Saving Space mini-HOWTO

作者: Guido Gonzato, <mailto:guido@ibogeo.df.unibo.it>
譯者: Jing-Jong SHYUE, <mailto:shyue@sonoma.com.tw>
v1.0.1, 7 April 1999

這份 mini HOWTO 提供盡可能減小 Linux 安裝所需空間的方向, 特別針對筆記型電腦的使用者.

目錄

1 簡介

我買了一台筆記型電腦, 而除了預先安裝好的 Windows 之外, 我想把 Linux 也裝上去. 我把 Windows 所使用的分割區減小到 500 MB, 如此我有 240 MB 的空間可以給 Linux 使用. 我必須使用一些技巧, 並儘量減少空間的需求, 才能把基於 RedHat 4.1 的完整功能系統安裝進去.

就算你不需要更多的空間, 但你也能在這份文件中找到釋放硬碟空間的方法. 但我要說在前面的是: 如果有任何問題發生, 請不要怪我!! 我將使用一些本質上有危險的方式, 雖然我用的很好. 我已經警告過你了!!

2 軟體需求

你必須要有:

- 一個完整可運作的 Linux 系統 (任何版本應該都可以);
- 一般的 gzip 壓縮工具程式, 或是 bzip2 壓縮程式. 你可以在 <ftp://sunsite.unc.edu:/pub/Linux/utils/compress> 找到這些工具; bzip2 的壓縮率比 gzip 要好, 但速度較慢且需要較多的記憶體.
- upx 執行檔壓縮程式, 他的網頁在 <http://cdata.tvnet.hu/~ml/upx.html> 或者 <http://wildsau.idv.uni-linz.ac.at/mfx/upx.html>;
- zlibc 套件, 你可以在 <ftp://sunsite.unc.edu:/pub/Linux/libs/compression> 找到. 檔名是 zlibc-X.X.tar.gz, 而 X.X 版本序號.

還有一些其他的執行檔壓縮器, 例如 gzexe (別用這個); 與在 upx 之前最好用的 tcx; 另外還有 tzx, 理論上比 tcx 好用; 但他把我的系統弄亂了, 雖然我不確定我是否犯了錯誤. 你最好還是用 upx, 因爲他可信賴, 有效率, 而且容易使用.

另外有一個核心修補檔讓你可以使用如 Stacker 等程式所壓縮的硬碟, 但是對於資料的寫入穩定性並不是很好. 爲了安全起見, 最好別用.

3 程序

3.1 移除核心原始檔

核心的原始檔會佔去約 20 MB 的空間, 所以你應該會想移除他. 在你移除之前, 建議你先建立一個針對你機器組態專用的核心檔.

我 不建議 把核心原始檔移除, 除非你確定你的機器已經設定完成. 此外, 當你要編譯程式時, 你也需要核心的 #include 檔.

如果你決定要移除核心原始檔, 不要 移除掉 include/linux 這個路徑, 除非你以後不在這台機器上編譯程式.

3.2 刪減應用程式

首先決定有哪些程式是你 眞的 需要的. 哪些則是多餘的: 例如你覺得你不能沒有 emacs 嗎? 你也可以改用 jed.

哪些程式應該要保留是由你決定的, 下面是一些一般性的要點:

- gcc 是非常大的套件. 但卻是編譯核心與任何原先沒有的應用程式所需要的. 當然, 如果你自行撰寫 C 或是 Fortran 程式時, 其中的 f2c 或 g77 也是必須的. 依據你的狀況來決定是否要移除他們. 而其他的編譯器如 lcc 也還不錯, 只是沒有 gcc 高階, 我想你可能會留著他.
- X11 是嚇人的大, 但有他是一件好事. 如果你決定不能沒有他, 至少作下面的事情: 只留下必要的 X 伺服器, 使用一個簡單的視窗管理介面, 只留下 xterm, 不要安裝 100 dpi 的字型等等.
- TeX 與伴隨的套件也相當的大. 放棄 LaTeX 而使用單純的 TeX 能省下很多的空間; 如果你使用 dvitty 或 dvivga, 則你可以放棄 X11 下的預覽程式, 例如 xdvi 與 ghostview. 除去這些 X11 的軟體, 也能減少 X11 所需的空間.
- 遊戲軟體是不需要的

3.3 壓榨二進位檔

我們可以開始減小二進位檔所佔的空間. 首先將檔案搬到 /usr/bin, 並使用這個指令

```
machine:/usr/bin# strip *
```

這會將所有嵌入於二進位檔中的符號移除. 你可以在 /usr/X11R6/bin/ 與其他包含有執行檔的目錄下執行這個指令. (別忘了找 TeX 與 gcc 的執行檔) 但是 不要 在 /sbin, /bin, 或 /usr/sbin/ 下面使用這個指令, 以免影響系統.

3.4 利用 upx 壓縮執行檔

首先安裝 upx 並閱讀相關說明文件. 然後切換到 /usr/bin, 並且執行 upx *; 他會把所有的執行檔壓縮, 包括有設定 suid 的檔案也會被壓縮 (tcx 不會壓這種檔案). 如同前面一樣, 在所有目錄下執行這個指令.

當你安裝了新的程式套件後, 記得把他壓起來.

3.5 把其他東西用 Gzip 壓起來

其他還有一堆檔案無法一次壓起來. 首先我們由 /usr/doc/ 開始. 移動到這個路徑後, 使用下面的指令:

```
machine:/usr/doc# find . -type f -exec gzip -9 {} \; 2> /dev/null
```

當你安裝了新的程式套件後, 記得把他的說明檔壓起來.

接著在包含 TeX 說明文件的目錄下重複上面的壓縮指令, 對我的系統而言, 這些文件是存放在 /usr/lib/texmf/texmf/doc/. 如果你希望的話, 也可以把這些東西直接殺掉.

接著安裝 zlibc, 並編譯他. 如果你的系統和我相似的話, 編譯的程序會因爲找不到靜態連結的 libc 而被中斷. 不用管他, 你會找到一個叫做 uncompress.o 的檔案, 這就是我們要的了. 把這個檔案移到 /usr/local/lib/ 下面, 並且把下面這行加到 /etc/profile 中.

```
export LD_ELF_PRELOAD=/usr/local/lib/uncompress.o
```

現在除了文件檔可以用 gzip 壓縮外, 其他的資料檔也可以被壓縮. 相關的程式應該還是可以存取這個資料檔. 理論上這個方法可以用在各種程式上.

4 一個實例

這是我在我的機器上用上面的方法的結果, 根據 df 的報告, 在開始減肥前, 我用了 398,798 個以 1024 位元組爲單位的區塊.

- 我沒有移除核心原始檔與核心標頭檔.
- 我把一些程式與遊戲給反安裝掉, 但我留下了 X11, X11 程式發展套件, C 與 Fortran 發展套件, Tcl/Tl, 網路工具, 和其他一些標準程式. df 回報我用了 244,668 個區塊.

- 我在 /usr/bin, /usr/X11R6/bin, /usr/lib/texmf/bin/i586-linux, 以及 /usr/lib/gcc-lib/i386-linux/2.7.2.1 下面使用 upx, df 回報我用了 226,270 個區塊.
- 我把 /usr/doc 與 /usr/lib/texmf/texmf/doc 下面的文件壓縮起來, df 回報我用了 198,745 個區塊.

總結而言, 我一開始用掉了 398,798 個區塊, 最後我只用了 198,745 個區塊. 想想看你在少於 200,000 個區塊的使用量下可以有多少東西! 如果我用 bzip2 而不是 gzip 的話, 我還能省下更多的空間.

平均而言, 如果一開始安裝時你只安裝必要的程式, 壓縮執行檔與說明檔, 可能節省 20 MB 的空間, 對筆記電腦而言, 這是個救命丹.

5 總結

5.1 版權聲明(此段不譯)

5.2 免責聲明(此段不譯)

"Saving Space mini-HOWTO"was written by Guido Gonzato, <REMOVE_MEguido@ibogeo.df.unibo.it>.

This document is provided "as is". I put great effort into writing it as accurately as I could, but you use the information contained in it at your own risk. In no event shall I be liable for any damages resulting from the use of this work.

Feedback is welcome. For any requests, suggestions, flames, etc., feel free to contact me.

Enjoy Linux and life,

Guido =8-)