTTP 安裝服務，請輸入下列資料：
- i386 平台為 <http://ftp.kh.edu.tw/Linux/Caosity/centos/6/os/i386/>  
x86\_64 平台為 [http://ftp.kh.edu.tw/Linux/Caosity/centos/6/os/x86\\_64/](http://ftp.kh.edu.tw/Linux/Caosity/centos/6/os/x86_64/)
- 完成輸入後，按下「OK」即可進行讀取的動作。



15. 接下來，開始讀取系統安裝程序檔。



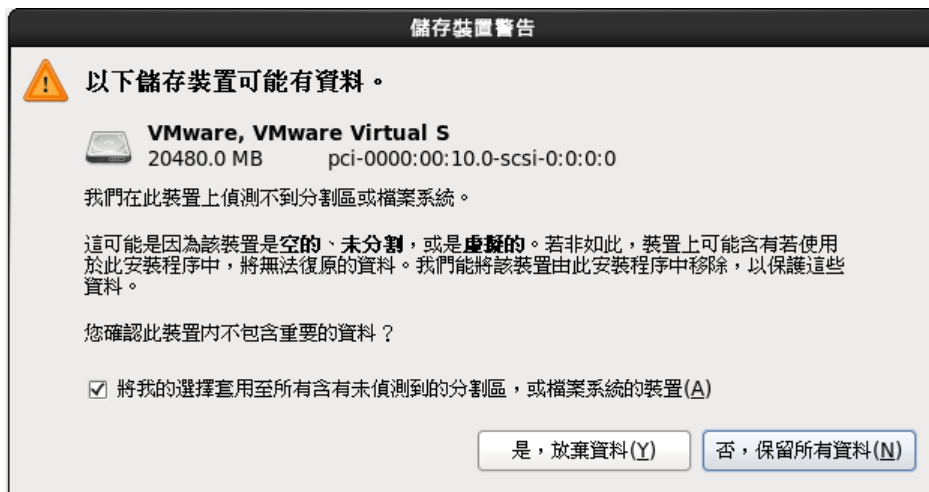
16. 因為各類主機的顯示卡種類繁多，萬一您無法直接用圖形模式安裝時，請重新開機後，在開機選單選取「Install system with basic video driver」後重新進行安裝程序。(請回到步驟 4.)
17. 若您的硬體偵測順利的話，您就可以看到 CentOS 的歡迎畫面了。請按下「下一步」進入下個安裝程序。



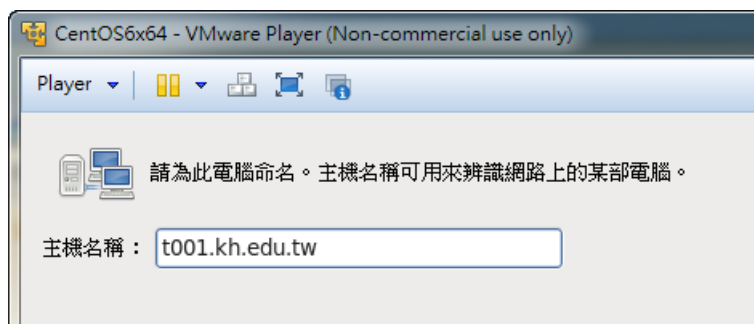
18. 接下來要選擇系統安裝的儲存裝置：因為近幾年已經發展的大型儲存設備如 SAN、NAS、FCoE、iSCSI 等，所以從 CentOS6 開始，也提供這些先進的儲存媒體供系統安裝使用。因為我們是安裝在本地(local)硬碟，所以我們先選取「基本儲存裝置」。



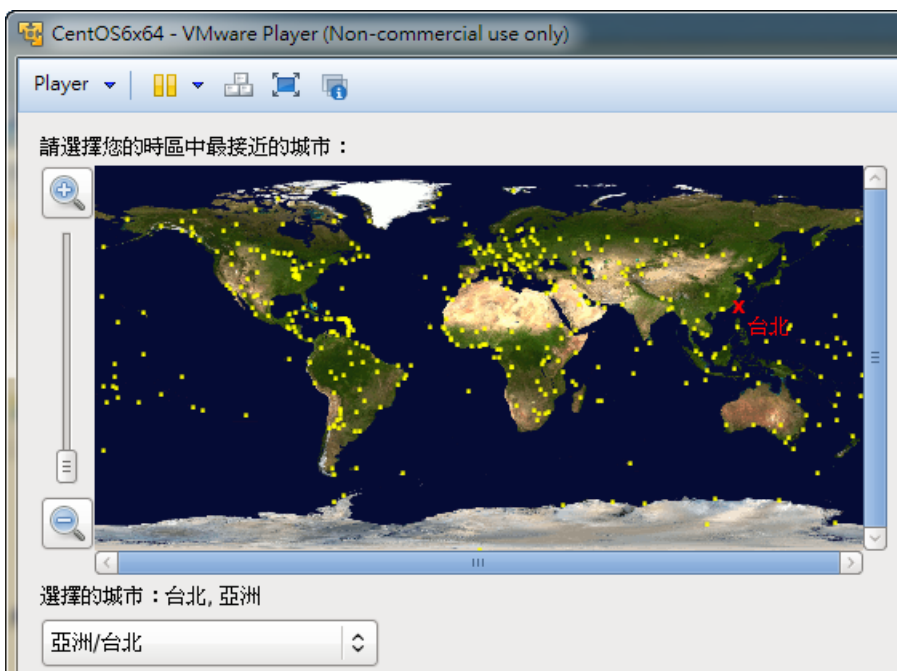
19. 接下來是分割硬碟，一般而言，我們會手動來分割所要使用的硬碟空間。呵呵~~~ 如果您‘不幸’碰到下面這個「儲存裝置警告」畫面，是說您的硬碟還沒準備好。不用緊張啦，直接按下「是，放棄資料(Y)」進行清除資料的動作就是了。如果您沒看到這個畫面，那根本不用擔心這個，直接看下一步就好。



20. 接下來就是設定主機名稱了。這裡的設定，通常會填上自己的主機名稱。嗯，這個可以隨便亂填啦，只要不和其它機器名稱衝突就行。如果您還沒註冊好 domain，那就隨便自己起一個橫看豎看都不會和別人 domain 衝突的名字吧。記住：不要用那些容易和別人衝突的名字，例如：abc.com (事實上有這個 domain 哦！)。



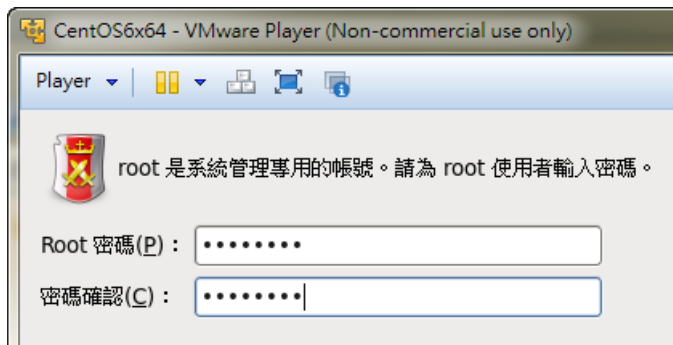
21. 再下來，是選擇時區。請選擇城市為「亞洲/台北」



「系統時鐘使用 UTC(S)」是要將主機的時鐘調成格林威治時間，通常我們不用選取，將主機時鐘維持在當地(local)時間。

☐ 系統時鐘使用 UTC(S)

22. 這裡要設定您的 root 帳號密碼，必須要選擇一個自己容易記，別人難猜的「合格」密碼哦。試試看吧，如果密碼太差，系統會不接受並且要您重新輸入，還要接連兩次輸入相同的密碼才放行。這個密碼一旦設好，一定要記著(但不要給別人看到，千萬千萬!)，否則，如果您沒有 root 密碼，可能做不了任何系統方面的修改，切記切記!



CentOS6x64 - VMware Player (Non-commercial use only)

Player | [Icons]

root 是系統管理專用的帳號。請為 root 使用者輸入密碼。

Root 密碼(P): [Password field]

密碼確認(C): [Confirm password field]

23. 然後系統會自動去搜尋可安裝 OS 的硬碟分割區域。
24. 我們先來試試看自動分割硬碟的部份好了！要讓安裝系統自動分割硬碟，只要選第一項目「使用所有空間」。



CentOS6x64 - VMware Player (Non-commercial use only)

Player | [Icons]

您希望進行哪一類型的安裝？

- ☒ **使用所有空間**  
移除已選擇裝置上的所有分割區。這包含了由其它作業系統所建立的分割區。  
提示：此選項會將已選擇裝置上的資料移除。請確認您已進行備份。
- ☐ **取代現有的 Linux 系統**  
只移除 Linux 分割區（由先前的 Linux 安裝所建立的）。這不會將您儲存裝置（例如 VFAT 或是 FAT32）上的其它分割區移除掉。  
提示：此選項會將已選擇裝置上的資料移除。請確認您已進行備份。
- ☐ **縮小目前的系統**  
將現有的分割區縮小，以便為預設的分割區格式釋放可用空間。
- ☐ **使用可用空間**  
保留您既有的資料和分割區，並只使用已選擇裝置上未分割的空間，假設您擁有足夠的可用空間。
- ☐ **建立自訂分割格式**  
使用我們的硬碟分割工具，以手動式地在所選擇的裝置上建立您自己的自訂格式。

想要看分割的結果或還要再修正的話，請勾選「檢視並修改分割區(V)」。

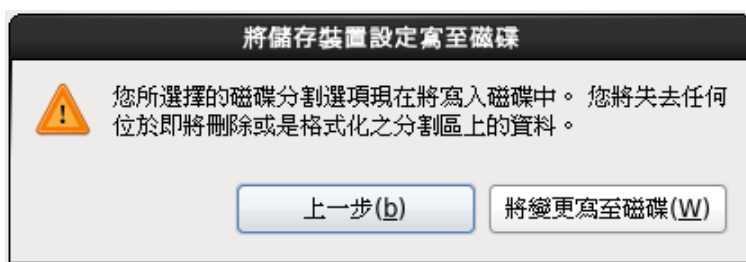
☐ 加密系統(E)

☐ 檢視並修改分割區(V)

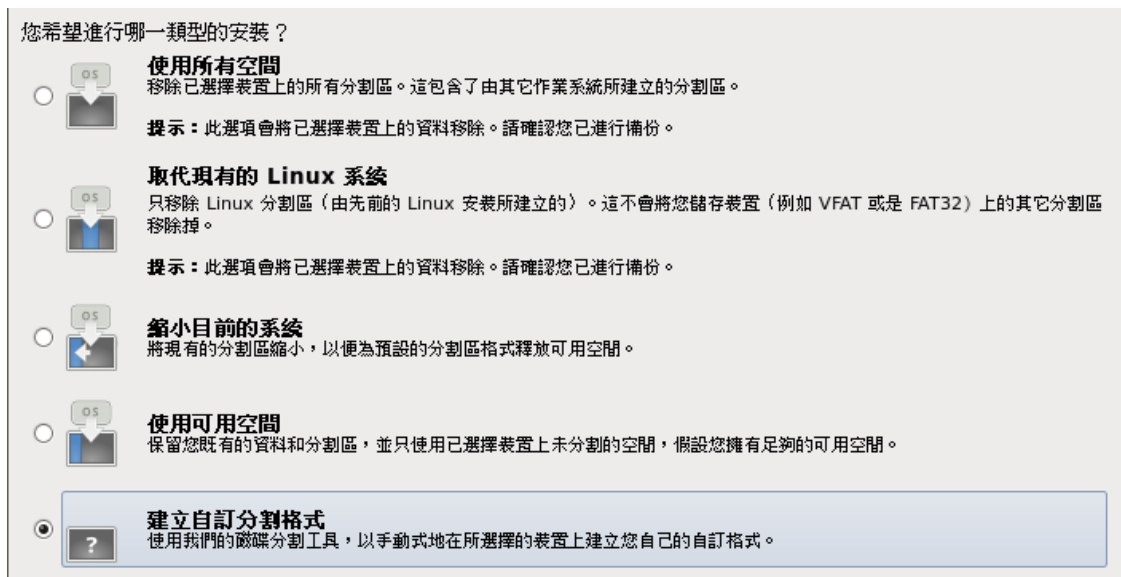
25. 若之前有勾選「檢視並修改分割區(V)」，會顯示自動分割的結果，若要再修正，請按下「重設」，若沒有問題，則按「下一步(N)」。

裝置	大小 (MB)	掛載點/ RAID/卷冊	類型	格式化
LVM 卷冊群組				
▼ vg_t001				
lv_root	15976	/	ext4	✓
lv_swap	4000		swap	✓
硬碟				
▼ sda (/dev/sda)				
sda1	500	/boot	ext4	✓
sda2	19979	vg_t001	physical volume (LVM)	✓

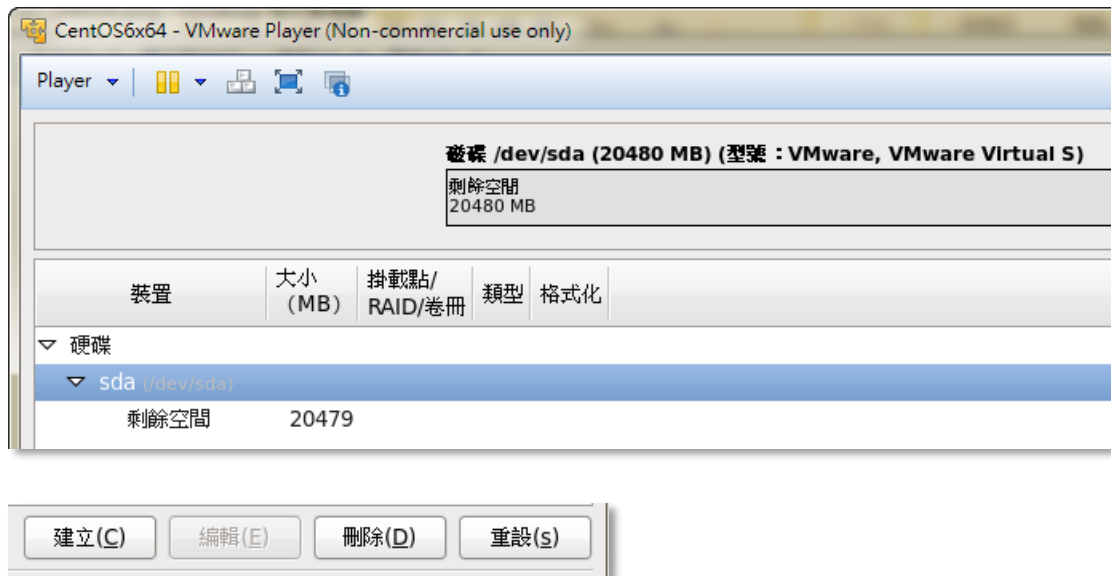
26. 事涉硬碟的分割，要謹慎一點，再跟您確認一次！



27. 若您想要手動分割新的分割區(Partition)，請選擇「建立自訂分割格式」後，按「下一步」即可。



28. 您必須先將游標點選硬碟裝置(sda)後，再按下「建立(C)」鈕。



接下來要選擇分割區的型態，這個版本的分割工具，已經非常完整了，可以分割 RAID 分割區及 LVM 功能。不過，我們初學者還是使用標準分割區方式做分割。



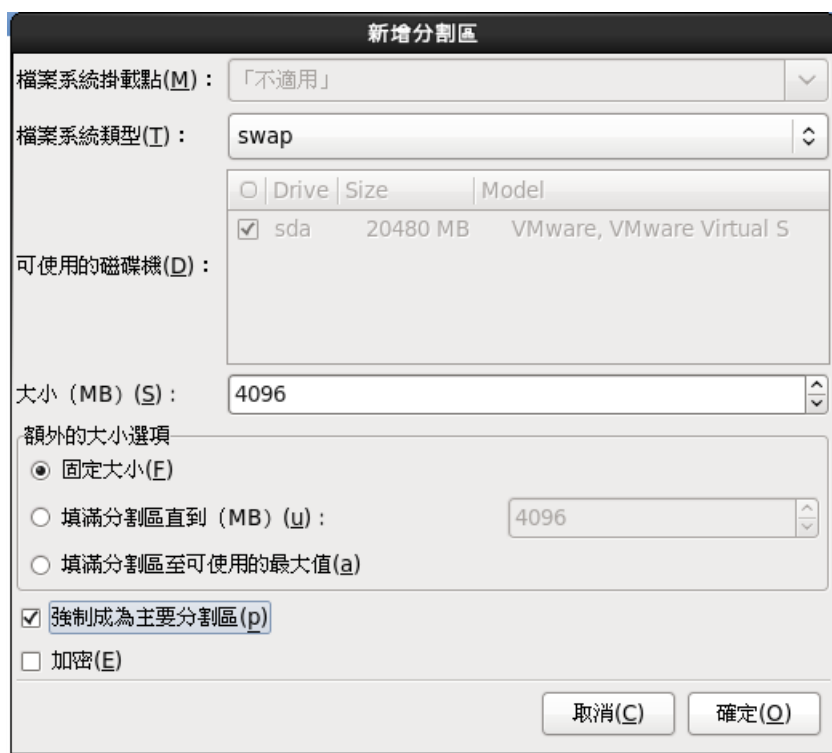
29. 我們先分割 SWAP 分割區：許多朋友在規範 swap 大小的時候，常不知道要切多大，有些人簡單的說：切 RAM 的兩倍就是了！這其實是很不科學的。您首先要知道 swap 是當您的 Physical Memory 不夠用的時候才使用的，假如您的實體記憶體非常充足，或許永遠用不到 swap。

那究竟要且多大呢？視實際情形而定啦。您要先將機器的角色和使用到的程式和服務做一個評估，得出您的總共 memory 使用量，然後扣除掉 RAM 的大小，就是 swap 的大小了。假如，您已經有 4G 的記憶體，難道別人說要兩倍您就切 8G 的 swap 不成？真有那麼多 RAM 的話，那只切 16M (一個 cylinder) 的 swap 未嘗不可，或完全不切 swap 也是可以的(不知道 CentOS 在安裝的時候是否接受？要試試才知道)。

比方說，您只有 128 MB，然後要裝 X window，跑一大堆服務，然後帳戶又很多、而且沒用 ulimit 和 resource 限制，那您可能使用到 300M 的記憶體，然後再加上 50% 的預留空間，也就是 450M，那減掉 128 就大概是 300M 左右。但，如果您可能會執行 database 或是大量連線的話，可能會用到 1G 以上的記憶體，那就再多切一點 swap。但是，畢竟 swap 是使用磁碟上的空間來虛擬記憶體，執行效率上當然會慢很多(我會建議您多加記憶體而不是增加 swap)。

然而，這裡僅是預估的值而已。要真正得出系統的記憶體使用量，要等機器滿載的時候執行 free 或 top 命令來看才準確。

我自己的習慣是：在分割 partition 的時候，總要預留一段 free space (大小看硬碟而定，如果是 40G 的硬碟，留個 4G 不為過)，看日後哪個 partition 或 swap 需要增加的時候，可以進行擴充。我非常建議您也學會如此規劃。



因為 SWAP 分割區對系統而言很重要，所以建議每次安裝，均要對這個 Partition 的磁區作檢查！



完成後，游標點選「確定(O)」即可完成分割 **SWAP** 分割區回到磁碟管理的畫面。



30. 回到「磁碟管理」畫面後，先點選硬碟代號的「剩餘空間」後，再選擇「建立(C)」，來分割其他的分割區。



因為 CentOS6 最基本的服務與工具(不含圖形介面)，安裝時需要近 1500MB 的空間，再加上設定檔與紀錄檔空間的考量；因此，日後不管您分割幾個 Partition，其可供系統安裝的空間總和必須大於 2.5GB 以上，才可讓系統順利地安裝與使用。若再加上圖形操作介面(Xwindows)的話，至少要準備 3.5-5.0GB 以上才夠用。

由於我們是在練習安裝，所以簡單地再分割一個分割區給系統的根目錄(/)用就好了：這裡我們選擇日誌型的檔案系統 **ext4**，來減少日後因斷電或不正常關機而導致檔案資料流失的狀況。完成後，游標點選「確定(O)」即可回到上一個畫面。



**新增分割區**

檔案系統掛載點(M) : /

檔案系統類型(T) : ext4

可使用的磁碟機(D) :

<input type="checkbox"/>	Drive	Size	Model
<input checked="" type="checkbox"/>	sda	20480 MB	VMware, VMware Virtual S

大小 (MB) (S) : 200

額外的大小選項

☐ 固定大小(F)

☐ 填滿分割區直到 (MB) (u) : 1

☒ 填滿分割區至可使用的最大值(a)

☒ 強制成為主要分割區(p)

☐ 加密(E)

取消(C) 確定(O)

31. 割完成後，我們可以再檢視一下，一共分割了兩個 Partition，一個是系統的 SWAP 空間，一個是系統安裝檔案的所在 / ；確定無誤之後，按「下一步(N)」。

CentOS6x64 - VMware Player (Non-commercial use only)


Player

請選擇一個裝置

裝置	大小 (MB)	掛載點/ RAID/卷冊	類型	格式化
硬碟				
sda (/dev/sda)				
sda1	4096	swap	✓	
sda2	16383 /	ext4	✓	

事涉硬碟的分割，要謹慎一點，再跟您確認一次！

**將儲存裝置設定寫至磁碟**

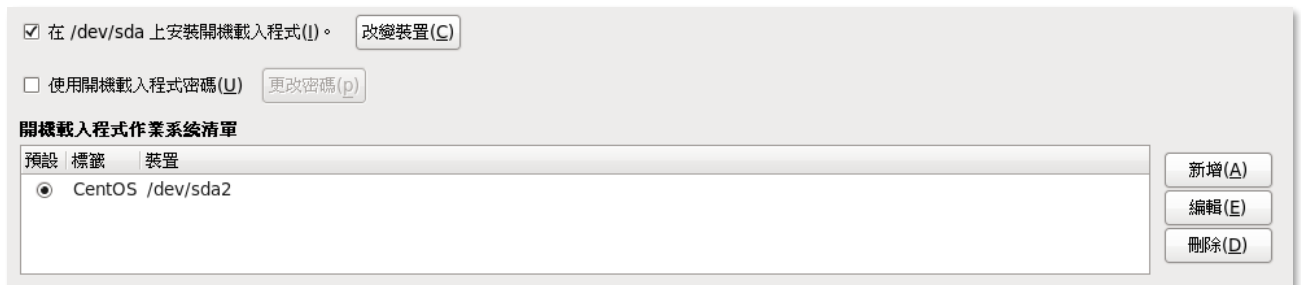
 您所選擇的磁碟分割選項現在將寫入磁碟中。您將失去任何位於即將刪除或是格式化之分割區上的資料。

上一步(b) 將變更寫至磁碟(W)

確認後，系統就開始將剛剛的設定，寫入硬碟，並進行格式化(format)的動作。



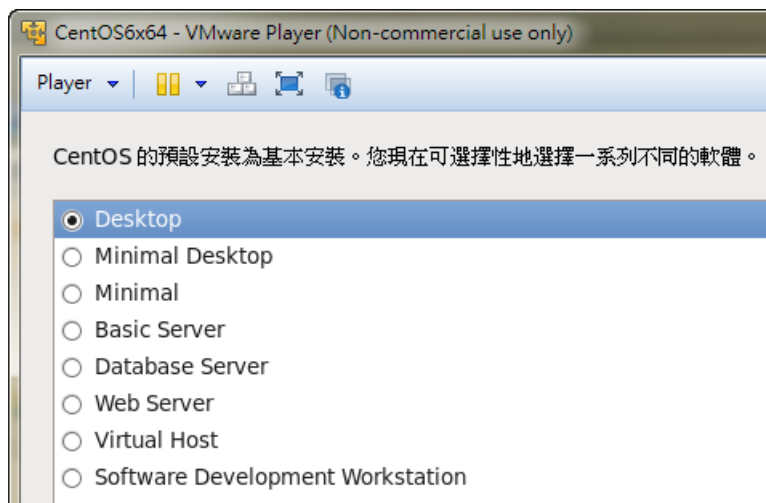
32. 接下來，是設定開機管理程式(Boot Loader)。安裝系統會自動選取您設定好根目錄(/)的分割區作為開機的預設選項，bootloader 則寫入於第一顆硬碟的開機區；若要做任何變更，請按下「改變裝置(C)」或開機清單右側的編修鈕，做進一步的設定。在此，初學者依系統預設的狀態設定即可。



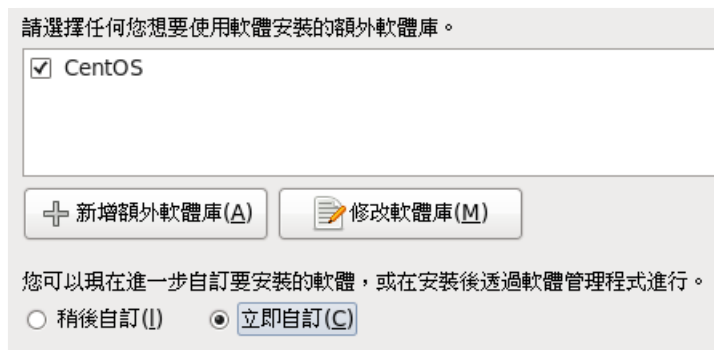
33. 開機管理程式安全的設定，若您很在意您系統在開機時就有密碼保護，您可啟動開機密碼來保護您的作業系統。但是若您的電腦是放置在有人員管制的機房中，而您又有必須從遠端控制電腦重新開機需求的話，建議不要啟用開機密碼。  
若要設定開機密碼，請勾選「使用開機載入程式密碼(U)」項目，並按下「更改密碼(P)」進行密碼的設定。
34. 接下來系統告訴您，正在蒐集安裝資訊。



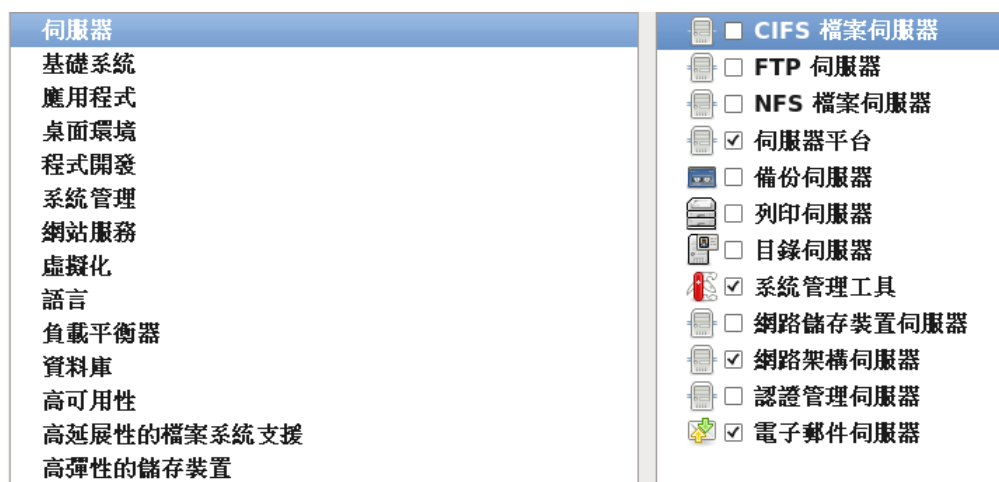
35. 接下來選擇要安裝的套件種類，因為初學，需要圖形模式及桌面工具比較方便，因此我們勾選「Desktop」。



要記得要勾選最下一行的「立即自訂(C)」，以便做細部安裝套件的選擇。然後按「下一步」進行下個步驟。



36. 現在我們要來選取要安裝到系統上的詳細類別，左邊選取窗為大類別，右邊選取窗為細部項目；您可以操作滑鼠勾選要安裝的各細項目，一切滿意之後，再按「下一步(N)」即可。



伺服器	<input type="checkbox"/> FCoE 儲存裝置客戶端	<input type="checkbox"/> 備份客戶端
基礎系統	<input type="checkbox"/> Infiniband 支援	<input type="checkbox"/> 儲存裝置可用性工具
應用程式	<input checked="" type="checkbox"/> Java 平台	<input checked="" type="checkbox"/> 列印客戶端
桌面環境	<input checked="" type="checkbox"/> Legacy UNIX 相容性	<input checked="" type="checkbox"/> 基礎
程式開發	<input type="checkbox"/> Mainframe 存取	<input type="checkbox"/> 大型檔案系統效能
系統管理	<input checked="" type="checkbox"/> Perl 支援	<input type="checkbox"/> 安全性工具
網站服務	<input type="checkbox"/> iSCSI 儲存裝置客戶端	<input type="checkbox"/> 客戶端管理工具
虛擬化	<input checked="" type="checkbox"/> 主控台網際網路工具	<input type="checkbox"/> 撥號網路支援
語言	<input type="checkbox"/> 備份客戶端	<input type="checkbox"/> 支援 Ruby
負載平衡器	<input type="checkbox"/> 儲存裝置可用性工具	<input checked="" type="checkbox"/> 效能工具
資料庫	<input checked="" type="checkbox"/> 列印客戶端	<input checked="" type="checkbox"/> 智慧卡支援
高可用性	<input checked="" type="checkbox"/> 基礎	<input checked="" type="checkbox"/> 目錄客戶端
高延展性的檔案系統支援	<input type="checkbox"/> 大型檔案系統效能	<input type="checkbox"/> 相容性函式庫
高彈性的儲存裝置	<input type="checkbox"/> 安全性工具	<input checked="" type="checkbox"/> 硬體監控程式工具
	<input type="checkbox"/> 客戶端管理工具	<input checked="" type="checkbox"/> 科學支援
	<input type="checkbox"/> 撥號網路支援	<input checked="" type="checkbox"/> 網路作業工具
	<input type="checkbox"/> 支援 Ruby	<input checked="" type="checkbox"/> 網路檔案系統客戶端
	<input checked="" type="checkbox"/> 效能工具	<input checked="" type="checkbox"/> 除錯工具

伺服器	<input type="checkbox"/> Emacs
基礎系統	<input type="checkbox"/> Tex 支援
應用程式	<input checked="" type="checkbox"/> 圖形設計工具
桌面環境	<input type="checkbox"/> 技術編寫
程式開發	<input checked="" type="checkbox"/> 網路瀏覽器
系統管理	<input checked="" type="checkbox"/> 網際網路應用程式
網站服務	<input checked="" type="checkbox"/> 辦公軟體和生產軟體
虛擬化	
語言	
負載平衡器	
資料庫	
高可用性	
高延展性的檔案系統支援	
高彈性的儲存裝置	


伺服器	<input checked="" type="checkbox"/> KDE 桌面環境
基礎系統	<input checked="" type="checkbox"/> Legacy X Window 系統相容性
應用程式	<input checked="" type="checkbox"/> X 視窗系統
桌面環境	<input checked="" type="checkbox"/> 一般用途的桌面環境
程式開發	<input checked="" type="checkbox"/> 圖形化管理工具
系統管理	<input checked="" type="checkbox"/> 字型
網站服務	<input checked="" type="checkbox"/> 桌面環境
虛擬化	<input checked="" type="checkbox"/> 桌面環境平台
語言	<input checked="" type="checkbox"/> 桌面環境除錯與效能工具
負載平衡器	<input checked="" type="checkbox"/> 輸入法
資料庫	<input checked="" type="checkbox"/> 遠端桌面環境客戶端
高可用性	
高延展性的檔案系統支援	
高彈性的儲存裝置	

伺服器	<input type="checkbox"/> Eclipse
基礎系統	<input type="checkbox"/> 伺服器平台開發
應用程式	<input type="checkbox"/> 桌面環境平台開發
桌面環境	<input type="checkbox"/> 開發工具
程式開發	<input type="checkbox"/> 額外開發
系統管理	
網站服務	
虛擬化	
語言	
負載平衡器	
資料庫	
高可用性	
高延展性的檔案系統支援	
高彈性的儲存裝置	





伺服器	<input type="checkbox"/> SNMP 支援
基礎系統	<input type="checkbox"/> 基於網站的企業管理
應用程式	<input checked="" type="checkbox"/> 系統管理
桌面環境	<input type="checkbox"/> 系統管理 Messaging 伺服器支援
程式開發	<input type="checkbox"/> 通訊客戶端支援
系統管理	
網站服務	
虛擬化	
語言	
負載平衡器	
資料庫	
高可用性	
高延展性的檔案系統支援	
高彈性的儲存裝置	

伺服器	<input checked="" type="checkbox"/> PHP 支援
基礎系統	<input type="checkbox"/> TurboGears 應用程式框架
應用程式	<input type="checkbox"/> 網站 Servlet 引擎
桌面環境	<input checked="" type="checkbox"/> 網頁伺服器
程式開發	
系統管理	
網站服務	
虛擬化	
語言	
負載平衡器	
資料庫	
高可用性	
高延展性的檔案系統支援	
高彈性的儲存裝置	

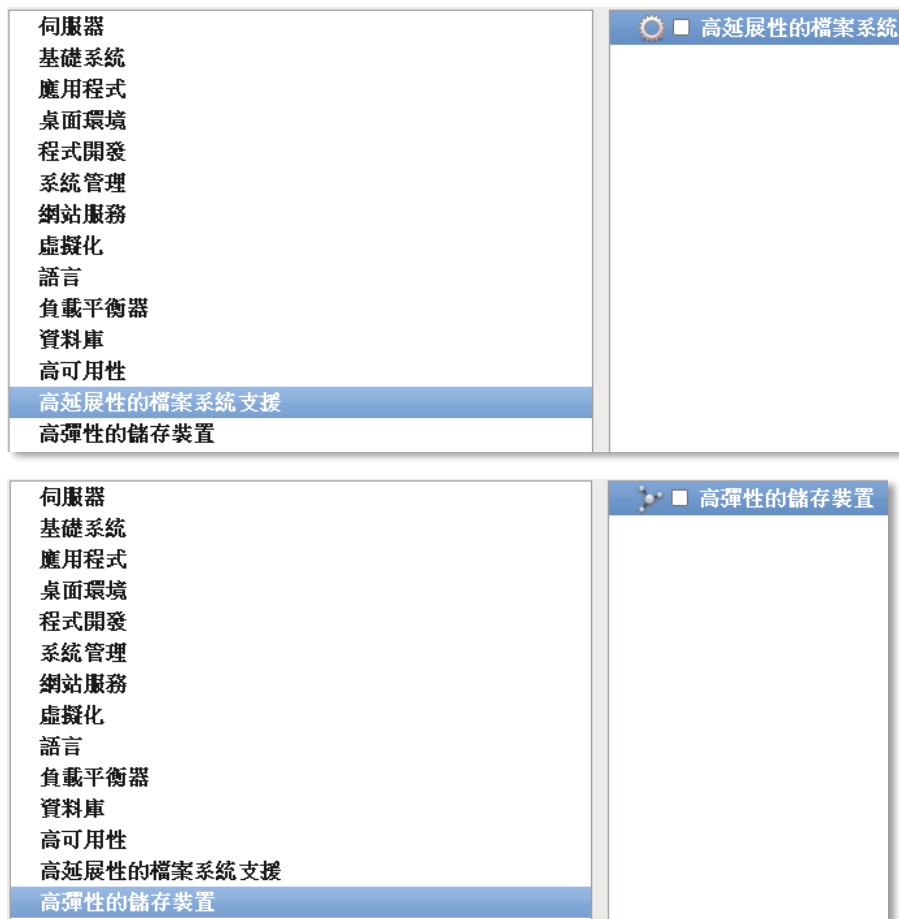
伺服器	<input type="checkbox"/> 虛擬化
基礎系統	<input type="checkbox"/> 虛擬化客戶端
應用程式	<input type="checkbox"/> 虛擬化工具
桌面環境	<input type="checkbox"/> 虛擬化平台
程式開發	
系統管理	
網站服務	
虛擬化	
語言	
負載平衡器	
資料庫	
高可用性	
高延展性的檔案系統支援	
高彈性的儲存裝置	

伺服器 基礎系統 應用程式 桌面環境 程式開發 系統管理 網站服務 虛擬化 <b>語言</b> 負載平衡器 資料庫 高可用性 高延展性的檔案系統支援 高彈性的儲存裝置	<div>  <input type="checkbox"/> 不丹語支援         </div> <div>  <input type="checkbox"/> 不列塔尼語支援         </div> <div>  <input type="checkbox"/> 世界語支援         </div> <div>  <input checked="" type="checkbox"/> 中文支援         </div> <div>  <input type="checkbox"/> 丹麥語支援         </div> <div>  <input type="checkbox"/> 亞塞拜然語支援         </div> <div>  <input type="checkbox"/> 亞美尼亞語支援         </div> <div>  <input type="checkbox"/> 低地德語支援         </div> <div>  <input type="checkbox"/> 俄語支援         </div> <div>  <input type="checkbox"/> 保加利亞語支援         </div> <div>  <input type="checkbox"/> 信德語         </div> <div>  <input type="checkbox"/> 僧伽羅語支援         </div> <div>  <input type="checkbox"/> 克什米爾語支援         </div>
--	--

伺服器 基礎系統 應用程式 桌面環境 程式開發 系統管理 網站服務 虛擬化 語言 <b>負載平衡器</b> 資料庫 高可用性 高延展性的檔案系統支援 高彈性的儲存裝置	<div>  <input type="checkbox"/> 負載平衡器         </div>
--	---

伺服器 基礎系統 應用程式 桌面環境 程式開發 系統管理 網站服務 虛擬化 語言 負載平衡器 <b>資料庫</b> 高可用性 高延展性的檔案系統支援 高彈性的儲存裝置	<div>  <input checked="" type="checkbox"/> <b>MySQL 資料庫伺服器</b> </div> <div>  <input checked="" type="checkbox"/> <b>MySQL 資料庫客戶端</b> </div> <div>  <input type="checkbox"/> PostgreSQL 資料庫伺服器         </div> <div>  <input type="checkbox"/> PostgreSQL 資料庫客戶端         </div>
--	---

伺服器 基礎系統 應用程式 桌面環境 程式開發 系統管理 網站服務 虛擬化 語言 負載平衡器 資料庫 <b>高可用性</b> 高延展性的檔案系統支援 高彈性的儲存裝置	<div>  <input type="checkbox"/> 高可用性         </div> <div>  <input type="checkbox"/> 高可用性管理         </div>
--	---



37. 過了這關之後，接下來會看到系統在檢查所有套件的相依性。



38. 等須安裝的套件計算完成後，就是套件安裝。這個步驟在整個安裝過程中最花時間，您可以利用這個空檔上上廁所，泡泡咖啡，或是參考下一個建議。
39. 系統會將安裝進度告訴您的，您可以根據數字來判斷剩餘的時間。如果機器慢，您去買份便當、炸雞或咖啡回來！！

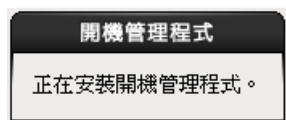




40. 若您是利用安裝光碟安裝的話，系統會提示您該換光碟了，這時您可以將安裝光碟的第二片換裝至光碟機內，然後按下「下一步(N)」即可繼續安裝。安裝至最後，您會看到套件安裝完成後的處理畫面。

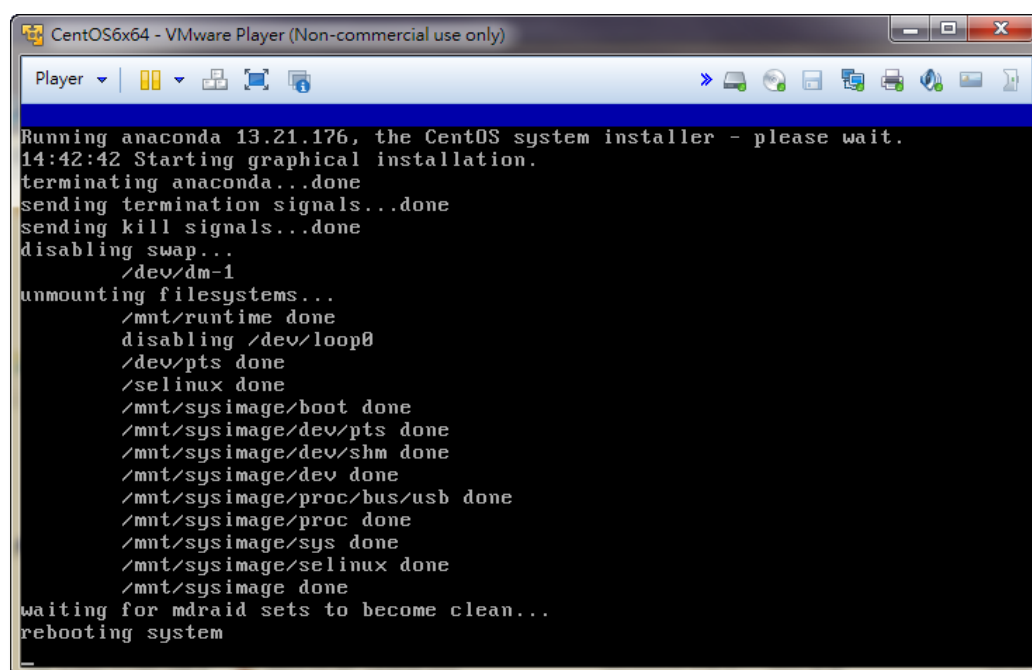
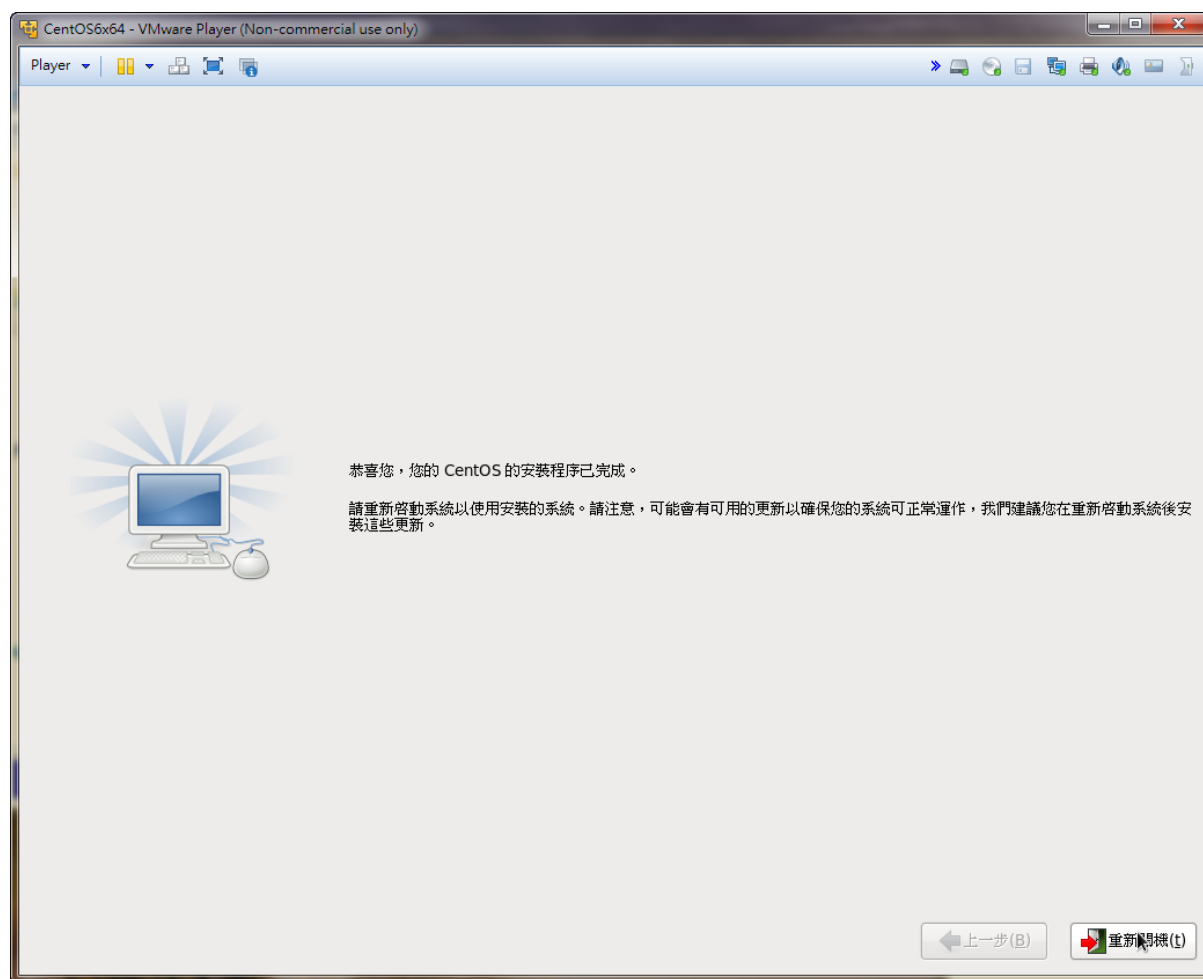


41. 套件安裝完成後，最後系統會自動安裝「開機管理程式」(Bootloader)，讓您下次直接用硬碟開機。

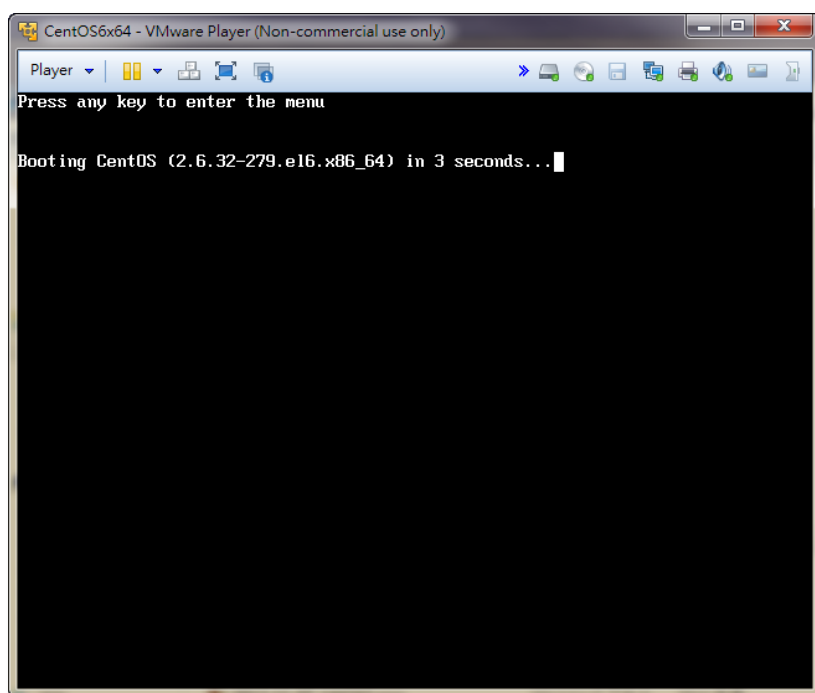




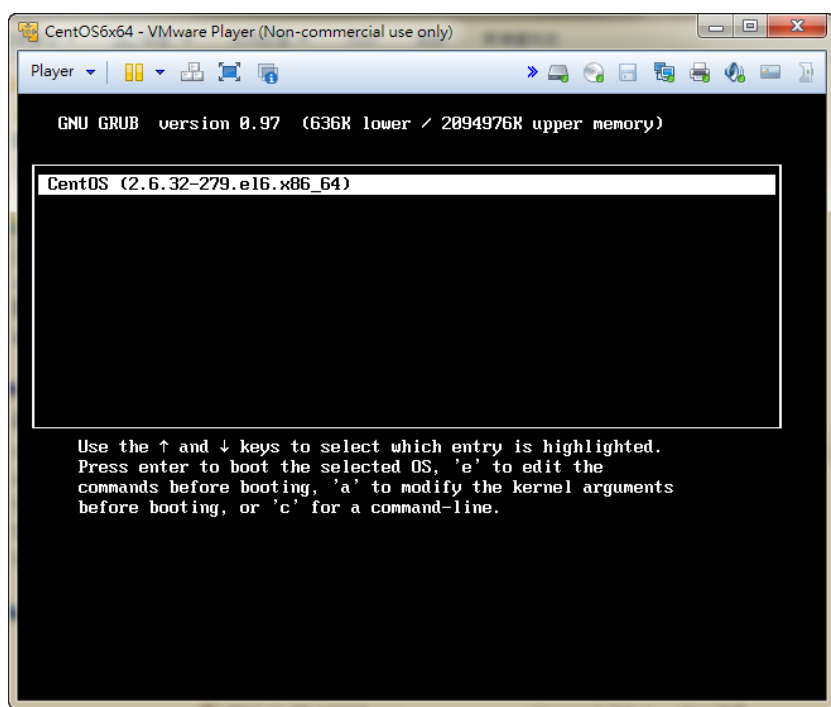
42. 到最後、最最後、最最最後，您應該會看到如下這個畫面：Complete! 真的要恭喜您啦~~!! 您剩下的工作是按「重新開機(t)」然後等機器重新開機(如果您的 CD 不能自動退出來，那請您手工將它取出來，或修改 BIOS 不要用 CDROM 做 First boot)。



43. 若您以為到這裡已經安裝完成了，那就錯了。在機器重新開機後，系統第一次開機，還會要求設定一些如：系統認證方式、防火牆、系統服務等設定。
44. 讓我們來迎接並體驗第一次系統開機的程序。在您看到「Press any key to enter the menu」，並會看到目前系統打算在幾秒鐘之後，開啟那個作業系統。

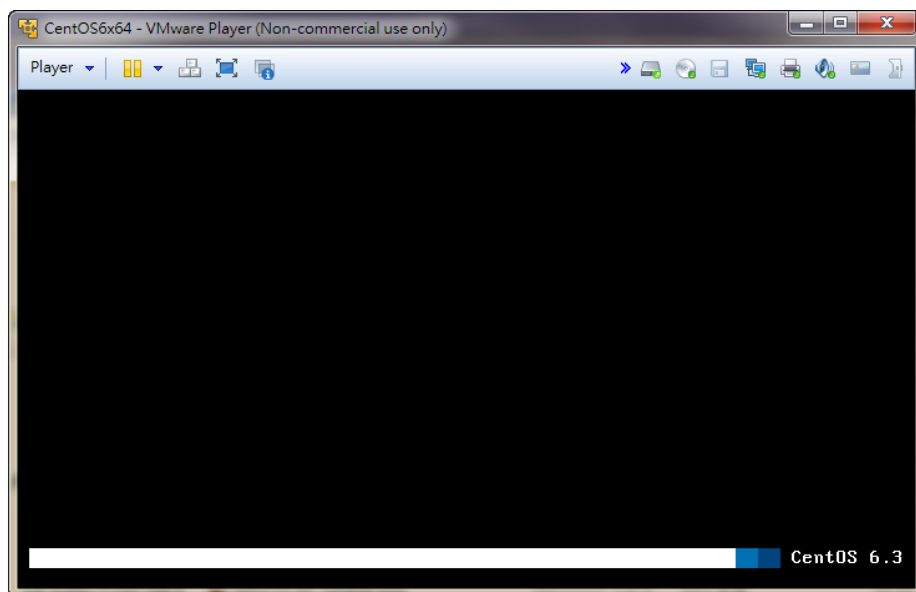


在這個倒數的時間內，您可按下任何一個按鍵，就可進入開機選單了。您可在開機選單中，移動上下方向鍵選擇您要進入的系統就可以了。



45. 若要手動去選擇開機選單，則請在倒數完成之前按下鍵盤任意按鍵，並選取您想要開機的項目後，按下 Enter 即可進行開機程序。

46. 接下來就進入開機的程序畫面：



47. 第一次開機時，因為系統還有一些環境設定要先設定；所以，在安裝完 OS 後第一次開機，會自動進入系統設定畫面。系統會要求您先設定系統中重要的設定值，您必須依設定程序，一步一步地將 OS 設定好。其可設定的項目如下：

歡迎：系統歡迎頁面。

授權資訊：該作業系統的授權與使用的相關訊息。

建立用戶：建立系統的使用者。

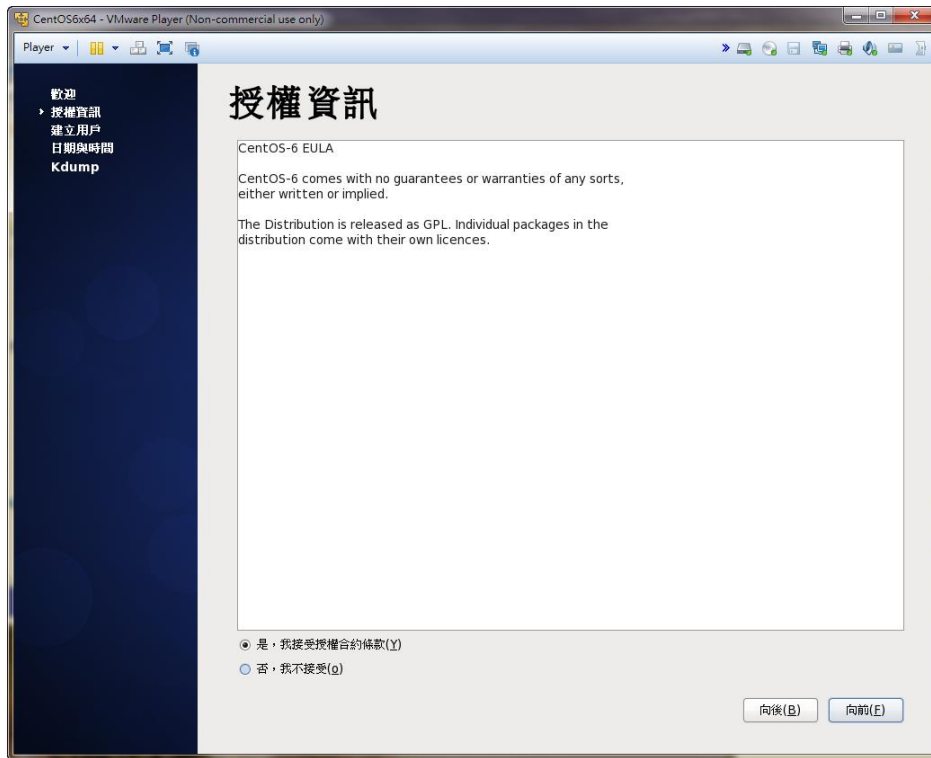
日期與時間：設定這個系統的日期與時間，同時也可以利用網路校時服務。

Kdump：系統核心錯誤時，記錄當時之各項記錄與狀態。

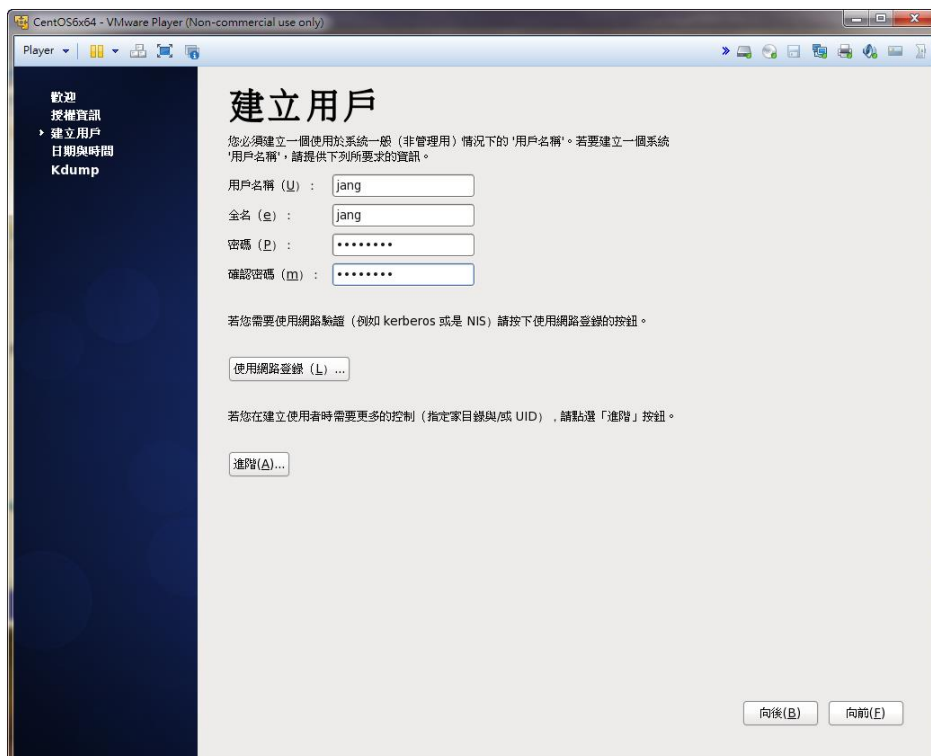


您可以按「向前(F)」及「向後(B)」兩個按鈕，依序設定必須設定的項目。

48. 授權資訊，主要顯示 CentOS6 的使用授權條款，您必須同意後，才能使用該系統。  
請勾選「是，我接受授權合約條款(Y)」後，按「向前(F)」進入下個程序。



49. 接下來設定系統的使用者相關資訊，一般而言，就是建使用帳號。



您須輸入的欄位有：

用戶名稱(U)：就是您在這個系統上使用的帳號

全名(e)：可以填入帳號或是英文全名

密碼(P)：使用這個系統的密碼，建議超過 8 個字元

確認密碼(m)：再次重覆輸入剛剛那組密碼，以確認您不是亂打的



**建立用戶**

您必須建立一個使用於系統一般（非管理用）情況下的 '用戶名稱'。若要建立一個系統 '用戶名稱'，請提供下列所要求的資訊。

用戶名稱 (U) :

全名 (e) :

密碼 (P) :

確認密碼 (m) :

若您需要使用網路驗證（例如 kerberos 或是 NIS）請按下使用網路登錄的按鈕。

若您在建立使用者時需要更多的控制（指定家目錄與/或 UID），請點選「進階」按鈕。

若您的使用者認證方式，不是本機認證的話，請按下「使用網路登錄(L)」鈕進行設定。CentOS6 提供網路的認證方式有 LDAP、IPAv2、NIS、Winbind 等四種。而密碼雜湊演算法最高提共至 SHA512。



**認證設定**

身份和認證(I) 進階選項(O)

**使用者帳號設定**

使用者帳號資料庫：(U)

LDAP  
IPAv2  
NIS  
Winbind

**認證設定**

認證方式：(H)

**本機認證選項**

☒ 啟用指紋掃描器支援

☐ 啟用本機存取控制(I)

提示：這是透過了 /

密碼雜湊演算法：(P)

**其它認證選項**

☐ 第一次進行登錄時建立家目錄(H)

**智慧卡認證選項**

☐ 啟用智慧卡支援(S)

提示：智慧卡支援登入本機與中央管理的帳號。

若還有要輸入多人使用的帳號與密碼，請您按下「進階(A)」功能，進入使用者管理介面，進行使用者的管理設定。



50. 再來設定系統的日期與時間：您可透過日期與時間的調整，讓系統上的時間與標準時間一致；若這台主機可以上網的話，建議透過網路校時的機制，自動讓系統去校正系統時間。



若需進行網路校時，請勾選「透過網路同步日期與時間(y)」選項後，在下方 NTP 伺服器欄內的三筆時間主機先刪除(因為遠在國外，校時速度太慢了!)，再按「加入(A)」鈕新增 2 筆市網提供的時間主機：163.28.136.2 及 163.28.136.10。



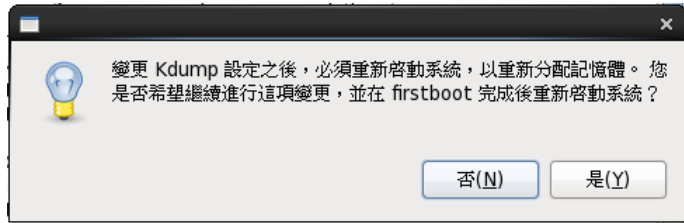
完成後按「向前(F)」以便進行網路校時。



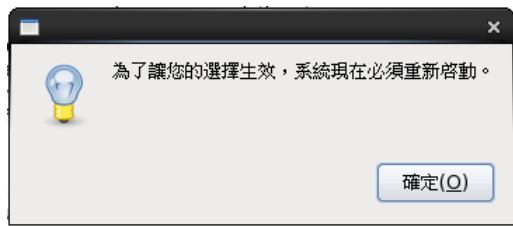
51. 再來是設定 Kdump，依系統的預設值就可以了。最後按下「完成(F)」鈕即可。



按下後，系統會提醒完成後會重新啟動系統。



再度確認系統要重新啟動。按下確定後，就真的完成了第一次開機設定的動作了。這時候要有點耐心，因為畫面會停止不動近 1 分鐘，然後系統會自動重新開機。



52. 到這裡，關於 Linux 的安裝就大功告成了。呼~~~ 不禁一聲長嘯：好不容易才殺出重圍啊！回首這條血路，放眼僅是屍骸、盔甲、血流成河！哈哈，太誇張了啦~~！等您多裝幾遍，有了心得，其實您會發現 Linux 的安裝一點也不難，搞不好，興之所至，或會專門找兩台機器放一起，一台裝 Linux，另一台裝 MS Windows，比比看誰裝得快呢~~~ 我保證(嘻~~ 又來了~~)，在同一台機器上，裝 Linux 絕對比裝 Windows 要快：光重新開機的次數就少了一半或更多。不信？試試看嘛~~~~ ^\_^



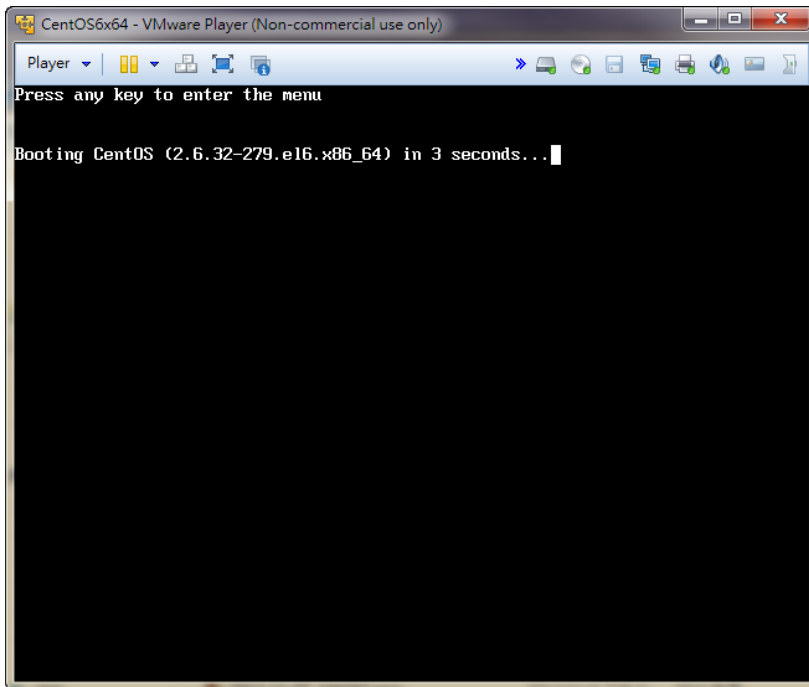
#### 四、開機與關機：

開機和關機誰不會哦？哈哈，不一定哦~~~

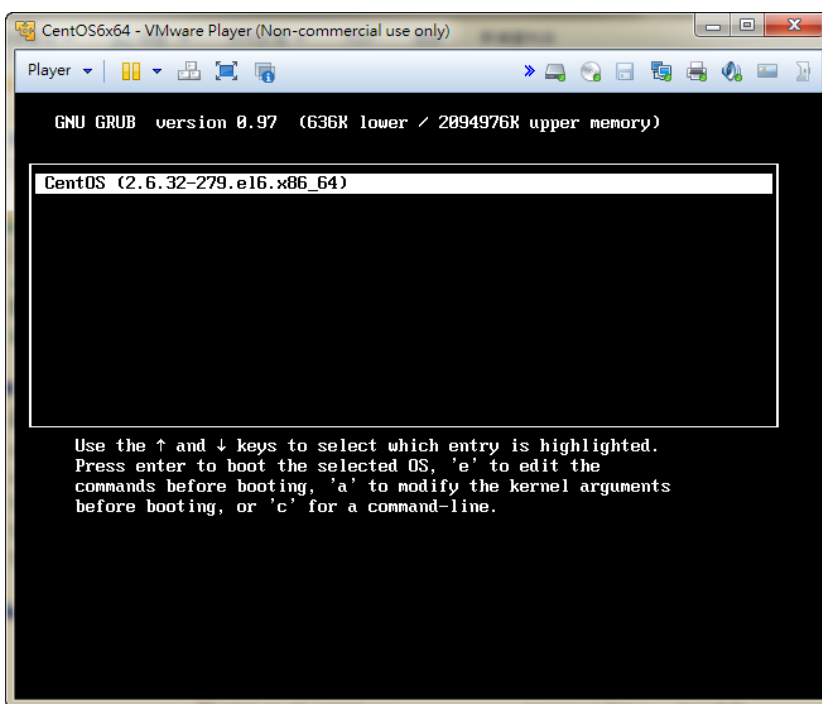
##### 開機

您或許會發現安裝完 Linux 之後，原來的 Windows 開機畫面不見了！那~~ 怎麼回去啊？

不知道不緊要啦。當機器開著之後，在您看到「Press any key to enter the menu」，並會看到目前系統打算在幾秒鐘之後，開啟那個作業系統。



在這個倒數的時間內，您可按下任何一個按鍵，就可進入開機選單了。



您可在開機選單中，移動上下方向鍵選擇您要進入的系統就可以了；若您的 CentOS6 是安裝在原有 Windows 系統的機器，您便可看到 CentOS6 自動將您的機器，做成雙系統的機器了。若您要開啟原來的 Windows 系統，只要您移動上下方向鍵選擇 Windows 系，按下 Enter 即可進入原來的 Windows 系統。

Centos 6 的開機管理程式是用 GRUB，它是 Linux 安裝在機器上面的系統載入程式。根據我們在安裝的時候所選擇，目前 GRUB 安裝在系統的 MBR (Master Boot Record)，也就是開機磁碟的第一個 Cylinder 之第一個 Sector 之上。MBR 所存放的 loader，是用來告訴機器在啟動之後要載入那個系統，和到哪裡載入。除了 GRUB 之外，其實我們也有許多其它 loader 可以選擇，例如 NT 的 NT Loader、System Commander、Boot Magic、以及國人所寫的 spfdisk、等等。假如您已經採用別的 loader 而不是 GRUB，那您在安裝的時候，就要確定您的 GRUB 是裝在 /boot partition (所謂的 super block) 上面而不是 MBR。無論您用哪個 loader，也不管 GRUB 裝在 MBR 還是 super block，在一些新的大容量硬碟上面(通常超過 8G)，要注意 /boot 分區所在位置不要超過 1024 磁柱的範圍。不過，新的 Linux 啟動程式都允許超出這個限制，只需確定磁碟使用 LBA 模式，以及在 GRUB 上使用 lba32 選項就可以了(這個以後我們在談到 GRUB 的設定時再說明)。

有些朋友，在碰到多重系統安裝的時候，會覺得有些疑惑，或許只是對 OS loader 不甚了解而已，單純將安裝順序換成：先 windows 後 linux 就可以了。

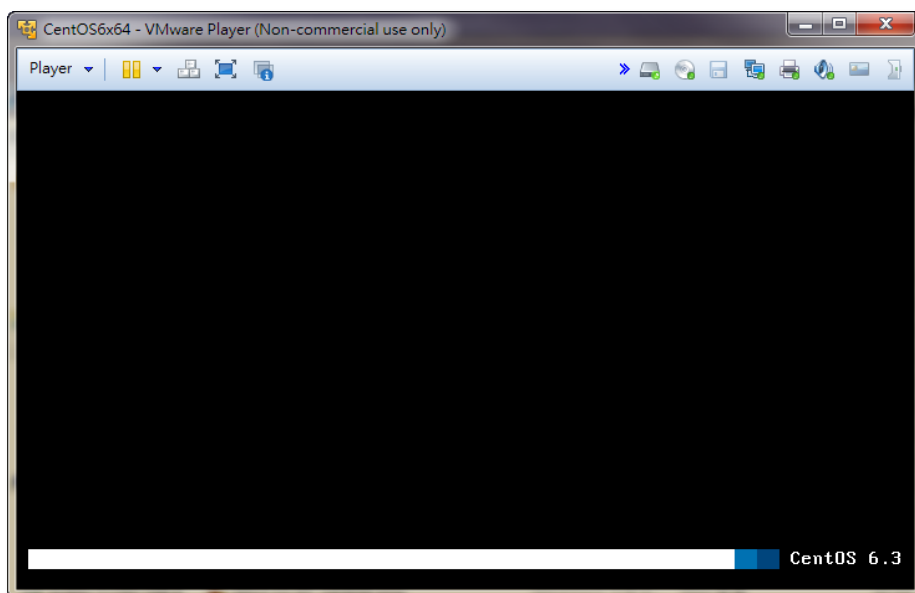
為什麼？那要知道各 OS 放在 MBR 裡面的 loader 能力了。一般的 OS loader，都會提供開機選單，讓您選擇用什麼系統開機，以及開機系統放在硬碟的什麼位置上。比如，linux 的 GRUB 就有這個功能：它除了開 linux 之外，還提供一個選單，給您開 Windows。但 windows 的 loader 卻不然，它的 loader 只能開 windows 而不讓您有選單開 linux。(真是不公平哦~~)

而 OS loader 一般都放在 MBR 位置上，也就是第一個 cylinder 的第一個 sector 上面，這樣，系統順利通過 BIOS 之後，就會到這裡讀入硬碟的開機資訊，也就是載入 loader。在 MBR 上面，每次只能放一個 loader，如果您先放 windows 的 loader，然後放 linux 的，MBR 的 loader 就變成是 GRUB，那您就有一個選單來選擇開機系統。如果先灌 linux 再 windows，那 MBR 的 loader 是 windows 的，因而沒有選單選擇，只能進入 windows。

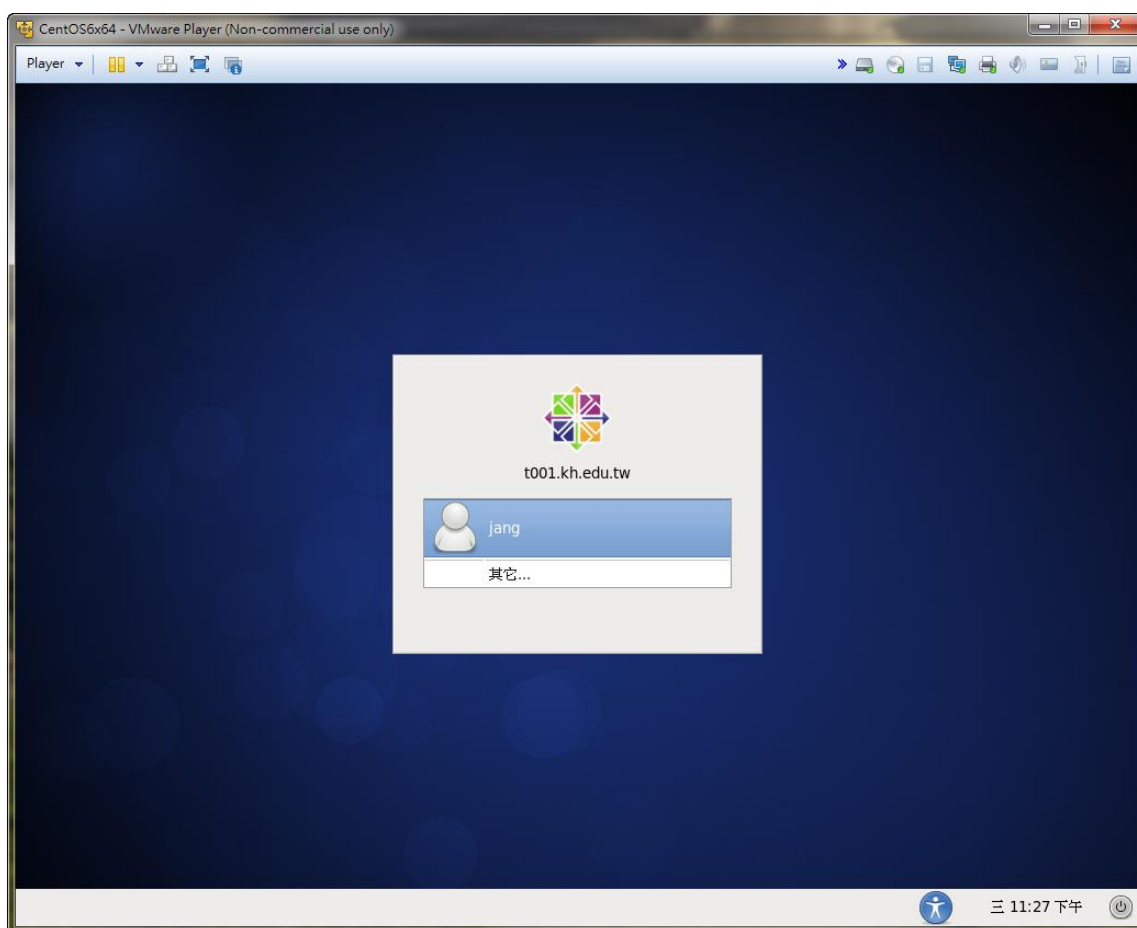
不過，Linux 的 GRUB，除了放在 MBR 之外，還可以放在 boot 的 partition 上面的第一個 block，也就是所謂的 superblock 是也。那您在安裝的時候，就要將 GRUB 裝在 boot 的 partition 上面；或是寫入時指定至 hd0,1 (假設您的 boot partition 是在 hda2 上面)，然後再執行 setup (hd0,1) 命令就可以完成。

假如您真的將 GRUB 裝在 superblock 上面，那您在 MBR 上面的 loader 必須有能力提供選單，而且能夠告訴系統到哪裡找到 GRUB。抱歉，windows 的 loader 都沒有這個能力。其它諸如 partition magic 或 spfdisk 之類的 loader，則有這個功能，但前提條件是：您的 GRUB 要裝在 superblock 上面。

當您在 GRUB 選擇 Linux 之後，您會看到一大堆開機過程的報告，有興趣可以逐行看看(假如您眼睛夠快的話)。如果開機過程卡在某些服務上面而等上好幾分鐘的話，可能您的網路設定不正確，最常見的問題是 /etc/hosts 或 DNS 沒設定好。為避免這些情況，請確定機器在開機的時候，啟動程式能夠連得上指定的伺服器地址。

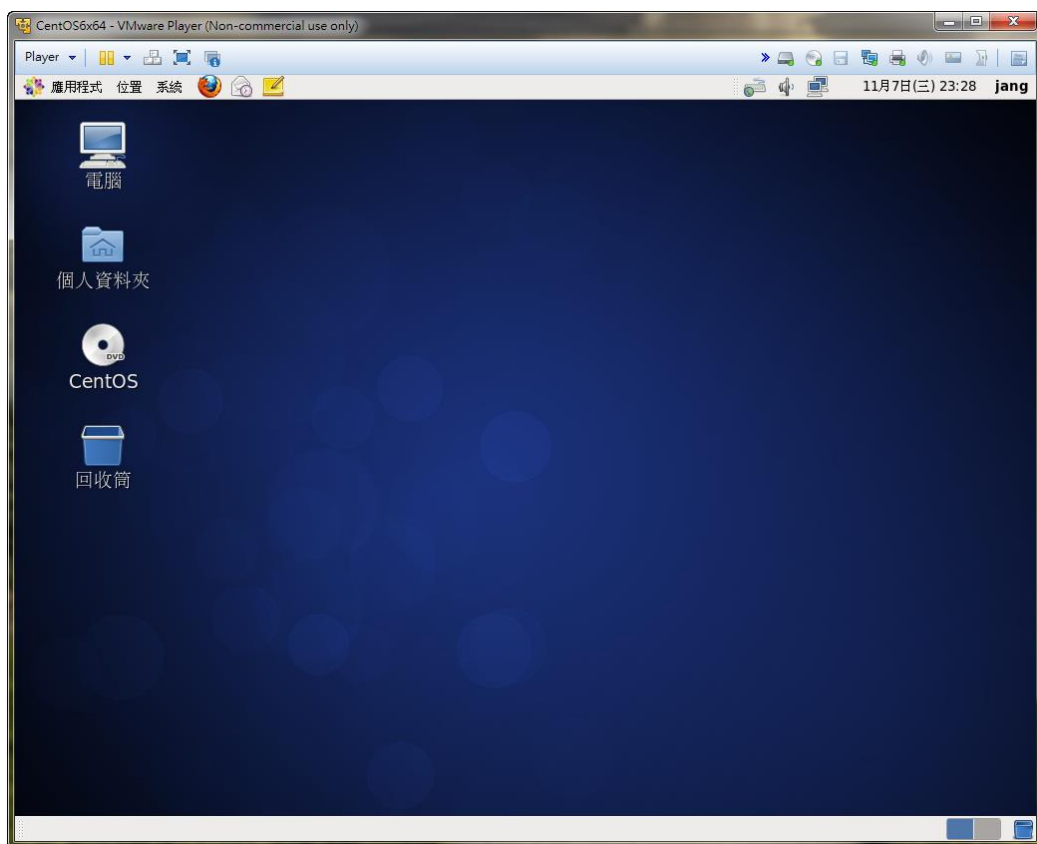
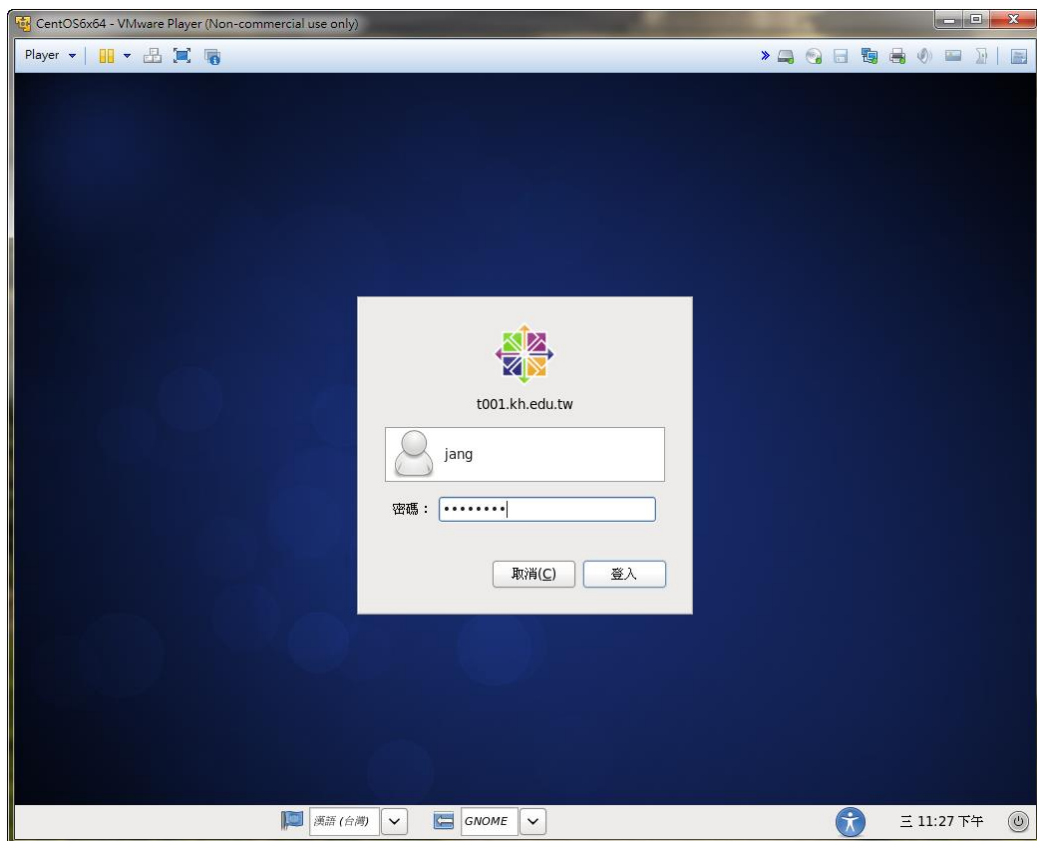


最終，您會看到 Login: 的提示，如果您的 X window 在安裝的時候設定好了，那麼登錄畫面會是圖形界面的。這時候您可以用 'root' 進行登錄，然後輸入您在安裝時設定的 root 密碼。之後，整個 Linux 系統就是您的啦！



不過，任何時候，我們都不鼓勵直接用 root 登錄，因為它的權限太大，會在系統安全上構成威脅。我們通常是用普通用戶進行登錄(也就是安裝的時候所設定的帳號)，然後真的碰到系統設定的時候，才用 su 或 sudo 命令切換為 root 身份執行。不過，這裡的例子，大部份

是教您設定伺服器的，通常都需要 root 來執行，只是日後換到真正的工作系統之後，就請盡量避免用 root 來工作了。



至於機器如何從啟動到成功載入，如果我們對這個過程了解的話，相信在日後的管理，或遇到問題的時候會非常有幫助。下面不妨讓我們看看 Linux 的開機過程和其中的一些選項。

### ***Init & Run Level***

系統的初始化步驟粗略如下：

1. 首先，當電源打開之後，機器會首先經過 POST 檢查，然後從 CMOS 載入 BIOS，以確定最基本的硬體架構，如時鐘和硬碟資訊等。
2. 然後會找到 MBR，載入 boot loader，以確定啟動哪一個作業系統。
3. 當作業系統選定之後，LOADER 會告訴機器到那裡載入作業系統的核心程式。
4. 當核心程式載入之後，機器基本上就交由核心處理了。核心程式首先要載入檔案系統，然後第一個執行的程式就是 init (它是所有程式的 parent)，init 的第一個動作是找到 /etc/rc.d/rc.sysinit 並執行之。
5. 這個 rc.sysinit 程式首先是執行 /sbin/initlog 並輸出 \$LOGLEVEL 變數，以決定記錄檔的信息量。
6. 接著檢查 /dev/.devfsd 和 /sbin/devfsd 這兩個檔案，以決定系統是否使用 devfs。
7. 然後設定 PATH 環境。
8. 然後檢查 /etc/sysconfig/network 檔是否存在，然則根據設定檔設定網路環境，否則用 localhost 命名系統。
9. 再來是執行 /etc/rc.d/init.d/functions 這隻 script，以提供其後的 shell 程式所使用的一些變數和功能。
10. 然後載入 keymap，按照您安裝時指定的鍵盤資訊來確定您的鍵盤類型。
11. 執行 /sbin/setsysfont 來設定系統使用的字型。
12. 啟動系統的置換(swap)空間。至此，系統之主機名稱(hostname)和網域名稱(domain)，將作為環境變數的依據。
13. 然後 rc.sysinit 會檢查 /fsckoptions 這個檔是否存在，然則，會改變這個檔的中的一個變數值；如果 /forcefsck 檔案存在的話，就會在這個變數值前面增加一個旗標，這樣會導致後面的步驟強迫系統對檔案系統進行檢查。
14. 接下來，會檢查 /fastboot 是否存在，然則，系統會略過檔案系統的檢查步驟；否則，系統會對 / (也就是 root) 進行一個完整的檢查。
15. 之後，系統會執行 mount -t proc /proc /proc 命令，為核心掛載一個特殊的檔案系統，它完全建立在記憶體之上。每次關機都會消失，而每次開機才被建立。其中的資料，純粹是給核心使用的，其中一些設定，會根據系統設定檔進行配置。
16. 完成這個動作之後，接下來是對 Plug and Play 設備的檢測。如果經過特別設定，還會執行 isapnp 程式來檢查 ISA 類型的 PnP 設備。
17. 現在，檔案系統的 / 才掛載為完整的讀和寫模式，並更新 quota 資訊，同時在 / 上面實施 quota 限制。
18. 再來呢，系統會清掉 IN\_INITLOG 變數，如果 /etc/HOSTNAME 檔不存在，則將 HOSTNAME 的變數值作為該檔的內容。

19. 再來是清空 `mtab`，並將 `/` 檔案系統和 `/proc` 置於其中。系統通常根據 `/etc/fstab` 來判定要掛載的檔案系統在哪裡，而 `mtab` 是一個檔案，算是從 `/etc/fstab` 提煉出來真正給系統使用的版本吧。
20. 然後，系統會載入一系列的核心模組。一旦完成，除非前面的 `/fastboot` 存在，否則系統會對 `/` 之外的所有檔案系統進行檢查。除了 NFS 之外，所有分區都會完成掛載，並且非 `/` 分區的 `quota` 也會打開。
21. 到這裡，系統會砍掉一些檔案，以清理這次啟動，同時也是為下次啟動作準備。前面提到的 `/fastboot` 和檔案系統檢查選項都會清掉，同時，`/var/lock` 其下的一些 `lock` 檔也會移除，`/var/run` 裡面的陳舊行程 ID 也會砍掉，更多在 `/tmp` 裡面的 `lock` 檔也會砍光。這樣，`rc.sysinit` 可以說為系統安排好一切了：設定好系統時鐘、啟用置換空間、載入相應模組、並根據 `/etc/sysconfig/desktop` 來決定 X 系統使用的視窗管理系統(除非使用者另行設定)。
22. 關於 `rc.sysinit` 在啟動過程中所顯示的所有信息，都會存放到 `/var/log/dmesg` 裡面去。
23. 此時，`rc.sysinit` 已經完成準備工作，並將系統交還 `/sbin/init` 程式，然後根據 `/etc/inittab` 選擇預設的 `run level` (這個容後解釋)，以決定執行哪些服務程式。具體要執行的程式，按照 `/etc/rc.d/rcX.d/` (`X` 代表不同的 `run level` 號碼)目錄裡面的 `script` 內容而定。而最後一個要執行的 `script`，則是 `/etc/rc.d/rc.local`。並執行 `/sbin/loglevel` 以確定系統的記錄等級(級別越高，記錄越詳細)。  
在 `/etc/rc.d/` 目錄下的 `rc.local`，有點像 DOS 裡面的 `autoexec.bat`，如果您有哪些在開機的時候需要執行的系統設定，可以將命令行，或 `script` 寫在這裡。比方說，您在用別台機器 `telnet` 進來的時候，會看到機器目前的作業系統資訊(這或許對別有用心的人非常有用)，您單純以為修改 `/etc/issue.net` 的內容就可以解決掉；但發現重新開機之後，檔案又復原了。呵~~ 這就是 `rc.local` 搞的鬼啦：當您修改好 `issue.net` 之後，還不要忘记修改 `rc.local` 這個檔，將其中一行 `cp -f /etc/issue /etc/issue.net` 註解起來。
24. 最後，`init` 將系統交由 `login` 程式接管，正式完成開機使命。
25. 假如您對上面的描述不是很理解，不用擔心，儘可略過您不懂的對方，等日後對系統有更充份的了解之後，再回來重拾靈感好了。不過，對開機過程的了解程度，會影像到您日後的系統管理和調整技巧，尤其是在進行除錯的時候。建議您最好將開機過程了解透切一些，是絕對有百利而無一害的。

好了，前面提到過 `run level` 這個詞，究竟是什麼東東呢？嗯，如果您以前玩過 DOS，為不同遊戲設定過不同的開機環境，或許比較容易了解。要不然，您可以簡單的認為：`run level` 就是不同的系統啟動配置，不同的 `run level` 會讓系統執行不同的服務和設定，最後獲得的環境和功能也不一樣。系統每次開機，只能進入其中一個 `run level` 而已，您可以在 LILO 提示的時候指定要進入哪個 `level` (或是在啟動時看到紅色的 RedHat 文字時按 `i` 鍵)，也可以調整 `/etc/inittab` 的設定來指定系統的預設 `level` (號碼)。不同的廠商，其 `run level` 的號碼不太一樣，如果以 RedHat 為例，號碼所代表的 `run level` 如下：

- Level 0: 關機
- Level 1: 單人模式
- Level 2: 多人模式，沒有網路功能

- Level 3: 完整多人模式，文字界面
- Level 4: 保留
- Level 5: 完整多人模式，圖形界面
- Level 6: 重新開機

以上這些號碼，分別對應不同的系統配置。在 `/etc/inittab` 檔案中，您會找到一行句子如：  
`id:5:initdefault:`

其中的數字 5 所代表的就是系統預設的 `run level`，這裡就是圖形界面，當您的系統完成啟動之後，您會看到 `X window` 的登錄畫面。如果您要將預設界面換成文字模式，那只要將 5 換為 3 就可以，這樣下次開機就會進入文字界面；當然，如果您換成 0 的話，那每次啟動完畢就會關機，這可不好玩哦~~~

有時候，因為系統的設定錯誤，或是檔案損壞，您可能必須進入單人模式進行一些維護工作。比方說，您的 `root` 密碼忘了，那您可以到 `LILO` 提示那裡(開機選單按 `Ctl+X`)，然後輸入：**`linux -s`**

若您是使用 `GRUB` 開機管理系統的話，會麻煩一點點；首先在開機的畫面，選定要開啟的 `Linux` 選項，按下「e」鍵進入編輯畫面，再將游標移到 `kernel` 那行，再按一次「e」鍵，在 `kernel` 這行的最後面加上 `single` 後，再按下 `Enter` 鍵確定後，再按「b」就可以進入單人模式了。

這樣啟動完成後就會進入 `run level 1`，這裡您是無須登錄就具備 `root` 權限的了，然後您輸入 `passwd root` 就可以重新設定 `root` 的密碼了。不過，這個便利也變成一個系統的安全問題。此時，如何確保您的系統不被外人實體上接觸到就顯得非常重要了，否則，別人只要重新啟動機器就可以為所欲為了。

## 關機

除非您不想機器不用的時候還開著(如為了省電)，或是進行硬體更換，`Linux` 甚少需要關機的(有些 `Linux` 跑好幾年還好好的呢!)。不過，您知道怎樣關機嗎?

如果您在安裝過程中能夠正確設定好 `XWindow`，並且選擇了開機進入 `X`，或許您還能找到關機的門徑(和 `windows` 和像)。不過，通常來說，除非是天才，在第一次使用 `Linux` 的時候(之前也沒 `Unix` 或 `Novell` 經驗)，還真不知道怎麼關機哦!(哈哈，指我自己啦~~~ ^\_^ 見笑見笑!)

不過，各位根據剛纔的 `run level` 和 `init` 命令，或許也不難找出關機的辦法吧? 輸入 `init 0` 就可以了! 不過，這其實不是一個正確的關機辦法，如果您輸入下面這個指令：`shutdown --help`

您會看到好些選項，請看看各項的意思。如果您要立即將機器關掉的話，建議您輸入：  
`shutdown -h now`

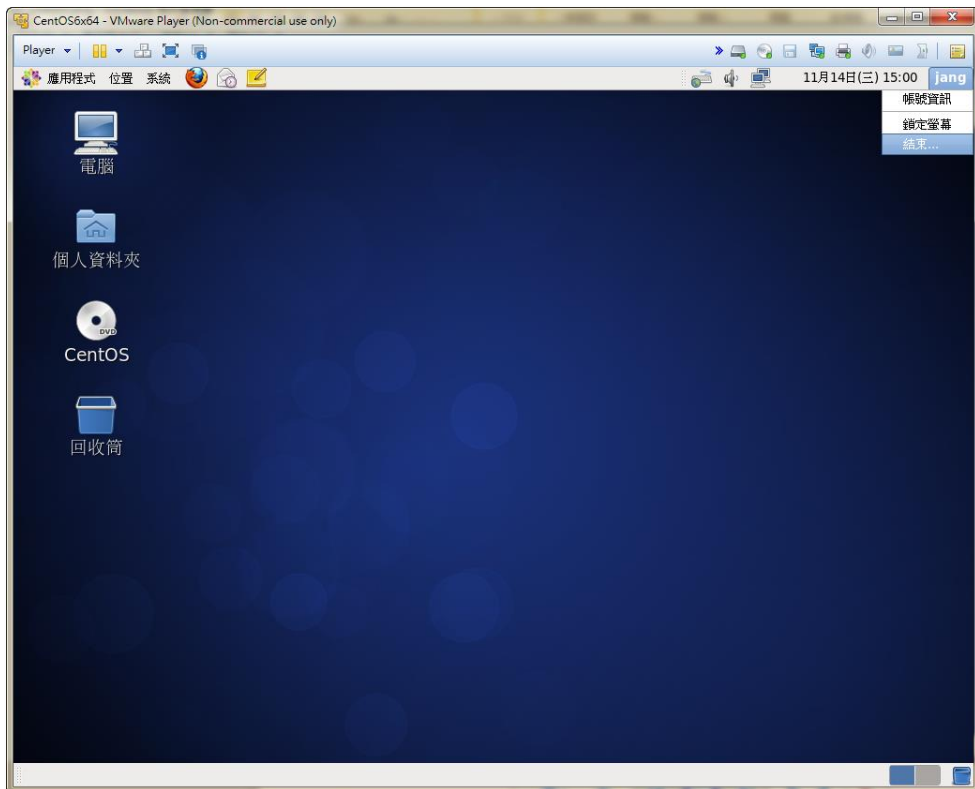
如果用 `r` 代替 `h` 的話，會進行重新啟動，不過，您可以用 `reboot` 這個指令代替，或可以按 `ctl+alt+del` (這個也不建議使用，更要注意所有資料已回存好和程式都關閉妥當)。

在某些舊系統中，您還必須執行 `sync` 命令多次，以確定所有資料都回存到硬碟中，才可以進行關機的動作。不過，如果您用 `shutdown` 或 `reboot` 命令的話，就不需擔心這個了。

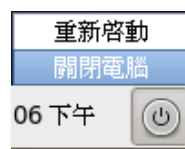
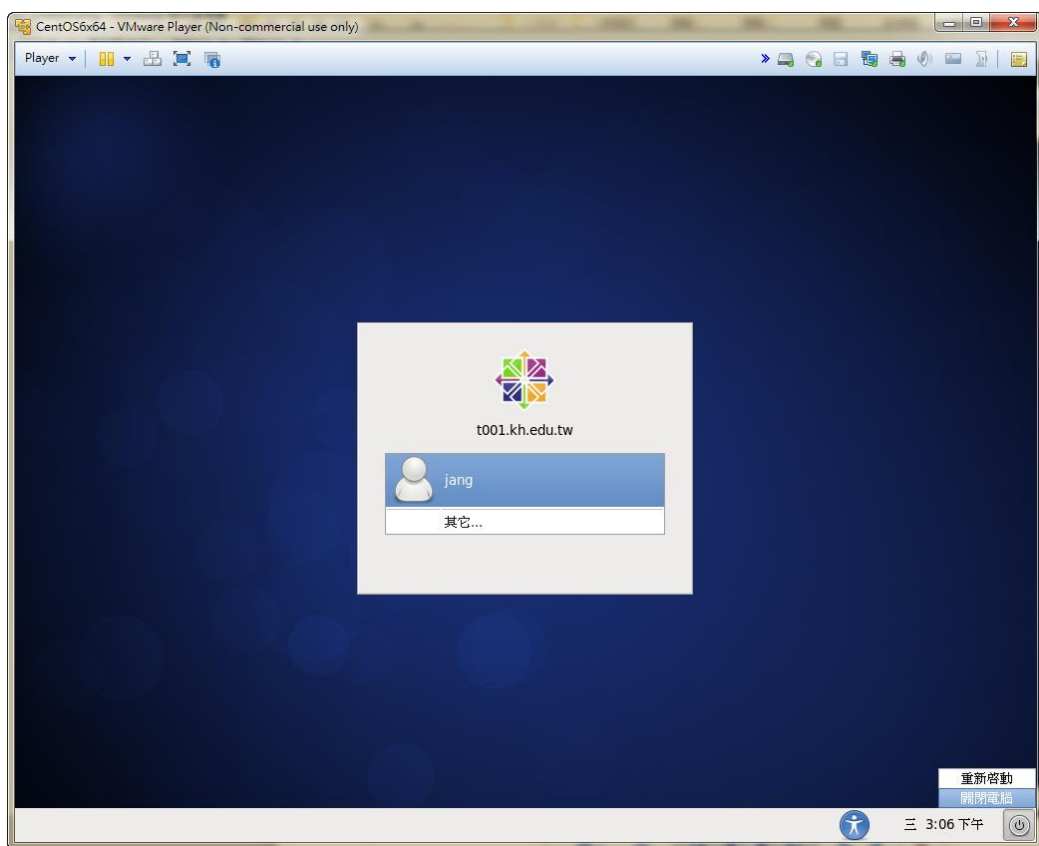
其實，常常遇到有人問說，`shutdown -r` 與 `reboot` 與 `halt` 這些指令的關係。其實 `shutdown` 指令，只是可以設定訊息，時間等，真正 `reboot` 之類的動作，還是呼叫 `reboot` 來

執行的。而 `reboot` 指令則不，將會立即 `reboot`。`halt` 也是一樣，將會立即關機。`reboot` 與 `halt` 預設都會先執行 `sync`，(也可強迫不要用 `sync`，比方 `reboot -n`)這兩個指令無法送出訊息到 `tty..` 等等。

另外還有人問到 `poweroff` 指令，其實 `halt` 指令，預設是會執行 `poweroff` 的，所以會將電源切掉。等您已經會得正常的開機和關機之後，我才可以說：您可以放心進行學習了。







參考資料（資料來源）：

1. <http://www.study-area.net/>
2. <http://www.vbird.org/>
3. <http://people.redhat.com/kwade/fedora-docs/selinux-faq-en/>
4. <http://fedora.redhat.com/docs/selinux-faq-FC4/>
5. <http://josesun.blogspot.com/2004/08/linux-selinux.html>
6. <http://www-900.ibm.com/developerWorks/cn/linux/s-selinux/index.shtml>
7. <http://mirror.centos.org/centos/4/docs/html/rhel-ig-x8664-multi-en-4/>
8. <http://mirror.centos.org/centos/4/docs/html/rhel-isa-en-4/>

# CentOS6 之昇級與補強

高雄市政府教育局  
資訊教育中心  
網路組 張宏明  
更新日期：2012/11/14

## 一、為什麼需要昇級與補強：

1. 世界上還未出現一套完整無缺陷作業系統，所以每一套作業系統，均有其不完整性與缺點，當缺失出現後，就需要新套件來昇級與補強。
2. 當發現套件上有漏洞時，就需要修正套件來昇級與補強，以避免 Server 被人藉著已發現的漏洞，遭受入侵。
3. 所以一個管理良善的 Server，必須每週去注意您所安裝的套件是否查出新漏洞，每月須去檢查是否有更新的套件出現。

## 二、需要用到的工具（命令）：

### 1. lftp：

最主要的是因為各版本的更新套件，均存於 ftp 中，我們需要一個強而有力的 ftp 程式，將該本中之補強與昇級套件，下載到 Server 中。而 ftp client 端的程式，lftp 會比 ftp 命令好用多了；其特點有：自動匿名登入 ftp 站、可利用 tab 鍵輸入又臭又長的檔名、自動判別檔案的屬性(binary 或 ascii)、自動比對已存在的檔案是否更新、自動續傳...等功能。

### 2. rpm：(redhat package manager)

為 redhat 系統的各套件之管理命令，其詳細的用法可由 man rpm 或 rpm --help 獲得，在此會用到 rpm --Fvh 及 rpm --Uvh。

-F 會先檢查系統上是否已經該安裝該套件，若已安裝，則會更版本。

-U 則不會檢查系統上是否已經安裝，若已安裝者，昇級之，若未安裝者，則安裝之。

### 3. yum：(Yellow dog Updater, Modified)

yum 是一個自動更新/移除 rpm 套件的程式。它會自動計算相依性，判斷哪些應該安裝，哪些則不必。和 yum 相同性質的還有 apt，不過 yum 比 apt 還要優秀好用。

## 三、昇級與補強的檔案來源：

1. <http://ftp.kh.edu.tw/Linux/Caosity/centos/6/updates/>
2. <ftp://ftp.kh.edu.tw/pub/Linux/Caosity/centos/6/updates/>
3. 錄底下，有 SRPMS、i386、ppc、x86\_64、等各種 CPU 用之套件。一般而言，像我們常用的 INTEL CPU(386、486、PENTIUM II III 4 等)均以 i386 目錄內的檔案為主；Mac 機器請使用 ppc 目錄，若您使用有 INTEL 64Bit CPU(EM64T)或 AMD64 的機器，請您使用 x86\_64；而這些套件的原始檔，則全部在 SRPMS 目錄內。
4. 下載原則上全部下載到您的系統中，再去篩選：(1)同一套件，不同版次的檔案，只保留最新版次，其餘均刪除(2)同一套件，不同 CPU 版本者，只保留您所需要的 CPU 版本，其餘均刪除；如同一套件有.i386.rpm、.i586.rpm 及.i686.rpm 版本者，若您的 CPU 是 PentiumII 以上者，只保留.i686.rpm 檔案，而刪除.i386.rpm 或.i586.rpm 檔案。

#### 四、快速安裝及修正自動昇級套件之設定：

1. 利用 **yum** 與自動排序機制來使作業系統能自動更新，固然方便，但對於初學者而言，光是修改設定檔就吃足了苦頭；因此，在考量初學者及各校忙錄的網管人員，市網中心這邊就編寫了自動修正的程式，來協助使用者方便地修正自己的系統，以便讓系統能夠自動地更新系統套件，以維護系統之基本安全。
2. 本次自動化修正檔，除了自動修正 CentOS6 系統更新套件的來源外，另增加一個額外常用的套件來源(CentOS-CR、kbs-CentOS-Extras、kbs-CentOS-Misc、crash-hat、rpmforge)。
3. 市網設定檔來源：  
`ftp://ftp.kh.edu.tw/pub/APPL/yum/CentOS6/fix-yum-CentOS6.sh`
4. 先下載 **ftp client** 工具(**lftp**):

```
[root@jserver ~]# yum install lftp
Loaded plugins: fastestmirror
Determining fastest mirrors
* base: ftp.cs.pu.edu.tw
* extras: centos.vr-zone.com
* updates: ftp.cs.pu.edu.tw
base | 3.7 kB 00:00
base/primary_db | 4.2 MB 00:42
extras | 3.0 kB 00:00
extras/primary_db | 1.9 kB 00:00
updates | 3.5 kB 00:00
updates/primary_db | 3.3 MB 00:27
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package lftp.x86_64 0:4.0.9-1.el6 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
lftp x86_64 4.0.9-1.el6 base 753 k

Transaction Summary
=====
Install 1 Package(s)
Upgrade 0 Package(s)

Total download size: 753 k
Installed size: 2.5 M
Is this ok [y/N]: y
Downloading Packages:
lftp-4.0.9-1.el6.x86_64.rpm | 753 kB 00:08
警告：rpmts_HdrFromFdno: 表頭 V3 RSA/SHA256 Signature, key ID c105b9de: NOKEY
base/gpgkey | 3.3 kB 00:00 ...
Importing GPG key 0xC105B9DE "CentOS-6 Key (CentOS 6 Official Signing Key) <centos-6-key@centos.org>"
from /etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6
Is this ok [y/N]: y
Running rpm_check_debug
```

```
Running Transaction Test
Transaction Test Succeeded
Running Transaction
Installing : lftp-4.0.9-1.el6.x86_64
```

```
Installed:
lftp.x86_64 0:4.0.9-1.el6
```

```
Complete!
```

1. 下載自動修正執行檔：

```
[root@jsserver root]# lftp ftp.kh.edu.tw
lftp ftp.kh.edu.tw:~> cd /pub/APPL/yum/CentOS6/
cd 成功，cwd 為 /pub/APPL/yum/CentOS6.]
lftp ftp.kh.edu.tw:/pub/APPL/yum/CentOS6> ls
drwxr-xr-x 2 root root 4096 Nov 4 04:43 .
drwxr-xr-x 15 root root 4096 Jul 12 01:29 ..
-r--r--r-- 1 root root 1977 Jul 12 01:47 CentOS-Base.repo
-r--r--r-- 1 root root 519 Oct 24 06:05 CentOS-CR.repo
-r--r--r-- 1 root root 1701 Sep 15 03:48 README.CentOS6
-r--r--r-- 1 root root 1365 May 1 2005 RPM-GPG-KEY-karan.org.txt
-r--r--r-- 1 root root 230 Oct 28 2008 crash-hat.repo
-r--r--r-- 1 root root 233 Aug 12 2007 dries.repo
-r--r--r-- 1 root root 1352 Oct 25 08:30 fix-yum-CentOS6.sh
-r--r--r-- 1 root root 1349 Nov 4 04:43 fix-yum-CentOS6_i686.sh
-r--r--r-- 1 root root 160 May 21 2007 i18n
-r--r--r-- 1 root root 161 May 21 2007 i18n-utf8
-r--r--r-- 1 root root 617 May 14 2010 kbsingh-CentOS-Extras.repo
-r--r--r-- 1 root root 752 May 14 2010 kbsingh-CentOS-Misc.repo
lftp ftp.kh.edu.tw:/pub/APPL/yum/CentOS6> get fix-yum-CentOS6.sh
傳送了 1352 位元組
lftp ftp.kh.edu.tw:/pub/APPL/yum/CentOS6> exit
```

2. 變更執行檔的檔案權限並執行之，以進行 yum 相關的設定：

```
[root@jsserver root]# chmod 755 fix-yum-CentOS6.sh
[root@jsserver root]# ls -al fix-yum-CentOS6.sh
-rwxr-xr-x. 1 root root 1352 2011-10-25 16:30 fix-yum-CentOS6.sh
[root@jsserver root]# ./fix-yum-CentOS6.sh

Now! I download the new configure files!
--2011-11-16 00:06:10-- http://ftp.kh.edu.tw/APPL/yum/CentOS6/CentOS-Base.repo
正在查找主機 ftp.kh.edu.tw... 163.28.136.5
正在連接 ftp.kh.edu.tw|163.28.136.5|:80... 連上了。
已送出 HTTP 要求，正在等候回應... 200 OK
長度: 1977 (1.9K) [text/plain]
Saving to: `CentOS-Base.repo'

100%[=====>] 1,977 --K/s in 0.001s

2011-11-16 00:06:10 (1.33 MB/s) - `CentOS-Base.repo' saved [1977/1977]

Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
Setting up Install Process
```

## Resolving Dependencies

--> Running transaction check

---> Package centos-release-cr.x86\_64 10:6-0.el6.centos set to be updated

--> Finished Dependency Resolution

## Dependencies Resolved

=====

Package Arch Version Repository Size

=====

Installing:

centos-release-cr x86\_64 10:6-0.el6.centos extras 3.9 k

## Transaction Summary

=====

Install 1 Package(s)

Upgrade 0 Package(s)

Total download size: 3.9 k

Installed size: 509

Downloading Packages:

centos-release-cr-6-0.el6.centos.x86\_64.rpm | 3.9 kB 00:00

Running rpm\_check\_debug

Running Transaction Test

Transaction Test Succeeded

Running Transaction

Installing : 10:centos-release-cr-6-0.el6.centos.x86\_64 1/1

Installed:

centos-release-cr.x86\_64 10:6-0.el6.centos

Complete!

--2011-11-16 00:06:12-- http://ftp.kh.edu.tw/APPL/yum/CentOS6/CentOS-CR.repo

正在查找主機 ftp.kh.edu.tw... 163.28.136.5

正在連接 ftp.kh.edu.tw|163.28.136.5|:80... 連上了。

已送出 HTTP 要求，正在等候回應... 200 OK

長度: 519 [text/plain]

Saving to: `CentOS-CR.repo'

100%[=====>] 519 --K/s in 0s

2011-11-16 00:06:12 (97.8 MB/s) - `CentOS-CR.repo' saved [519/519]

--2011-11-16 00:06:12-- http://centos.karan.org/kbsingh-CentOS-Extras.repo

正在查找主機 centos.karan.org... 66.36.231.4

正在連接 centos.karan.org|66.36.231.4|:80... 連上了。

已送出 HTTP 要求，正在等候回應... 200 OK

長度: 664 [text/plain]

Saving to: `kbsingh-CentOS-Extras.repo'

100%[=====>] 664 --K/s in 0s

2011-11-16 00:06:13 (88.1 MB/s) - `kbsingh-CentOS-Extras.repo' saved [664/664]

--2011-11-16 00:06:13-- http://centos.karan.org/kbsingh-CentOS-Misc.repo

正在查找主機 centos.karan.org... 66.36.231.4

```
正在連接 centos.karan.org|66.36.231.4|:80... 連上了。
已送出 HTTP 要求，正在等候回應... 200 OK
長度: 493 [text/plain]
Saving to: `kbsingh-CentOS-Misc.repo'

100%[=====>] 493 --K/s in 0s

2011-11-16 00:06:14 (66.2 MB/s) - `kbsingh-CentOS-Misc.repo' saved [493/493]

--2011-11-16 00:06:14-- http://ftp.kh.edu.tw/APPL/yum/CentOS6/crash-hat.repo
正在查找主機 ftp.kh.edu.tw... 163.28.136.5
正在連接 ftp.kh.edu.tw|163.28.136.5|:80... 連上了。
已送出 HTTP 要求，正在等候回應... 200 OK
長度: 230 [text/plain]
Saving to: `crash-hat.repo'

100%[=====>] 230 --K/s in 0s

2011-11-16 00:06:14 (48.2 MB/s) - `crash-hat.repo' saved [230/230]

--2011-11-16 00:06:14-- http://ftp.kh.edu.tw/APPL/yum/CentOS6/dries.repo
正在查找主機 ftp.kh.edu.tw... 163.28.136.5
正在連接 ftp.kh.edu.tw|163.28.136.5|:80... 連上了。
已送出 HTTP 要求，正在等候回應... 200 OK
長度: 233 [text/plain]
Saving to: `dries.repo'

100%[=====>] 233 --K/s in 0s

2011-11-16 00:06:14 (38.2 MB/s) - `dries.repo' saved [233/233]

--2011-11-16 00:06:14-- http://packages.sw.be/rpmforge-release/rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm
正在查找主機 packages.sw.be... 78.46.17.228
正在連接 packages.sw.be|78.46.17.228|:80... 連上了。
已送出 HTTP 要求，正在等候回應... 301 Moved Permanently
位置: http://pkgs.repoforge.org/rpmforge-release/rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm [跟隨至新的 URL]
--2011-11-16 00:06:15-- http://pkgs.repoforge.org/rpmforge-release/rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm
正在查找主機 pkgs.repoforge.org... 78.46.17.228
正在連接 pkgs.repoforge.org|78.46.17.228|:80... 連上了。
已送出 HTTP 要求，正在等候回應... 302 Found
位置: http://rpmforge.sw.be/redhat/el6/en/x86_64/rpmforge/RPMS/rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm
[跟隨至新的 URL]
--2011-11-16 00:06:16--
http://rpmforge.sw.be/redhat/el6/en/x86_64/rpmforge/RPMS/rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm
正在查找主機 rpmforge.sw.be... 78.46.17.228
正在連接 rpmforge.sw.be|78.46.17.228|:80... 連上了。
已送出 HTTP 要求，正在等候回應... 301 Moved Permanently
位置:
http://tree.repoforge.org/redhat/el6/en/x86_64/rpmforge/RPMS/rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm [跟
隨至新的 URL]
--2011-11-16 00:06:19--
http://tree.repoforge.org/redhat/el6/en/x86_64/rpmforge/RPMS/rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm
正在查找主機 tree.repoforge.org... 78.46.17.228
正在連接 tree.repoforge.org|78.46.17.228|:80... 連上了。
已送出 HTTP 要求，正在等候回應... 301 Moved Permanently
```

```

位置: http://apt.sw.be/redhat/el6/en/x86_64/rpmforge/RPMS/rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm [跟隨至新的 URL]
--2011-11-16 00:06:20--
http://apt.sw.be/redhat/el6/en/x86_64/rpmforge/RPMS/rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm
正在查找主機 apt.sw.be... 193.1.193.67
正在連接 apt.sw.be|193.1.193.67|:80... 連上了。
已送出 HTTP 要求，正在等候回應... 200 OK
長度: 12700 (12K) [application/x-redhat-package-manager]
Saving to: `rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm'

100%[=====>] 12,700 13.8K/s in 0.9s

2011-11-16 00:06:23 (13.8 KB/s) - `rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm' saved [12700/12700]

警告：rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm: 表頭 V3 DSA/SHA1 Signature, key ID 6b8d79e6: NOKEY
正在準備... ##### [100%]
1:rpmforge-release ##### [100%]

Successe! I had fix the yum configure!

```

3. 因應 rpmforge 套件庫所有的 mirror 站都在國外，而目前國外連線的速度奇慢，為避免第一次做 update 時等待時間過久，甚至中斷，請增修/etc/yum.repos.d/rpmforge.repo 設定：

```

### Name: RPMforge RPM Repository for RHEL 6 - dag
### URL: http://rpmforge.net/
[rpmforge]
name = RHEL $releasever - RPMforge.net - dag
baseurl = http://apt.sw.be/redhat/el6/en/$basearch/rpmforge
mirrorlist = http://apt.sw.be/redhat/el6/en/mirrors-rpmforge
#mirrorlist = file:///etc/yum.repos.d/mirrors-rpmforge
enabled = 1
protect = 0
gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY-rpmforge-dag
gpgcheck = 1
proxy=http://open.kh.edu.tw:3128/
timeout=300

```

4. 安裝 Extra Packages for Enterprise Linux 套件庫：

EPEL11 套件庫為 RedHat 的 Fedora 計畫下的一個方案，主要是為了產生、維護及經營高品質的軟體供企業等級的 GNU/Linux 發行版使用，企業等級的版本是指 RedHat Enterprise Linux(RHEL)、CentOS 及 ScientificLinux(SL)。當然，EPEL 中的軟體都是自由軟體，其它發行版本是可以自由的取用的。事實上，EPEL 中的軟體大都來至 Fedora 發版中的軟體，這些軟體都已由全世界 Fedora 使用者及社群長時間使用過。

```

[root@test ~]# rpm -Uvh http://download.fedora.redhat.com/pub/epel/6/i386/epel-release-6-7.noarch.rpm
取回中 http://download.fedora.redhat.com/pub/epel/6/i386/epel-release-6-7.noarch.rpm
警告：/var/tmp/rpm-tmp.GOMiMx: 表頭 V3 RSA/SHA256 Signature, key ID 0608b895: NOKEY
正在準備... ##### [100%]
1:epel-release ##### [100%]

```



5. 因應 EPEL11 套件庫所有的 mirror 站都在國外，而目前國外連線的速度奇慢，為避免第一次做 update 時等待時間過久，甚至中斷，請增修/etc/yum.repos.d/epel.repo 設定：

```
[epel]
name=Extra Packages for Enterprise Linux 6 - $basearch
#baseurl=http://download.fedoraproject.org/pub/epel/6/$basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-6&arch=$basearch
failovermethod=priority
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6
proxy=http://open.kh.edu.tw:3128/
timeout=300
```

6. 進行自動化更新程序：

```
[root@jsrver root]# yum update
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
epel/metalink | 3.8 kB 00:00
* epel: mirror.bjtu.edu.cn
* rpmforge: fr2.rpmfind.net
cr | 3.5 kB 00:00
cr/primary_db | 3.9 MB 00:03
epel | 4.3 kB 00:00
epel/primary_db | 4.2 MB 01:45
kbs-CentOS-Extras | 1.9 kB 00:00
kbs-CentOS-Extras/primary_db | 1.1 kB 00:00
kbs-CentOS-Misc | 1.9 kB 00:00
kbs-CentOS-Misc/primary_db | 1.1 kB 00:00
rpmforge | 1.1 kB 00:00
rpmforge/primary | 1.4 MB 00:10
rpmforge 4090/4090
Setting up Update Process
Resolving Dependencies
--> Running transaction check
---> Package abrt.x86_64 0:1.1.16-3.el6 set to be updated
--> Processing Dependency: abrt-plugin-mailx = 1.1.16-3.el6 for package: abrt-1.1.16-3.el6.x86_64
---> Package abrt-addon-ccpp.x86_64 0:1.1.16-3.el6 set to be updated
---> Package abrt-addon-kerneloops.x86_64 0:1.1.16-3.el6 set to be updated
:::
---> Package ibus-libs.x86_64 0:1.3.4-5.el6 set to be updated
---> Package ibus-table-chinese-cangjie.noarch 0:1.3.4-1.el6 set to be updated
--> Processing Dependency: ibus-table-chinese = 1.3.4-1.el6 for package: ibus-table-chinese-cangjie-1.3.4-1.el6.noarch
---> Package ibus-table-chinese-erbi.noarch 0:1.3.4-1.el6 set to be updated
---> Package ibus-table-chinese-scj.noarch 0:1.3.4-1.el6 set to be updated
---> Package ibus-table-chinese-wubi-haifeng.noarch 0:1.3.4-1.el6 set to be updated
:::
---> Package libsdl.x86_64 0:2.0.94-5.el6 set to be updated
---> Package libsndfile.x86_64 0:1.0.20-5.el6 set to be updated
--> Processing Dependency: libFLAC.so.8()(64bit) for package: libsndfile-1.0.20-5.el6.x86_64
---> Package libss.x86_64 0:1.41.12-7.el6 set to be updated
---> Package libssh2.x86_64 0:1.2.2-7.el6_1.1 set to be updated
---> Package libstdc++.x86_64 0:4.4.5-6.el6 set to be updated
:::
---> Package rpm.x86_64 0:4.8.0-16.el6_1.1 set to be updated
---> Package rpm-build.x86_64 0:4.8.0-16.el6_1.1 set to be updated
```

```

--> Processing Dependency: /usr/bin/gdb-add-index for package: rpm-build-4.8.0-16.el6_1.1.x86_64
--> Package rpm-libs.x86_64 0:4.8.0-16.el6_1.1 set to be updated
--> Package rpm-python.x86_64 0:4.8.0-16.el6_1.1 set to be updated
--> Package rsync.x86_64 0:3.0.6-5.el6_0.1 set to be updated
:::
--> Package yum-plugin-fastestmirror.noarch 0:1.1.30-6.el6 set to be updated
--> Package yum-utils.noarch 0:1.1.30-6.el6 set to be updated
--> Package zlib.x86_64 0:1.2.3-26.el6 set to be updated
--> Running transaction check
--> Package abrt-plugin-mailx.x86_64 0:1.1.16-3.el6 set to be updated
--> Package flac.x86_64 0:1.2.1-6.1.el6 set to be updated
--> Package gdb.x86_64 0:7.2-48.el6 set to be updated
--> Package ibus-table-chinese.noarch 0:1.3.4-1.el6 set to be updated
--> Finished Dependency Resolution

```

#### Dependencies Resolved

```

=====
Package Arch Version Repository
Size
=====

```

#### Installing:

```

ibus-table-chinese-cangjie noarch 1.3.4-1.el6 epel 3.7 M
replacing ibus-table-cangjie.noarch 1.2.0.20100210-1.el6
ibus-table-chinese-erbi noarch 1.3.4-1.el6 epel 4.2 M
replacing ibus-table-erbi.noarch 1.2.0.20090901-6.el6
ibus-table-chinese-scj noarch 1.3.4-1.el6 epel 1.7 M
replacing ibus-table-cangjie.noarch 1.2.0.20100210-1.el6
ibus-table-chinese-wubi-haifeng
noarch 1.3.4-1.el6 epel 4.3 M
replacing ibus-table-wubi.noarch 1.2.0.20090715-8.el6
ibus-table-chinese-wubi-jidian noarch 1.3.4-1.el6 epel 4.1 M
replacing ibus-table-wubi.noarch 1.2.0.20090715-8.el6
kernel x86_64 2.6.32-131.17.1.el6 cr 23 M
kernel-devel x86_64 2.6.32-131.17.1.el6 cr 6.9 M

```

#### Updating:

```

abrt x86_64 1.1.16-3.el6 cr 323 k
abrt-addon-ccpp x86_64 1.1.16-3.el6 cr 67 k
abrt-addon-kerneloops x86_64 1.1.16-3.el6 cr 63 k

```

```

:::

```

```

yum noarch 3.2.29-17.el6.centos.1 cr 973 k
yum-metadata-parser x86_64 1.1.2-16.el6 cr 27 k
yum-plugin-fastestmirror noarch 1.1.30-6.el6 cr 26 k
yum-utils noarch 1.1.30-6.el6 cr 93 k
zlib x86_64 1.2.3-26.el6 cr 72 k

```

#### Installing for dependencies:

```

abrt-plugin-mailx x86_64 1.1.16-3.el6 cr 56 k
flac x86_64 1.2.1-6.1.el6 base 243 k
gdb x86_64 7.2-48.el6 cr 2.3 M
ibus-table-chinese noarch 1.3.4-1.el6 epel 19 k

```

#### Transaction Summary

```

=====
Install 11 Package(s)
Upgrade 296 Package(s)

```

```

Total download size: 309 M

```

```

Is this ok [y/N]: y
Downloading Packages:
Downloading Packages:
(1/307): abrt-1.1.16-3.el6.x86_64.rpm | 323 kB 00:00
(2/307): abrt-addon-ccpp-1.1.16-3.el6.x86_64.rpm | 67 kB 00:00
(3/307): abrt-addon-kerneloops-1.1.16-3.el6.x86_64.rpm | 63 kB 00:00
(4/307): abrt-addon-python-1.1.16-3.el6.x86_64.rpm | 56 kB 00:00
(5/307): abrt-cli-1.1.16-3.el6.x86_64.rpm | 76 kB 00:00
(6/307): abrt-libs-1.1.16-3.el6.x86_64.rpm | 103 kB 00:00
(7/307): abrt-plugin-logger-1.1.16-3.el6.x86_64.rpm | 54 kB 00:00
(8/307): abrt-plugin-mailx-1.1.16-3.el6.x86_64.rpm | 56 kB 00:00
(9/307): abrt-plugin-rhtsupport-1.1.16-3.el6.x86_64.rpm | 58 kB 00:00
(10/307): abrt-plugin-sosreport-1.1.16-3.el6.x86_64.rpm | 51 kB 00:00
      :
      :
      :
(300/307): xmlrpc-c-client-1.16.24-1200.1840.el6_1.4.x86_64 | 25 kB 00:00
(301/307): xorg-x11-server-utils-7.4-15.el6_0.2.x86_64.r | 130 kB 00:00
(302/307): xulrunner-1.9.2.23-1.el6.centos.1.x86_64.rpm | 8.9 MB 00:06
(303/307): yum-3.2.29-17.el6.centos.1.noarch.rpm | 973 kB 00:00
(304/307): yum-metadata-parser-1.1.2-16.el6.x86_64.rpm | 27 kB 00:00
(305/307): yum-plugin-fastestmirror-1.1.30-6.el6.noarch. | 26 kB 00:00
(306/307): yum-utils-1.1.30-6.el6.noarch.rpm | 93 kB 00:00
(307/307): zlib-1.2.3-26.el6.x86_64.rpm | 72 kB 00:00
-----
Total 853 kB/s | 309 MB 06:11
警告 : rpmts_HdrFromFdno: 表頭 V3 RSA/SHA256 Signature, key ID 0608b895: NOKEY
epel/gpgkey | 3.2 kB 00:00 ...
Importing GPG key 0x0608B895 "EPEL (6) <epel@fedoraproject.org>" from /etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6
Is this ok [y/N]: y
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
Warning: RPMDB altered outside of yum.
Updating : libgcc-4.4.5-6.el6.x86_64 1/606
Installing : ibus-table-chinese-1.3.4-1.el6.noarch 2/606
Updating : setup-2.8.14-13.el6.noarch 3/606
warning: /etc/shadow created as /etc/shadow.rpmnew
Updating : cjkuni-fonts-common-0.2.20080216.1-35.el6.noarch 4/606
Updating : filesystem-2.4.30-3.el6.x86_64 5/606
Updating : kernel-firmware-2.6.32-131.17.1.el6.noarch 6/606
Updating : hwddata-0.233-7.1.el6.noarch 7/606
Updating : tzdata-2011h-3.el6.noarch 8/606
      :
      :
      :
      :
Updating : man-pages-overrides-2.5-1.el6.noarch 303/606
Updating : man-pages-3.22-17.el6.noarch 304/606
Updating : iwl5000-firmware-8.83.5.1_1-1.el6_1.1.noarch 305/606
Updating : ql2500-firmware-5.03.16-1.el6.noarch 306/606
Updating : ql2400-firmware-5.03.16-1.el6.noarch 307/606
Cleanup : rpm-build-4.8.0-12.el6.x86_64 308/606
Cleanup : openssh-server-5.3p1-20.el6.x86_64 309/606
Cleanup : abrt-cli-1.1.13-4.el6.x86_64 310/606
Cleanup : hal-0.5.14-8.el6.x86_64 311/606
Cleanup : 1:dovecot-2.0-0.10.beta6.20100630.el6.x86_64 312/606
      :
      :
      :
      :
Cleanup : man-pages-overrides-1.0-1.el6.noarch 602/606
Cleanup : man-pages-3.22-12.el6.noarch 603/606

```

```
Cleanup : iwl5000-firmware-8.24.2.12-3.el6.noarch 604/606
Cleanup : ql2500-firmware-5.03.02-2.el6.noarch 605/606
Cleanup : ql2400-firmware-5.03.02-1.el6.noarch 606/606
```

Installed:

```
ibus-table-chinese-cangjie.noarch 0:1.3.4-1.el6
ibus-table-chinese-erbi.noarch 0:1.3.4-1.el6
ibus-table-chinese-scj.noarch 0:1.3.4-1.el6
ibus-table-chinese-wubi-haifeng.noarch 0:1.3.4-1.el6
ibus-table-chinese-wubi-jidian.noarch 0:1.3.4-1.el6
kernel.x86_64 0:2.6.32-131.17.1.el6
kernel-devel.x86_64 0:2.6.32-131.17.1.el6
```

Dependency Installed:

```
abrt-plugin-mailx.x86_64 0:1.1.16-3.el6
flac.x86_64 0:1.2.1-6.1.el6
gdb.x86_64 0:7.2-48.el6
ibus-table-chinese.noarch 0:1.3.4-1.el6
```

Updated:

```
abrt.x86_64 0:1.1.16-3.el6
abrt-addon-ccpp.x86_64 0:1.1.16-3.el6
abrt-addon-kerneloops.x86_64 0:1.1.16-3.el6
abrt-addon-python.x86_64 0:1.1.16-3.el6
abrt-cli.x86_64 0:1.1.16-3.el6
abrt-libs.x86_64 0:1.1.16-3.el6
:::
yum.noarch 0:3.2.29-17.el6.centos.1
yum-metadata-parser.x86_64 0:1.1.2-16.el6
yum-plugin-fastestmirror.noarch 0:1.1.30-6.el6
yum-utils.noarch 0:1.1.30-6.el6
zlib.x86_64 0:1.2.3-26.el6
```

Replaced:

```
ibus-table-cangjie.noarch 0:1.2.0.20100210-1.el6
ibus-table-erbi.noarch 0:1.2.0.20090901-6.el6
ibus-table-wubi.noarch 0:1.2.0.20090715-8.el6
```

Complete!

7. 因為 **CentOS6** 有多個套件需要更新，您可起來喝個水或上個廁所、泡杯咖啡，再回來  
看正否更新完成。
8. 更新到最近後，記得清除一下下載的檔案及暫存的檔。

```
[root@jsserver root]# yum clean packages
Cleaning up Packages
21 packages removed
```

9. 別忘了加入自動化排程，讓作業系統每天自動去檢查是否要更新套件！

方法一：

安裝 **yum-cron** 套件：

```
[root@test ~]# yum install yum-cron
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
* epel: ftp.cuhk.edu.hk
* rpmforge: fr2.rpmfind.net
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package yum-cron.noarch 0:3.2.29-17.el6.centos.1 will be installed
--> Processing Dependency: yum-plugin-downloadonly for package: yum-cron-3.2.29-17.el6.centos.1.noarch
--> Running transaction check
--> Package yum-plugin-downloadonly.noarch 0:1.1.30-6.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository
Size
=====
Installing:
yum-cron noarch 3.2.29-17.el6.centos.1 cr 32 k
Installing for dependencies:
yum-plugin-downloadonly noarch 1.1.30-6.el6 cr 18 k

Transaction Summary
=====
Install 2 Package(s)

Total download size: 50 k
Installed size: 47 k
Is this ok [y/N]: y
Downloading Packages:
(1/2): yum-cron-3.2.29-17.el6.centos.1.noarch.rpm | 32 kB 00:00
(2/2): yum-plugin-downloadonly-1.1.30-6.el6.noarch.rpm | 18 kB 00:00
-----
Total 438 kB/s | 50 kB 00:00
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
Installing : yum-plugin-downloadonly-1.1.30-6.el6.noarch 1/2
Installing : yum-cron-3.2.29-17.el6.centos.1.noarch 2/2

Installed:
yum-cron.noarch 0:3.2.29-17.el6.centos.1

Dependency Installed:
yum-plugin-downloadonly.noarch 0:1.1.30-6.el6

Complete!
```

若上述檔案存在，則請將 **yum-updatesd** 服務設定成開機啟動：

```
[root@jsserver root]# chkconfig --level 345 yum-cron on
```

然後手動啟動 yum 服務，即可讓系統每天定時去更新。

```
[root@jsserver root]# service yum-cron start
```

正在啟用每晚的 yum 更新： [ 確定 ]

方法二：

利用系統排程命令，自行將 yum 加入系統排程中：

```
[root@jsserver root]# crontab -u root -e
```

內容加入以下這行：

```
00 07 * * * /usr/bin/yum -y update;/bin/sleep 30;/usr/bin/yum clean packages
```

# Linux 常用命令與系統管理

高雄市政府教育局  
資訊教育中心  
網路組 張宏明  
更新日期：2012/11/14

## 壹、Linux 基本指令介紹：

Linux 指令繁多，在此僅列出常用的給大家參考  
系統管理

指令	說明
adduser	<div>釋:建立使用者帳號</div> <div>例:</div> <div>adduser coco #新增一個 coco 帳號</div>
userdel	<div>釋:刪除使用者帳號</div> <div>例:</div> <div>userdel -r coco #刪除 coco 帳號，並且將他的目錄一併刪除</div>
groupadd	<div>釋:建立群組</div> <div>例:</div> <div>groupadd student #新增一個 student 群組</div>
groupdel	<div>釋:刪除群組</div> <div>例:</div> <div>groupdel student #刪除 student 群組</div>
passwd	<div>釋:使用者可以更改自己的密碼，而系統管理者可以管理使用者的密碼</div> <div>例:</div> <div>passwd #更改自己的密碼</div> <div>passwd coco #設定或更改使用者密碼</div>
date	<div>釋:顯示或設定系統時間與日期</div> <div>例:</div> <div>date MMDDhhmmCCYY.ss</div> <div>date #顯示系統目前的時間與日期</div> <div>date 122012301999.20 #將系統時間設為 1999 年 12 月 20 日 12 時 30 分 20 秒</div>
login	<div>釋:登入系統</div>
logout	<div>釋:登出系統</div>
reboot	<div>釋:重新開機</div>
shutdown	<div>釋:系統關機</div> <div>例:</div> <div>shutdown -h now #馬上關閉</div> <div>shutdown +10 #10 分鐘後關閉</div> <div>shutdown 14:30 #在 14:30 關閉</div> <div>shutdown -h +10 "系統十分鐘後關機" #10 分鐘後關閉並通知使用者</div>

	shutdown -c "Shutdown Canceled" #取消關閉並通知通知使用者 Ctrl + C #取消 shutdown 指令
halt	<b>釋</b> :關閉系統
su	<b>釋</b> :轉換其他使用者身份 <b>例</b> : su coco #轉成 coco 使用者身份，下 exit 可以切回自己身份
w	<b>釋</b> :顯示目前登入系統的使用者資訊
crontab	<b>釋</b> :設定計時器 <b>例</b> : -e #編輯該使用者的計時器設定 -l #列出該使用者的計時器設定 -r #刪除該使用者的計時器設定 -u #指定要設定計時器的使用者 * * * * * Command 分 時 日 月 星期 命令 crontab -e -u root #設定 root 的計時器
setup	<b>釋</b> :設定公用程式
linuxconf	<b>釋</b> :設定 Linux 系統，linuxconf 是 Red Hat Linux 發行版專門用來調整 Linux 各項設定的程式，藉由互動式操作介面，進行 Linux 系統的各種設定，而不需下達各項參數或撰寫複雜的設定檔。

## 檔案管理

指令	說明
cd	<b>釋</b> : 切換目錄 <b>例</b> : cd #馬上回到目前使用者的目錄 cd /home/coco #從現在目錄切換到 /home/coco 目錄中 cd .. #回到上層目錄
clear	<b>釋</b> :消除螢幕畫面
ls	<b>釋</b> :列出目錄內容 <b>例</b> : ls #僅列出目錄檔案名稱 ls -la #列出目錄檔案名稱及詳細資料
mkdir	<b>釋</b> :建立目錄 <b>例</b> : mkdir dog #建立一個 dog 目錄 mkdir -m 755 cat #建立一個 cat 目錄，並將該目錄權限設為 755
rmdir	<b>釋</b> :刪除空目錄 <b>例</b> : rmdir cat #刪除空目錄 cat



mount	<p>釋:掛入檔案系統</p> <p>例:</p> <p>mount /mnt/cdrom #掛入光碟機</p> <p>mount /mnt/floppy #掛入軟碟機</p>
umount	<p>釋:卸除檔案系統</p> <p>例:</p> <p>umount /mnt/cdrom #卸除光碟機</p> <p>umount /mnt/floppy #卸除軟碟機</p>
pwd	<p>釋:顯示目前工作目錄</p>
cp	<p>釋:複製檔案或目錄</p>
mv	<p>釋:搬移或更名現有的目錄或檔案</p> <p>例:</p> <p>mv dog /home/ #將 dog 檔案或目錄搬移到 /home 目錄下</p> <p>mv dog /home/cat 將 dog 檔案或目錄搬移到 /home 目錄下，並將目錄改為 cat</p>
rm	<p>釋:移除檔案或目錄</p> <p>例:</p> <p>rm -r sheep #刪除 sheep 目錄及目錄裡的所有檔案</p> <p>rm sheep #刪除 sheep 這個檔案，會詢問是否確定？</p>
find	<p>釋:尋找檔案或目錄</p> <p>例:</p> <p>find / -name pwd -print (尋找/下，檔名為 pwd 的檔案)</p> <p>find / -name 'pr???' -print (尋找/下，檔名為 pr???的檔案)</p> <p>find /usr -iname 'pr*' -print (尋找/usr 下，檔名開頭為 pr 的檔案，不分大小寫)</p>
whereis	<p>釋:搜尋 path 下的檔案或目錄 (找系統執行檔的路徑有用)</p> <p>例:</p> <p>whereis cp #尋找 cp 指令</p>
touch	<p>釋:更改檔案日期，不加參數時，可建立空檔案</p> <p>例:</p> <p>touch file1 #建立一個 file1 空檔案</p>
df	<p>釋:顯示檔案系統的利用度</p>
du	<p>釋:檔案或目錄所佔空間大小</p> <p>例:</p> <p>-m #單位為 megabyte(預設為 k)</p> <p>-s #全部加總</p> <p>du</p> <p>du -s /usr</p> <p>du -sm /usr</p>
more	<p>釋:分頁顯示</p> <p>例:</p> <p>ls -la   more</p>
格式化	<p>例:</p>

(format)軟碟	fdformat /dev/fd0H1440 #低階格式化 mke2fs /dev/fd0 #建立 ex2 filesystem
mkbootdisk	釋:製作 Linux 開機片 例: mke2fs /dev/fd0 #先格式化磁片 mkbootdisk --device /dev/fd0 kernel 版本 kernel 版本可下 uname -a 得知或 ls /lib/modules mkbootdisk --device /dev/fd0 2.2.17-14

## 貳、檔案目錄簡介

在 Unix 系統中會將檔案按照其類型分門別類的置放於不同目錄下，所以身為系統管理者若要更改系統或管理系統的東西時要先了解檔案放哪邊，如您對目錄結構夠了解的話馬上就能找到所需要的檔案加以更改。

實體上，Linux 中的「檔案系統」，相當於 DOS 中的 partition。一個獨立的檔案系統擁有獨立的 file table(inode table)，一個磁碟中，可以切割成一個上的檔案系統。

但就邏輯上而言，一個 Linux 中，僅有一個檔案系統，亦即，一個 Linux 作業系統中僅有一個「根目錄」，一切檔案結構依此根目錄，往下發展。

UNIX like 的目錄結構基本上屬於樹狀結構（階層式），類似 MS DOS 和 WIN32 系統中的樹狀目錄架構，正確說法應該是 MS DOS 和 WIN32 系統中的樹狀目錄架構是源自 UNIX 系統中的目錄架構。

### ◆ 常見重要的目錄(掛入點)

/	根目錄，包含整個 Linux 系統的所有目錄和檔案。
/usr	用來存放系統指令、應用程式等。
/var	系統執行時，需要暫時記錄部分資料或存放一些暫存檔，都會放置在這個目錄裡。如 Apache、FTP、MAIL 等。
/home	預設用來放置帳號的自家目錄。
/tmp	供全部使用者暫時放置檔案的目錄。
/bin	放置操作系統時，所需使用的各種指令。例如 cp、login、mv、rm 等。
/boot	系統啟動時必須讀取的檔案，包含系統核心在內。
/dev	存放周邊設備代號的檔案。如硬碟的 /dev/hda 。
/etc	放置與系統設定、管理相關的檔案。如記載帳號名稱的 passwd 檔，投影密碼檔 shadow 檔。
/mnt	預設有 /mnt/cdrom 和 /mnt/floppy 兩個目錄在這裡，用來做為光碟機和軟碟機掛載點。
/root	系統管理者的家目錄。
/sbin	存放系統啟動時需執行的程式。

### ◆ 檔案系統架構

UNIX 檔案系統和 DOS 一樣，採階層式(Hierarchy)或樹狀(Tree)之目錄架構，最上一層目錄稱為根目錄(root)，每一目錄中均可含有檔案與子目錄，路徑名稱中目錄與目錄之分隔符號為「/」。

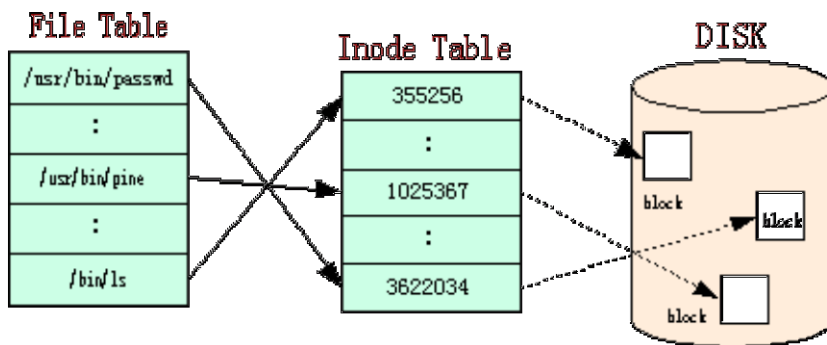
### ◆ Linux 檔案種類

- 一般檔案(ordinary files)：程式、文件、資料…等。
- 目錄檔案(directory files)：類似資料夾，目錄檔用於分類檔案，使檔案系統更加簡潔。
- 特殊檔案(Special device files)：代表某種特殊硬體設備，如：印表機、磁碟機、磁帶機…等，通常存在於 /dev 目錄之下。此種檔案類型又可區分為區塊(block)及字元(Character)兩種。(UNIX 作業系統下，所有設備均可視為檔案，並依檔案方式運作)
- 檔案權限開頭為 d 者，表示目錄；若是一個 - 號，表示普通檔案；以“.”開頭之檔名代表隱藏檔，必需用“ls -l”才可看到，如 .profile。
- 符號鏈結檔(Symbolic link)：

- 所謂的符號連結檔，就是該檔案的別名的意思
- Linux 中有二種符號連結檔:
- 硬式連結 (hard link)
- 軟式連結 (soft link)
- 所謂硬式連結檔，就是某一個檔案的複本，其檔內容大小完全和原本的檔案一模一樣，當原本檔案內容改變時，硬式連結檔也跟著作相同的改變，但若原檔被刪除，hard link 檔仍然會存在。
- 而軟式連結，則是該檔案的一個別名，用來連結至原本的檔案。
- 不管是那一種，存取符號連結檔，就等於是存取原來的那一個檔案。
- 符號連結檔有非常大的用途，這是 DOS/WIN 等其它平台無法比擬的。例如：某一個目錄空間不足，那麼，我們可以將新的硬碟空間，用符號連結檔的方式掛入，同時可保持原有的目錄結構不變。
  - 如何製造 symbolic link files ?
- **ln -sf target source**
- s 表示 soft link，f 表示若有同名檔存在，則將它覆蓋過去。
- 早期 UNIX 檔名長度限制不可超過 14 個字元，現今 Linux 系統則可允許檔名長度至 256 個字元。
- UNIX 檔案系統中，無所謂延伸檔名，對 Linux 而言檔名：aa.bb.cc.dd 是合法的，UNIX 檔案命名原則可用任何字元來命名(“/” 除外)，然特殊字元(\*?^< ...)，除“.”外最好避免使用，免生誤解。

#### ◆ inode

inode(information node)是一個資訊節點，記錄著其所指向檔案的一切重資訊，包括 inode 編號，檔案型態，檔案使用者以及群組 ID，檔案大小，時間，以及在磁碟中的區塊號碼(blocks)等。



## 參、檔案系統權限

Linux 檔案系統將檔案或目錄的使用者分成檔案擁有者(**Owner**)、群組(**Group**)及其他(**Other**)三大類；存取權力則分成讀取(**r**)、寫入(**w**)及執行(**x**)三種；藉著使用者區分與存取權力之給予之組合，組成 Linux 的檔案保護系統。

權限有三種，說明如下：

權限	顯示	二進位表示	十進位表示
Read(可讀)	r	100	4
Write(可寫)	w	010	2
eXcute(可執行)	x	001	1
Read,Write,eXcute	rwX	111	4+2+1=7
Read,eXcute	r-x	101	4+1=5
none	---	000	0
Read,Write	rw	110	4+2=6

### ◆誰才能改檔案權限

每一個檔均有一個擁有者(owner)，僅有 owner 及 root 可以更改檔案之存取保護模式(利用 **chmod** 指令)

### ◆只有 root 可以更改檔案之「擁有者」與「群組」(利用 **chown** 指令)。

### ◆特殊的檔案權限

權限	顯示	二進位表示	十進位表示
Setuid	s	100	4
Setgid	s	010	2
sTicky	t	001	1

### 特別說明：Setuid

在執行程式時，程式執行的身份為啟動該程式的 **user**，若設為 **setuid**，則該程式在執行身份就改為該程式的 **Owner**。

通常 **Setuid**(Set user id)的情況會發生在使用者需要 **root** 的權限時發生，意謂著這支程式必須要有 **root** 的權力才能做相關的動作。例如線上改密碼程式，一般人都可以執行它，但是這支程式需 **setuid**，其權限應該是 **-rwsr-xr-x**，表示在執行時是以 **root** 身份執行，能夠改某使用者的密碼，但是 **setuid** 有時是系統上安全的漏洞，在管理上必須嚴格控管具有 **setuid** 權限的檔案。

### ◆檔案權限修改指令 -- **chmod**，其語法如下：

**chmod [-R] 【mode=4,2,1 or {a,u,g,o}{+,-,=}{r,w,s,t}】 【檔案/目錄】**

其中

a	所有 User
u	檔案擁有者
g	群組

o	其他 User
+	增加權力
-	減少權力
=	權力指定
r	讀取
w	寫入
x	執行
s	setuid
t	sTicky(少用)
-R	Recursive，目錄下的檔案一併更改

例:

```
ls -al test
-rwxr-xr-x 1 air teacher 1224 Jul 4 12:02 test
chmod g+w,o-x test rwxrwxr--
chmod u=rwx test (=rwxrwxrwx)
chmod 755 test rwxr-xr-x
chmod a-x test (=rw-r--r--)
chmod 4655 test (=rwsr-xr-x)
chmod -R 755 test2 將 test2 這目錄下的檔案改為 755
```

◆改變檔案或目錄的擁有人-- chown，其語法如下：

chown [-R] 【使用者.[群組]】 【檔案/目錄】

## 肆、 Vi 編輯工具

要管理好 Linux 系統，少不了要對設定檔案進行編輯，Linux 提供許多文書編輯程式，例如 ed , vi , pico , joe , emacs 等，vi 可說是在 Linux 中功能最強的編輯工具，也是 Linux 預設的編輯程式。

vi 可說是 Linux 中功能相當強的編輯工具，也是目前 Linux 預設的編輯軟體。

vi 有三種模式：

▼ **command mode**：供使用者執行命令

▼ **input mode**：輸入內容

▼ **last line mode**：讓使用者做一些輸入文字無關的事，例如搜尋字串、儲存檔案或結束編輯等。

### ■ 執行與結束 vi

在提示符號下，執行 vi 指令，或是執行 vi myfile 指令指定要編輯的檔案。

### ■ 切換不同模式

進入 vi 時，預設的模式是 command mode，而要進入 input mode 輸入資料時，可按下列三個鍵：

a	從目前游標所在位置的下一個字元開始輸入
i	從目前游標所在位置插入新輸入的字元
o	新增一行，並將游標移到下一行的開頭

※ 在 command mode 輸入命令時（如"a"、"i"、"o"），字元並不會顯示出來。

若要從 input mode 回到 command mode，則需按下 Esc 鍵。

如要轉換為 last line mode 時，則在 command mode 輸入 ":"即可。

### ■ Command mode 模式的操作

Command mode 模式提供了相當多的按鍵以及組合鍵來執行命令，幫助使用者編輯文字。由於命令相當多，在此只能介紹一些平常比較常用到的。

#### ▼移動游標位置

無論在 command mode 或 input mode，都可以使用方向鍵來移動游標所在的位置，但若你的方向鍵不能使用的時候，則必須切換而 command mode 再配合下列幾組按鍵命令來移動游標：

h	游標往左移動一格
l	游標往右移動一格
j	游標往下移動一格
k	游標往上移動一格
0	將游標移動該行的最前面
\$+A	將游標移動該行的最後面
G	將游標移動最後一行的開頭
w 或 W	將游標移動到下一個字
e	將游標移動本單字的最後一個字元。如果游標所在位置為本單字的最後一個字

	元，則會跳到下一個單字的最後一個字元。標點符號 "."、"、"、"/"等字元都會被當成一個字。
b	將游標移動本單字的第一個字元。
^b	將游標向前面捲動一頁，相當於按 Page Up 鍵
^f	將游標向文章後面捲動一頁，相當於按 Page Down 鍵
^u	將游標向前面移動半頁
^d	將游標向後面移動半頁
^e	螢幕向下捲動一行，也可以說是文章和游標向上捲動一行
^y	螢幕向上捲動一行，也可以說是文章和游標向下捲動一行

### ▼複製文字

複製文字可以省下重複輸入的時間，vi 也提供了以下操作按鍵：

yy	連續按 y、y 鍵，可將游標目前所在位置的整行複製，複製時並不會反白顯示。
yw	y、w 鍵可複製游標所在位置到整個單字結束的字元
nyw	若按 3、y、w 鍵，則會將目前游標所在位置到單字結束，以及後面的 2 個字，共三個字（n 表示要複製的單字數目）
nyy	若按 3、y、y 鍵，則會將目前游標所在位置的一行，以及下面的 2 行一起複製（n 表示要複製的行數）
p	可將複製的字貼在目前游標所在的位置。如果複製的文字是整行文字，則會將整行內容貼在游標所在位置的下一行。

### ▼刪除文字

d	先按 d 鍵放開後， 按 鍵，可將游標位置的前一個字元刪除。 按 鍵，則會將游標位置的字元刪除。 按 鍵，可將目前游標所在的行與前一行一併刪除。
dd	連按再次 d 鍵，可刪除游標所在的那一行。所以千萬不要按著 d 鍵不放，因為這樣會連著刪除好幾行。
dw	將游標停在某一個字的第一個字元，按 d、w 鍵時，會將此字整個刪除。如果是停在一個字的中間字元上，則會將此字在游標後面的字元刪除。
nd	n 為數字，若按 3 d，表示標示游標所在位置的上或下方 3 行。 若按 鍵，則刪除游標上方 3 行，再包括本身 1 行，共刪除 4 行。 若按 鍵，則刪除游標下方 3 行，再包括本身 1 行，共刪除 4 行。
ndd	若按 3 d d 鍵，表示刪除目前游標位置的 1 行與下面的 2 行。
D	可刪除一行中，游標所在位置之後的所有字元。
x	刪除游標所在位置的字元，用起來比 d 方便多了
X	刪除游標所在位置前面的一個字元，與 d+ 鍵一樣
nx	按 4 x 鍵，表示刪除游標所在位置後的 3 個字元，與游標本身所在的字元



nX	按 4 X 鍵，表示刪除游標所在位置之前的 4 個字元(不包括游標本身所在的字元)
----	---

#### ▼顯示游標所在行數、移到指定行數

^g、^G	按此組合鍵，則會在最下方的 last line 處顯示游標所在位置的行數，以及文章的總行數。
nG	n 為數字，若按下 20 G 則可將游標移到第二十行

#### ▼其他按鍵

r	修改內容不一定要在 input mode，在 command mode 中只要將游標停在要改的位置，然後按 r 鍵，就可以改掉一個字元，而改完後仍然在 command mode。
u	按此鍵可取消前一次的動作，再按一次就回來
ZZ	連按兩次 Z 鍵，可將目前編輯的檔案儲存並結束 vi
%	當我們在寫程式時，常會用到大小括號，當找不到位置時，就可以將游標移到其中一個括號處，按此鍵可以找到對應的括號。

#### ▼ Last line mode 的操作

要進入 Last line mode，請在 command mode 按 : 鍵，此時游標會移到螢幕的最下面一行。Last line mode 的按鍵就少多了，請看下面的說明：

w	將檔案儲存。若先前沒有指定檔名，則要輸入 ":w filename"
q	結束 vi，但當檔案有修改時，並無法執行
q!	強制結束 vi，並且不存檔
wq	將檔案儲存，並離開 vi
e	若要在 vi 建立一個新檔，則輸入"e"加上指定檔名，例如"e newfile"。
n	若要在 vi 中載入另外一個檔案。則可輸入"n"再加上指定的檔名，例如：":n file.txt"
set nu	在檔案內容每一行加上行號":set nu"
set nonu	將每一行的行號取消
搜尋字串	要搜尋文章中的某一個字串時，可輸入"/"。例如：用"/gz"可找到游標位置後面第一個出現的"gz"字串，然後再輸入"n"可以去找下一個"gz"字串。
替換字串	當我們要將文章中某一個字串全部換掉時，則更以使用":g/temp/s/tmp/g"，如此一來，會將所有的 temp 字串替換成 tmp 字串。 如果不是所有的 temp 字串都要換掉時，可用":g/temp/s/tmp/gc"。它會將找到的每一個"temp"字串用其他顏色顯示，並在螢幕下方出現"replace with abc (y/n/a/q/^E/^Y)?"的詢問字串，可按"y"，即替換，並繼續往下找另一個"temp"字串

★如果，你真的不知道目前在哪一個模式下，可按 Esc 鍵，便會回到 command mode 模式下了。

## 伍、使用 tar 壓縮工具

tar 指令位於/bin 目錄中，它能將使用者所指定的檔案或目錄打包成一個檔案，但它並不作壓縮動作。一般都是先用 tar 打包後，再以 gzip 等壓縮指令壓縮成 xxx.tar.gz 檔案。  
tar 的指令相當多，可以執行 tar --help 指令得到參數及語法

常用的有：

-c	建立一個新的 tar 檔
-v	顯示運作過程的資訊
-f	指定檔案名稱
-z	呼叫 gzip 指令執行壓縮
-t	檢視壓縮檔案內容
-x	解開 tar 檔

一些常用的範例

tar cvf data.tar	將所有檔案打包成 data.tar
tar cvzf data.tar.gz	將所有檔案打包成 data.tar，再用 gzip 指令壓縮
tar tvf data.tar	檢視 data.tar 檔案中包括了哪些檔案
tar xvf data.tar	將 data.tar 解開

進階範例

tar --delete -vf text.tar uuencode.txt	刪除 text.tar 中 uuencode.txt 檔案
tar -rvf text.tar zip.new	將 zip.new 檔加在 text.tar 檔之後
tar -uvf text.tar *.txt	更新 *.txt 檔的內容
tar -xkvf text.tar	打開 tar 檔時，不覆蓋已存在的檔案
tar --remove-files -cvf text.tar *.txt	使用 --remove-files 參數將檔案移入 tar 檔案中，並移除原檔案
tar -Wcvf text.tar *.txt	寫入時檢查 tar 檔

呼叫壓縮指令

可以在 tar 檔產生之後，立即使用壓縮指令，省去輸入兩次指令的麻煩。

tar -xzvf foo.tar.gz	將 foo 解開至目前目錄之下
----------------------	-----------------

為了了解程式目前運作狀態，可使用以下的指令：

tar -tRvf text.tar

以下的指令可顯示寫入的 Bytes 數

tar --totals -cvf text.tar \*.txt

## 陸、壓縮及解壓縮的使用

並不是所有的程式套件都是以 RPM 套件的型式發行的，大部分都還是以壓縮檔的形式散布，所以常可見到.Z、.tgz、.ZIP、.gz、.bz2 等為副檔名的檔案。

### ▼zip 和 unzip

這兩個程式位於 /usr/bin 目錄中，它們和 DOS 下的 pkzip、pkunzip 及 Windows 的 WinZip 軟體功能一樣，可以將檔案壓縮成.zip 檔以節省空間。

#### ◎壓縮指令

zip	查詢版權及語法說明
zip myfile *.txt	將所有副檔名為 txt 的檔案壓縮成 myfile.zip 檔
zip myfile *.txt *.doc	將所有副檔名為 txt 及 doc 的檔案壓縮成 myfile.zip 檔
zip -g myfile *.txt	將副檔名為 txt 的檔案附加到 myfile.zip 中
zip -g myfile . -i *.txt -i *.new	將不同副檔名的檔案附加到 myfile.zip 中
zip -f myfile *.txt	更新壓縮檔中的所有*.txt 檔
zip -u myfile *.txt	更新修改過的檔案並加入新檔

#### ◎解壓縮指令

unzip 可以將 zip 所壓縮產生的檔案解壓縮。

unzip myfile.zip	將 myfile.zip 全部解壓縮
unzip myfile.zip -d mydir	建立 mydir 目錄，並解壓縮到此目錄
unzip -u myfile.zip	如原來的檔案已存在目錄中，就不解壓縮；反之，才解壓縮
unzip -j myfile.zip	解壓縮時不按原來的目錄結構，而將檔案置於目前目錄

#### ◎壓縮檔案的 gzip 指令

gzip -h

可得到指令的參數及語法說明。

gzip data.txt	壓縮檔案，不加任何參數
gzip -d data.txt.gz	將 data.txt.gz 解壓縮
gzip -S .gzip data01.txt	以.gzip 為副檔名
gzip -S .gzip -d data01*	解壓縮時，也必須以.gzip 為副檔名
gzip -d -f data02.txt.gz	強迫覆蓋舊檔，不會有詢問的訊息

#### ◎解壓縮 .gz 檔案的 gunzip 指令

使用者可以把 gunzip 視為 gzip -d 指令。其實它們是同一支程式，只是預設選項不同而已。

### ▼bzip2、bunzip2 和 bzip2recover

還有一種以\*.tar.bz2 為副檔名的壓縮檔，它就是以 bzip2 來壓縮的。通常拿來壓縮系統 kernel 或 patch 檔。

\*.tar.bz2 的解壓縮方法：

bzip2 -dc linux-2.1.xxx.tar.bz2 | tar xvf -

bzip2 這支程式位於 /usr/bin 目錄中

⊙修復 .bz2 檔的 **bzip2recover**

執行 **bzip2recover filename**

## 柒、排程指令：

電腦有很多程式是需要週期性地被執行的，例如：清理硬碟中的暫存檔、備份系統資料、檢查遠端郵件伺服器的信件……等

### ■排程的常駐指令

**crond** 是 Linux 用來定期執行程式的指令。**crond** 指令每分鐘會定期檢查是否有要執行的工作，如果有便會執行該工作。而 Linux 所排程的工作主要分成下列兩種：

- ◎系統執行的工作：如備份系統資料、清理暫存檔等
- ◎個人執行的工作：某個使用者定期想執行的工作，如每十分鐘檢查郵件伺服器是否有新信。

### ■排程設定檔的寫法

格式如下：

Minute Hour Day Month DayOfWeek Command

其所代表的意義如下：

欄位	所代表的意義	可用的範圍
Minute	每個小時的第幾分鐘執行該程式	0~59
Hour	每一天的第幾小時執行該程式	0~23
Day	每個月的第幾天執行該程式	1~31
Month	每一年的第幾個月執行該程式	1~12 亦可用 jan、feb 等英文的月份表示法
DayOfWeek	每一週的第幾天執行該程式	0~6 (0 代表星期天，亦可使用 sun、mon 表示)
Command	指定要執行的程式	填入要執行的指令，可加上執行指令時所需的參數

其中，除了 Command 是每次都必須指定的欄位以外，其他欄位可視需求決定是否要指定。對於不指定的欄位，請填上 "\*" 即可。

1 * * * * program -a -b -c	指定每小時的第一分鐘執行 program 指令
20 4 * * * program	指定每天的四時二十分執行 program 指令
50 3 9 * * program	指定九日的三時五十分執行 program 指令
1 0 8 9 * program	指定每年的九月八日零時一分執行 program 指令
1 0 8 sep * program	同上
20 2 1,15 * * program	指定每月的一日及十五日的 2:20 執行 program
25 1-4 * * * program	指定每天的 1:00 至 4:00 的第 25 分鐘執行 program
*/10 * * * * program	每小時的第 0、10、20、30、40、50 分執行指定指令
35 5 */10 * * program	指定每月隔 10 天的 5:35 分執行一次 program

### ■排程的系統工作

/etc/crontab 是 Linux 系統工作的排程設定檔，其內容如下：

## ■排程的個人工作

一般使用者可以利用 `crontab` 指令，自行設定要定期執行的工作。

### ▼使用者新增排程工作

`crontab -e`

執行此指令後，會進入 [vi 文書編輯器](#) 中，接下來即可自行編輯排程的工作。

以下的範例：

`55 14 * * * ls /home/terry`

`crond` 指令將會在 14:55 執行 `terry` 使用者的指定的指令後，便會將輸出結果或錯誤訊息，以 `E_mail` 的方式寄送給使用者。

### ▼使用者檢視排程工作

`crontab -l`

### ▼使用者刪除排程工作

`crontab -r`

### ▼系統管理者可管理所有使用者排程的工作

`root` 系統管理者有權力管理一般使用者的排程工作

例如要編輯 `terry` 使用者的排程工作：

`crontab -e -u terry`

`-u` 參數：指定要管理的使用者名稱

### ▼個人排程設定檔擺放的位置

會被儲存在 `/var/spool/cron` 目錄中，以帳號名稱為檔名的設定檔。如 `terry` 使用者的設定檔即為 `/var/spool/cron/terry` 檔。但使用者並無權限可以讀取該檔，只有 `root` 可以讀寫它。

### ▼排程程式的輸出結果

`crond` 指令預設會將執行的結果及錯誤訊息，以 `E_mail` 的方式寄給要求執行的使用者。你如果不想收到這些郵件，可以加入以下內容，將執行結果導向到一個記錄檔：

<code>1 * * * * program &gt;&gt; /home/terry/cron.log</code>	將執行結果導向到一個記錄檔
<code>1 * * * * program &gt;&gt; /home/terry/cron.log 2&gt;&amp;1</code>	將執行結果及錯誤訊息導向到一個記錄檔
<code>1 * * * * program &gt; /dev/null 2&gt;&amp;1</code>	不儲存執行結果及錯誤訊息

## 捌、 RPM 套件管理指令

RPM 全名 Red Hat Package Manager，它是 Red Hat 公司所開發的一個開放性程式管理套件系統。

### ■使用 RPM 優點

1. 易於安裝、升級與移除套件
2. 有強大的查詢功能
3. 能夠進行套件認證
4. 支援原始碼形式的發行套件

### ■RPM 指令概念

RPM 也可以在其他 Linux 及 UNIX 平台上執行。

通常它以 xxx.rpm 的格式命名，如 rhviewer-3.10a-13.i386.rpm，它包含了套件名稱：rhviewer

版本：3.10a

次版本：13

平台：i386

不過，並不是所有套件都是如此命名。

rpm 基本操作模式：安裝、移除、升級、查詢、驗證

利用 rpm --help 可以查詢指令細節

另外，**GnoRPM** 是 RPM 圖形化使用者介面，它還可以在網路上尋找套件是否有新的版本。

安裝、移除、更新套件，必須取得 root 權限

查詢、驗證套件則不必。

### ■RPM 套件安裝

```
ncftp 163.26.111.1
```

```
cd pub
```

```
ls -la
```

```
get svga*
```

```
get jmcce*
```

```
exit
```

```
rpm -ivh rhviewer-3.10a-13.i386.rpm
```

i：使用 RPM 的安裝程式

v：在安裝過程中提供更多資訊

h：在安裝過程中印出 # 號

⊙ rpm 指令還能透過 FTP 來進行遠距的安裝或查詢，只要檔案前加上適當的 URL 位址即可。（務必使用 -h 參數以了解安裝程序是否仍進行中）

⊙ 重覆安裝套件

如果某個套件已安裝過了，又重覆安裝時，會出現以下的錯誤訊息：

```
package rhviewer-3.10a-13.i386.rpm is already installed
```

如果使用者確定要安裝此套件，可以使用如下的指令：

```
rpm -ivh --replacepks rhviewer-3.10a-13.i386.rpm
```

⊙ 套件中某個檔案已安裝過了

如果套件中某個檔案已安裝過了，又重覆安裝時，會出現以下的錯誤訊息：

```
rhviewer /usr/bin/rhviewer conflicts with file from msviewer-1.10b-01 error :
```

```
rhviewer-3.10a-13.i386.rpm cannot be installed
```

如果要直接取代舊檔案，可使用如下的指令：

```
rpm -ivh --replacefiles rhviewer-3.10a-13.i386.rpm
```

⊙不同套件之間有關聯性

有些套件必須先安裝於系統中，才能正常安裝。常見的錯誤訊息是：

failed dependencies : rhviewer is needed by rhpainter-2.24-20

解決的方法就是將其它所需的套件先裝好，以此例來說，先安裝 rhviewer 套件之後，再安裝 rhpainter 套件即可。

亦可下 **--nodeps** 參數來略過相關性檢查，不過如此一來，可能會使套件無法執行。

## ■RPM 套件移除

移除模式會將指定的套件全部刪除，但不包括已更改過的設定檔。移除的指令如下：

```
rpm -e rhviewer
```

**【注意】**這裡必須使用套件的名稱"rhviewer"或"rhviewer-3.10a-13"，而不是當初的名稱"rhviewer-3.10a-13.i386.rpm"

## ■RPM 套件升級

升級會安裝使用者指定的更新版本，並移除已安裝於系統中的相同套件。指令如下：

```
rpm -Uvh rhviewer-3.10a-13.i386.rpm
```

在沒有舊版本的情形下，-U 這個參數仍可正常運作。

如果在升級套件時，發現有舊版本存在時，可以看到以下的訊息：

```
saving /etc/rhviewer.conf as /etc/rhviewer.conf.rpmsave
```

這是 rpm 指令特有的設定檔智慧升級訊息。使用者可以在安裝完成後，比較新舊設定檔藉以決定是否要換回原來的設定。

如果安裝的是一個比現有套件更舊的版本時，rpm 指令會顯示：

```
rhviewer package rhviewer-3.10a-13 (which is newer) is already installed
```

```
error: rhviewer-3.10a-13.i386.rpm cannot be installed
```

如果決心要安裝舊版本，可以使用如下的指令：

```
rpm -Uvh --oldpackage rhviewer-3.10a-13.i386.rpm
```

## ■RPM 套件更新

更新可以算是升級模式的其中一種使用方式。

指令如下：

```
rpm -Fvh rhviewer-3.10a-13.i386.rpm
```

在更新模式下，會檢查在指令列中所指定的套件，是否比系統中原來的套件來得新，如果是，rpm 指令會自動更新；反之，rpm 指令並不會安裝此套件。不同的是，在升級模式下，不管系統中是否有較舊的版本，rpm 指令都會安裝指定的套件。

⊙更新模式可以一次更新許多套件，假設我們從網路上已下載了許多套件，我們可以下：

```
rpm -Fvh *.rpm
```

## ■RPM 套件查詢

```
rpm -q rhviewer
```

在查詢模式中，可用**套件指定參數**配合查詢：

-a	查詢所有目前已安裝套件
-f 〈檔案〉	查詢包括〈檔案〉的套件
-p 〈套件檔名〉	以檔案名稱來查詢套件

也可以再加上**資訊選擇參數**來決定所要顯示的資訊



-i	顯示包含套件名稱、描述、大小、建立日期及其他雜項資訊
-l	顯示套件中包含的檔案列表
-s	顯示套件中所有的檔案狀態
-d	顯示在套件中被標示為文件的檔案列表
-c	顯示在套件中被標示為設定檔的檔案列表

## ■RPM 套件驗證

要驗證套件中的元件是否與原始套件相符，可使用驗證模式。-V 參數可以來驗證套件的正確性。

rpm -V rhviewer	驗證 rhviewer 套件的正確性
rpm -Vf /bin/vi	驗證包含/bin/vi 檔案的套件
rpm -Va	驗證所有套件
rpm -Va rhviewer-3.10a-13.i386.rpm	使用.rpm 檔來驗證套件的正確性

下表是可能列出的錯誤碼：

5	MD5 checksum 不合
S	檔案大小不合
L	符號連接錯誤
T	檔案修改時間不合
D	設備屬性不合
U	使用者不合
G	檔案群組不合
M	模式不合，包括權限和檔案型態

以下是常見的應用實例

rpm -ql xv	查詢此套件所安裝的檔案
rpm -Va	查詢是否誤刪了某些檔案
rpm -qf file-name	遇到不認得的檔案，查詢它到底屬於哪一個套件
rpm -Vf /bin/vi	進行 vi 這個程式的驗證

## 玖、帳號管理

### ■設定 root 帳號之密碼

變更 root 的密碼，可以透過 `linuxconf` 這個圖形模式的程式去更改。

`linuxconf` 有三種方法可以執行：

▼在 GNOME 桌面環境中，按左下角的主選鈕，執行「程式／系統／LinuxConf」命令

▼在 KDE 桌面環境中，請按 K 鈕執行「系統／LinuxConf」

▼在文字模式視窗內執行 `/sbin/linuxconf` 指令。

以下用文字模式為例來練習：

依步驟執行下列之指令 `Config/Users accounts/Change root password`

會要求輸入目前 root 之密碼，按 `Accept`

再按新的密碼，按 `Accept`

再按一次新密碼，按 `Accept` 確定

如果系統覺得該密碼可能輕易被人猜中，就會在輸入新密碼之後，出現一個警告訊息。

### ■變換身份為管理者

當管理者以一般帳號登入系統，隨後想轉變成管理者身份，進行系統相關設定時，並不需要重新登入，只要執行 `su` 指令，並輸入 root 帳號的密碼，即可轉換為 root 帳號。

```
[helen@free helen]$ su
```

```
Password:
```

```
[root@free helen]#
```

這樣只是將登入的身份轉變為 root，但諸如檔案路徑之類的環境變數仍然是原登入帳號 helen 的，例如在提示符號下執行 `shutdown` 指令，則會找不到指令，必須輸入完整的路徑 `/sbin/shutdown` 才行。若是想在變換身份並同時更改環境變數，則必須執行 `su -` 指令。

```
[helen@free helen]$ su - root
```

```
Password:
```

```
[root@free /root]# 完全轉變為 root 了！
```

倘若要切換成其他使用者，可以下 `su - helen` 即可

### ■只允許 root 登入

通常管理者在維護系統時，要讓其他的帳號都不能登入時，可在 `/etc` 目錄中，執行 `touch nologin` 指令，會產生一個檔名為 `nologin` 的檔案。當其他使用者登入時，系統只要發現這個檔存在，就會禁止他們登入。

如要再度恢復，則只要將 `nologin` 檔刪除即可。

### ■單人模式——忘記 root 密碼時的救星

如果只有一人使用系統，又不想每次啟動系統都經過登入、回答密碼的程序，則可以考慮使用單人模式（Single User Model）。

在開機顯示 LILO 畫面時，按 `Ctrl + X` 鍵，切換到文字模式的 LILO 畫面，並輸入以下的參數：

```
linux s
```

單人模式的字串和多人模式的字串不同

```
sh-2.04#
```

管理者的密碼忘記了，可以依下列步驟解決：

進入單人模式後，

```
sh-2.04# passwd -d root 取消 root 密碼
```

sh-2.04# passwd root    設定 root 密碼

## ■建立使用者帳號

### ▼在新建帳號的同時建立 html 及 cgi-bin 目錄

如果您有某些檔案，固定要給新建立的 user，可以將這些檔案，放入 `/etc/skel` 中，則建立帳號時，會自動拷貝至使用者的自家目錄中。

(skel 是 skeleton 骨架,基本的意思)

也就是說，只要在 `/etc/skel` 中先建個 html 目錄，在 html 目錄裡面再建個 cgi-bin 目錄，那你開設帳號時，這些都會一併拷貝過去的喔！

### ▼使用 useradd/adduser 指令建立帳號

在 Linux 系統中，不論執行 `adduser` 或 `useradd` 指令，其實都是指向 `/usr/sbin/useradd` 這個檔案，所以任選其中一個指令即可。

在一個系統中，使用者的帳號名稱（Login Name）必須是唯一的，假如要建立的帳號先被人佔用了，則會出現使用者已存在的訊息。

英文字母大小寫是有差別的，比如 `benny` 不等於 `Benny`，對於系統來說，它們是不一樣的帳號。

建帳號前，可以先檢查 `/etc` 目錄下的 `passwd` 檔案。是否該使用者是否存在。

另外一種方法，執行 `finger` 帳號指令，查看該帳號是否存在。

### ▼編輯 passwd 與 shadow 檔來建立帳號

假如一次要建許多帳號，用 `useradd` 指令，顯然過於麻煩，此時可以直接修改位於 `/etc` 目錄下的 `passwd` 和 `shadow` 檔案。

### ◆passwd 檔案的內容格式

`useradd` 所建立的帳號，實際上是儲存在 `/etc/passwd` 檔案中，修改此檔就可以新增帳號。以下是 `passwd` 檔案的內容格式：

`bin:x:1:1:bin:/bin:`

`daemon:x:2:2:daemon:/sbin:`

`adm:x:3:4:adm:/var/adm:`

`lp:x:4:7:lp:/var/spool/lpd:`

`sync:x:5:0:sync:/sbin:/bin/sync`

每個使用者帳號在 `passwd` 檔案中，都有 7 個欄位，由左到右分別用冒號（：）隔開。

1. 帳號名稱
2. 使用者密碼
3. 使用者識別碼
4. 群組織別碼
5. 使用者相關資訊
6. 使用者目錄
7. 使用者環境，Linux 預設是使用 `bash`

在預設的情況下，任何使用者都可以讀取 `/etc/passwd` 檔案，我們可以用 `ls` 指令來查看一下。

`/etc/passwd` 檔預設的權限是 `"rw-r--r--"`，表示所有人皆可讀取。因為 `passwd` 檔內容有 UID 與 GID 資訊，負責轉換擁有者和 UID 之間的關係，所以必須讓使用者能夠讀取。

我們不應限制 `passwd` 的讀取權限，而應該啟用投影密碼改善保密問題。。

### ▼shadow 檔案內容格式

由於 `passwd` 並不安全，因此在安裝 Linux 時，系統內定會啟動投影密碼功能。每當我們用 `useradd` 指令建立使用者帳號，系統不僅更動 `passwd` 檔，同時還會修改 `shadow` 檔。以下是 `shadow` 檔的部份內容：

```
sync*:11439:0:99999:7:::  
shutdown*:11439:0:99999:7:::  
halt*:11439:0:99999:7:::  
mail*:11439:0:99999:7:::
```

每個帳號在 `shadow` 檔中都有 9 個欄位，分別用 8 個冒號 ":" 隔開，以下依欄位分別說明：使用者密碼

1. 密碼最後變動的時間
2. 密碼變動兩次之間，至少需間隔的日數
3. 密碼變動後，距離下次一定要更改密碼的日數
4. 離下次密碼必須變動日期多少日，就開始警告使用者
5. 超過密碼必須變動日期多少日，就將該帳號停權
6. 帳號的使用期限
7. 最後一個欄位是保留欄位

#### ▼加入新的帳號記錄

我們可以用文書編輯器，將新的使用者帳號加入 `passwd` 和 `shadow` 檔案。

舉例來說，我們新增一個帳號名稱為 `silent`，

我們先利用文書編輯器在 `passwd` 檔案後面加入一行：

```
silent:x:600:100::/home/silent:/bin/bash
```

存檔後，接著編輯 `shadow` 檔案：

```
silent::11325::99999:::
```

這樣就產生了一個 `silent` 帳號了。

帳號名稱：`silent`

使用者識別碼：`600`

隸屬於 `users` 群組，群組識別碼為 `100`

使用者家目錄位於 `/home/silent`

採用 `bash` 做為使用者環境

沒有密碼

最後更動密碼的日期，距離西元 1970 年 01 月 01 日有 11325 天

隨時可更換密碼

每間隔 9999 天就必須更換密碼

系統不會警告使用者密碼逾期

就算密碼逾期也不會被停權

帳號沒有期限，可以永遠使用

最後請用 `passwd` 指令更改 `silent` 帳號的密碼，並建立 `/home/silent` 目錄，然後將設好的密碼通知使用者即可。

不過，在建立 `/home/silent` 目錄時，若單純用 `mkdir /home/silent` 指令，則會缺少許多預設的檔案。因此，筆者建議最好先建立一個樣板帳號，例如 `template`。如此在 `/home/template` 目錄中就會有 `/bash_profile`、`.bashrc` 等預設的檔案。然後再將樣板目錄複製一份給新帳號即可。

```
[root@dns /root]# useradd template 建立樣板帳號
```

```
[root@dns /root]# passwd silent 指定 silent 的密碼
```

Changing password for user silent

New UNIX password:

Retype new UNIX password:

passwd: all authentication tokens updated successfully

```
[root@dns /root]# cp -r /home/template /home/silent 將樣板帳號的目錄複製給 silent
```

```
[root@dns /root]# chown -R 600.100 /home/silent 將擁有者指定為 600，群組指定為 100
[root@dns /root]#
```

### ▼增加大量使用者帳號

當我們要新增大量使用者時，如果用 `useradd` 指令一個一個加入的作法，似乎缺乏效率。因此這邊介紹一個由縣內大師[臥龍小三](#)所設計的大量建帳號程式。

首先到縣內 FTP 主機去下載該程式：

[大量建帳號程式 1.0.1 版下載\(01/22/2001\)](#)

### ★安裝方法：

步驟 1：

將 `cmpwd101.gz` 解壓: (最好放在 `/root` 目錄中)

```
gzip -d cmpwd101.gz
```

步驟 2：

增加執行的權限：

```
chmod a+x cmpwd101
```

步驟 3：

在中文環境下(或以 **netterm** 遠端登入), 以 `root` 身份來執行:

```
./cmpwd101
```

以下是執行範例畫面:

```
#-----#
# 臥龍小三工具箱: ( RedHat Linux 專用 ) #
# 大量建帳號工具: cmpwd 1.0.1 #
# http://www.ols3-cgi.com 臥龍小三 CGI 天堂 #
# Copyright 2001 OLS3 All rights reserved. #
# 本程式僅授權供非營利之學術教育單位免費使用 #
#-----#
學生或其它身份文字代碼? (例: stu 或 tch 等)
stu
學生年級? (例: 1 代表一年級, 若是其它身份, 可按 Enter 不填)
3
起始學號? (例: 650, 若是其它身份, 可選用其它號碼或按 Enter 不填)
650
開始的使用者編號(uid)? 例: 600 (uid 觀念請您務必弄清楚!)
700
欲開設的帳號數量? (例: 100 代表 100 位使用者帳號)
10
自家目錄位置? 例: /home (按 Enter 預設為 /home)

密碼複雜度? (請輸入 1~6)
1: 帳號和密碼相同, 但不可用 telnet 登入主機
2: 帳號和密碼部份相同
3: 密碼統一
4: 簡易
5: 中度
6: 安全

建帳完成!
*****
請注意: 本程式已幫您完成以下幾件事情:
1. 建置帳號及密碼. 密碼均記錄在您目前路徑下的
newpwd.txt 中, 並加上建帳日期及帳號筆數.
```

2. 建立每個帳號的初始化隱藏檔。
3. 設定正確的目錄檔案權限屬性。
4. 開設網頁空間及 `cgi-bin` 目錄：  
`/home/帳號/html` 及 `/home/帳號/html/cgi-bin`
5. 設置簡易的網頁起始檔: `index.html`  
位於 `/home/帳號/html/index.html`  
觀看法: `http://貴校網址/~帳號/`

\*\*\*\*\*

說明：

本程式預設的使用者網頁空間目錄名稱為 `html`，若不是這個名稱，  
您應修改 `apache` 的 `srm.conf` 或 `httpd.conf` 中的 `UserDir` 設定：

`UserDir html`

然後，重新啟動 `apache`，即可生效。

所有建好的帳號密碼，會自動以附加的方式，記錄在 `newpwd.txt` 之中，  
各列資料以虛線隔開，您可以將它列印下來，再按虛線處剪開，  
然後，分發給申請帳號的學生保管。

**Bug 處理：**

若您有發現任何執行時的異常或錯誤現象，請和 [OLS3](#) 聯絡。

## ■停用與移除使用者帳號

當使用者畢業或離職，我們可以考慮停用或刪除使用者帳號，以避免其繼續登入系統。

### ▼停用帳號

將帳號停用的意思是暫時不允許使用者登入系統，而其資料仍要保留。我們可編輯  
`/etc/passwd` 檔案，把欲停用的該帳號 `silent` 標註起來：

`mysql:x:529:529::/home/mysql:/bin/bash`

`template:x:530:530::/home/template:/bin/bash`

`#silent:x:531:531::/home/silent:/bin/bash` 在最前頭加上 `"#"` 號

### ▼刪除帳號

當確定使用者已不再需要使用本系統或列為拒絕往來戶時，可考慮將該帳號完全刪除。  
使用 `userdel` 指令可以很方便地將使用者刪除，

`[root@dns template]# userdel -r silent`

加上 `-r` 參數，表示一併將該帳號的使用者目錄及郵件檔案（位於 `/var/spool/mail` 目錄中的  
同名檔案）都刪除。若不加，則只會刪除帳號而保留該帳號的相關目錄。

當然，我們亦可直接編輯 `passwd` 和 `shadow` 檔案，直接刪除其帳號，也請記得隨後馬上  
刪除該帳號之家目錄與郵件目錄。

### ▼刪除背景執行程式

為避免使用者還有程式遺留在系統中，請執行下列指令檢查背景執行的程式，並把背景  
中執行的程式刪除：

`[root@dns /root]# ps aux | more "silent"`

這個指令，會列出使用者 `silent` 的背景執行程式，可以看到類似的結果：

`silent 1802 0.1 0.9 1380 608 ? s 13:12 0:00 maybetrojan`

要刪除這些程式，可執行 `kill 1802` 指令。

### ▼刪除計時器

還有一點很重要，就是把使用者所設定的計時器去掉。在 `Linux` 系統中，使用者可以自行  
設定計時器，時間一到就會自動執行某些程式。這些計時器所執行的指令，有些或許  
會影響系統之安全與保密性，因此必須特別注意使用者所留下的計時器。

可執行以下指令檢查計時器：

```
[root@dns /root]# crontab -u silent -l
```

0,30 \* \* \* \* date 每當整點、30 分時，就會在背景執行 date 指令

當發現使用者自行設定的計時器還在系統中時，直接執行 `crontab -u silent -r` 指令，便能刪除該使用者的計時器檔案。

## ■自訂群組

我們可以使每個使用者擁有自己專屬的群組，也能將新增的使用者統統編到 `users` 群組。其實我們也可以依不同的性質自行建立群組，當檔案或目錄隸屬於不同群組時，群組的使用者也會受到群組權限的限制。

### ▼group 檔案的內容格式

系統除了 `root` 帳號的 `root` 群組，及一般使用者的 `users` 群組之外，還有許多其他的群組，詳細內容都記錄在 `/etc/group` 檔案中：

```
adm:x:4:root,adm,daemon
```

```
tty:x:5:
```

```
disk:x:6:root
```

```
lp:x:7:daemon,lp
```

```
mem:x:8:
```

```
kmem:x:9:
```

```
wheel:x:10:root
```

```
mail:x:12:mail
```

檔案內的每一筆記錄可分為 4 個欄位，由左到右分明用冒號（：）隔開：

1. 群組名稱
2. 密碼
3. 群組識別碼
4. 使用者帳號

### ▼建立群組

我們可以使用 `groupadd` 指令來建立群組，例如要建立 `GID` 編號 700，名稱為 `staff` 的群組：

```
[root@dns /root]# groupadd -g 700 staff
```

```
[root@dns /root]# more /etc/group
```

```
.....
```

```
sinbar:x:520:
```

```
etn2::526:
```

```
etn3::527:
```

```
etn:x:528:
```

```
mysql:x:529:
```

```
template:x:530:
```

```
staff:x:700: 已加進來了
```

`-g` 參數用來指定群組編號，0~499 請保留給系統使用。若省略此參數，則系統會自動指定 `GID`，從編號 500 開始找尚未用掉的號碼。

### ▼變更所隸屬的群組

已存在的使用者帳號，若要更動所隸屬的群組，請直接編輯 `passwd` 檔，變動各使用者隸屬的群組。並用 `chown` 指令，改變使用者家目錄及檔案所隸屬的群組。

```
[root@dns /root]# chown -R lambert.cassia /home/lambert
```

將 `lambert` 帳號的家目錄與目錄下的所有檔案，其隸屬之群組皆改為 `cassia`

### ▼刪除群組

```
groupdel staff
```

當某個群組被刪除後，請記得用 `find / -group staff` 指令檢查系統，將原本隸屬於被刪除群組的目錄及檔案，用先前提到的 `chown` 指令，改變其所屬群組。

**參考資料：**

1. <http://www.study-area.net/>
2. 台南縣莊士毅老師的 Linux 研習講義，<http://s7.tnc.edu.tw/~linux5/>
3. 台南縣王勝忠主任的 Linux 研習講義，<http://s7.tnc.edu.tw/~linux6/>
4. 高雄市邱志忠老師的 RedHat 筆記，<http://www.hmes.kh.edu.tw/~jona/>



# Linux 系統備份、轉移與重建

高雄市政府教育局  
資訊教育中心  
網路組 張宏明  
更新日期：2006/10/28

不管是 Windows 還是 Linux，甚至是 Unix 作業系統，拿來當 Server 服務人群的話，只要是提供服務，備份與資料的重建就變成了日常維護工作的重點。

要特別注意的是，愈是重要的資料，備份的份數與存儲的方式就要愈多種，以免常您要回復或重建系統時，才發現所備份的資料是壞掉的！

一個基本的常識，不要將備份的資料放在同一個媒體上，要不然，媒體遭到破壞或自然損壞時，備份的資料也隨著系統一併消失了！

## 一、系統的備份：

1. 備份系統中的重要資料，是最重要的一件事，一方面可解空間不足的硬碟空間內，有個基本的保障，但是要注意備份的資料，不可跟資料來源放在同一顆硬碟上。
2. 您可用 **tar** 命令來將要備份的目錄給壓縮起來，再放至磁帶機或燒錄燒成光碟片。
3. 在 windows 平台上找一顆大硬碟，作為備份資料的儲存所在
4. 在這顆大硬碟上建立一個新資料夾，並設成網路分享；記得要設定密碼！
5. 檢查 Linux 系統是否安裝 **samba** 套件，若沒有安裝，則請先安裝：  
`# yum install samba-common samba-client`
6. 建立一個目錄，作為掛載 Windows 分享資料夾的掛載目錄  
`# mkdir /mnt/backup`
7. 將 windows 的分享資料夾掛載到系統目錄來：  
`# mount -t cifs -o username=帳號,password=分享夾密碼,icharset=utf8,codepage=cp950 //windows 電腦名稱/分享夾名稱 /mnt/backup`  
[說明]:網路分享資料夾的檔案格式，對 Linux 而言有兩種:cifs 及 smbfs,cifs 為 Linux Kernel 所支援，它可支援單一檔案超過 2G 的限制；而 smbfs 為 samba 套件所支援，但其單一檔案大小限制在 2G 以內。就備份功能而言，對於大型伺服器而言，備備檔超過 2G 是常有的事，所以在此選定檔案格式為 cifs。
8. 利用 **tar** 命令，將需要備份的目錄，壓縮儲存到 /mnt/backup 中  
`# tar czvf /mnt/backup/home.tar.gz /home`
9. 將 windows 的分享資料夾從系統目錄中卸載：  
`# umount /mnt/backup`
10. 完成 Server 端的備份
11. 定期將 Windows 分享資料夾內的備份檔案，用 DVD-R/RW 或 CD-R/RW 將檔案資料燒錄成光碟妥善保存
12. 另外您也可以寫一個備份的 script，放入系統自動排程中，讓系統自動去執行：  
建立一個備份的執行 script 檔(backup.sh):

```
#!/bin/sh
LANG=zh_TW.UTF-8
export LANG
```

```

ser=`date +%Y%m%d`
mount -t cifs -o username=帳號,password=密碼,iocharset=utf8,codepage=cp950 //電腦名稱/分享夾名稱 /mnt/backup
sleep 10
mkdir /mnt/backup/$ser
mkdir /mnt/backup/$ser/mail
mkdir /mnt/backup/$ser/mail/home
cp -f /root/backup.sh /mnt/backup/$ser/mail/
cp -f /root/crontab.mail /mnt/backup/$ser/mail/
tar czvf /mnt/backup/$ser/mail/home/office-admin.tar.gz /home/office/admin
tar czvf /mnt/backup/$ser/mail/home/office-work.tar.gz /home/office/work
tar czvf /mnt/backup/$ser/mail/home/score.tar.gz /home/score
tar czvf /mnt/backup/$ser/mail/home/httpd.tar.gz /home/httpd
tar czvf /mnt/backup/$ser/mail/home/focalmail.tar.gz /home/focalmail
tar czvf /mnt/backup/$ser/mail/home/imp.tar.gz /home/imp
tar czvf /mnt/backup/$ser/mail/home/samba.tar.gz /home/samba
tar czvf /mnt/backup/$ser/mail/etc.tar.gz /etc
tar czvf /mnt/backup/$ser/mail/var.tar.gz /var
umount /mnt/backup

```

將備份執行檔加入系統自動排程中：

***crontab -u root -e***

加入以下這列設定

**10 2 \* \* \* /root/bakcup.sh**

## 二、系統의完整備份：

若您希望在另一顆硬碟完整備份一個一模一樣的系統，以備不時之需，可以最短時間內重新開啟系統，您可參考以下作法：

1. 假設您的第二顆硬碟是/dev/sdb。
2. **# fdisk /dev/sdb**  
分割一個可以容納得下您要備份的 partition 的空間。
3. **# mkfs.ext3 /dev/sdb1**
4. 將新硬碟掛到系統目錄來  
**# mkdir .backup**  
**# mount -t ext3 /dev/sdb1 /.backup**
5. **# ls**  
autoupdate boot etc initrd lost+found mnt proc sbin usr bin dev home lib misc opt root tmp var
6. **# tar clfp - Sys\* b\* d\* e\* h\* lib m\* o\* r\* s\* t\* u\* v\* / (cd /.backup ; tar xlpf -)**  
特別注意！！千萬不要 tar 到 /proc，還有 /lost+found 也不用
7. **# mkdir /.backup/proc**  
**# chown root:root /.backup/proc**  
**# chmod 555 /.backup/proc**
8. **umount /.backup**
9. 要特別注意，第二顆新硬碟尚未製作開機區，必須還要再製作一片開機片，以防止第一顆硬碟損壞時，換上第二顆硬碟時無法開機。  
**mkbootdisk --device /dev/fd0 2.6.8-1.521**

10. 換上第二顆硬碟時，必須將第二顆硬碟的 **ID Jump** 跳成第一顆，再利用開機片開機後，以 **root** 身份登入，再執行 **grub-install /dev/[hs]da** 命令將開機區寫入新的硬碟，如此才可在下次開機時由硬碟開機。

### 三、系統的轉移與重建：

系統要能順利的轉移或重建，在建立作業系統時，硬碟分割的時候，就考慮將系統與網站資料分開。

#### 備份：

1. 平時就要將 **/etc /var/log** 等與作業系統相關的設定檔予以備份。
2. 網站我們所建立的資料，則另外備份。  
如：**Web Server** 則 **Web** 文件(**/var/www/html**)必須備份，  
若開放 **user** 的網頁服務，則必須備份 **/home** 目錄。
3. 若有開放對外服務的話，則對每個人的個別資料，也必須分開備份。  
如：**mail server** 則內收(**/var/spool/mail**)與外寄(**/var/spool/mqueue**)的暫存區必須備份。
4. 資料庫部份，為考量版號相容問題，請盡量利用 **dump** 指令，將資料庫轉換成純文字 **SQL** 指令檔備份之；應避免直接備份資料庫之實體檔案。  
**MySQL** 部份，因 **3.xx** 與 **4.xx** 版資料庫內碼有大變動，**4.xx** 版使用 **mysqldump** 時必須多加入資料庫預設內碼：  
**3.xx** 版：**mysqldump -uroot -p -A**  
**4.xx** 版：**mysqldump -uroot -p -A --default-character-set=latin1**  
若您使用的 **4.xx** 版預設內碼不是 **latin1** 時，請依實際狀況改變您的備份參數。
5. 備份時，至少要作到不同之 **Server** 相互備份，可以的話，用一台專用的備份機器來備份會心較理想。

#### 轉移與重建：

每個版本的作業系統，均有其先天的缺點，漏洞也會日漸增加，所以不可能灌一次作業系統之後，就可永保平安，當這個作業系統漏洞出到您不能忍受時，就必須更新版本（通常一年一次），或是硬體也會在一定時使用時期後換新，如此均必須轉移您的系統到新的版本或新的硬體上來。

萬一不幸，網站遭受入侵，資料毀損，也必須將備份的資料分以回復，一般只要遭受入侵，通常會將作業系統整個換新重灌（就算沒有新版也要重灌），因為我們不知道入侵的對方，在我們的 **Server** 上到底做了什麼事，放了什麼後門。這時，就必須要整個系統重建。

不管是轉移或是重建系統，整個程序上而言，並沒什麼兩樣，不一樣的就只有造成的原因而已，所以我們統一來介紹。

1. 舊有 **user** 帳號密碼的轉移
  - (1) 找到您最新的 **/etc** 備份檔
  - (2) 找出 **passwd, group, shadow** 這三個檔案
  - (3) **password, group, shadow, gshadow** 這四個檔案中，**UID** 與 **GID** 小於 **500** 的帳號與群組刪除，並更名成 **password.old, group.old, shadow.old, gshadow.old**。

2. 重灌系統時，先 format 系統所在的 partition，再開始安裝新系統。
3. 系統完成後，先到所屬版本的 updates 去更新最新的套件。(參考「03.Redhat7.3 的昇級與補強」一文)
4. 更新完成後，將舊有的帳號、密碼與群組，匯入新的作業系統。
  - (1) # cat password.old >> /etc/passwd
  - (2) # cat group.old >> /etc/group
  - (3) # cat shadow.old >> /etc/shadow
  - (4) # cat gshadow.old >> /etc/gshadow
5. 依 Server 的原來屬性，將備份的相關 Service 設定檔放回新系統中。
6. 將其他的非系統資料(網頁資料、user 個人資料等)，放回新系統中。
7. 若有備份資料庫，必須將備份檔灌回 MySQL 資料庫中。特別注意，若 MySQL 版本由 3.xx 版轉移至 4.xx 版，必須注意預設的資料庫內碼的轉換；若您懶得做內碼轉換，則您必須在備份的 mysql dump 文字檔的最前端加上一行：  
SET NAMES latin1
8. 因為 OS 的中文語系有兩種，BIG5(zh\_TW.Big5)及 UTF8(zh\_TW.UTF-8)，若要將舊有的檔案系統(BIG5)轉換到新檔案系統(UTF8)上時，最重要的是中文檔名的轉換。您可利用 convmv 命令進行中文檔名的轉換。
9. 重新開機。
10. 檢測各項 Service 是否正常。
11. 完成轉移或重建。