Linux Remote-Boot mini-HOWTO: 使用 Red-Hat Linux, DOS, Windows 3.1 及 Windows 95/98 及 Windows NT 遠端開機工作站設定

作者: Marc Vuilleumier Stückelberg, David Clerc *mvuilleu@rembo.com*
譯者: 曾元佑 *yytseng@ms16.hinet.net*
v3.26, 二月 2000 翻譯日期: 六月 2000

這份文件將說明如何設定出一種, 使一群 PC 具有以伺服器爲基礎的穩定且安全的架構, 此法可使每台用户端 PC 在開機時選擇使用何種作業系統. 而此種架構的精要是使用以 BootPROM 爲基礎的程式, 讓使用者在開機時, 選擇不同的開機區塊. 我們是用 Bootix 的 TCP/IP BootPROM (可附加於大部份的網路卡), 或者其他與 PXE 相容的 Boot ROM (已在近期出現的內建網路卡的 PC 上使用). 此份文件的最新版本, 包含超文件連結到可供下載的軟體及其他相關的資訊, 都可以在這個網址找到 *http://cuiwww.unige.ch/info/pc/remote-boot/howto.html*. *Linuxdoc-SGML*, *DVI*, *PostScript* 及 *PDF* 等版本都在這個目錄下. 如果您想獲得最新的發展資訊, 可以送一封信給 *info@bpbatch.org*.

目錄

1 免責聲明及版權

在使用這些軟體前, 請仔細閱讀.

本節定義你對本軟體所能做與不能做的事, 及我們所擔保與補償的限制.

安裝, 複製, 或使用軟體都表示你已承認, 並讀過這個版權且已接受這些條文.

請注意這份版權只對個人的使用有效, 就如在通用條文的第 1 項所定義的.

通用條文

1. 授權執照

BpBatch, MrBatch, MrZip 及 相關工具程式 (以下指 "軟體") 的作者, 允許你有一份 不可獨占, 不可傳遞, 免權利金的執照, 用以下載並使用這些軟體但必須是在下列條文所約束下.

這份執照只授權個人的軟體使用. 是否視爲個人的使用, 乃是以軟體是放在個人, 公司或學校所擁有的電腦來認定.

軟體的散佈, 販賣或租賃, 不管是 "照原形式" 或當成產品的附件, 或把它安裝到屬於第三者的電腦上都是禁止的, 除非得到作者群的商業型態授權或他們所認定的代理人的授權.

請透過 *info@rembo.com* 聯絡以得到更多資訊. 我們提供您付得起的商業型態授權.

2. 免責擔保

這些軟體規定 "照原形式", 不具任何直接或間接的擔保. 也無法擔保或預期是否有任何錯誤.

使用這個產品所導致的結果與風險都必須由你承擔. 特別是, 作者群所提醒你的, 這個軟體能在沒有任何警告下, 把硬碟內容清除掉.

作者群不論在任何情況下, 都不需對你負責. 侵權行爲或契約, 在經濟上的影響或任何形式的非直接性損害, 包含未限制到的, 財務或獲利上的損失, 或者無法使用, 消失或損害到資料或導致中斷生意, 使用這些或其它軟體所導制或與其有關的問題, 甚至所建議的可能的損害以至預期發生的損害.

3. 版權

這些軟體是有版權的並受到國際法的版權法及條約所保護. 你必須同意避免任何非經授權的軟體複製. 你不可以把版權宣告或專利權, 自軟體中移去或使其朦朧不清.

你不可以任何方式對軟體做反編譯, 反組譯, 或逆向工程.

4. 條文可分割

這些通用條文的分割聲明. 如果這些條文中任一項目或規定, 因爲任何因素而失效或不具權力, 則其他項目或條文依然有效並具權力.

5. 適用法律

這些與此軟體有關的通用條文以及所有的權利或義務, 將一律在瑞士的法律下管理與釋意, 適用於 日內瓦.

6. 獨占司法權

任何涉及與這個軟體有的爭議都必須服從於瑞士日內瓦法庭的獨占司法權, 保留上訴到瑞士的聯合最高法庭.

2 有那些變更...

2.1 ...從 2.x 版後 ?

直接的說, 全部都改了. 只有基本的概念是相同的, 有關軟體的部份已全部重新設計以克服先前版本的一些限制. 新功能的重點有:

- 所有的功能 (BpMenu, BpClean, BpUnzip) 已整合在同一個程式.
- 此程式不只可以由 BootPROM 裡執行, 並可以在 DOS, Windows 95/98 及 Linux 下執行.
- 此程式現在已可以回存 FAT16, FAT32 及 EXT2FS 的分割區. 如果有人想寫支援 NTFS 的部份, 請讓我知道... 目前, NT 使用者還是得用 FAT16.
- 此程式不僅可以回存磁碟映像檔, 還可以附加或更新單獨的檔案用以改變用户端的功能特性.
- 磁碟映像檔不再有 87 MB 的限制. 他們現在是與檔案系統無關的程式.
- 我們打算讓它在同一時間内以自動的方式下載磁碟映像檔到一堆用户端 (採網路廣播的技術).
- 現在你可以自行撰寫更可靠的開機程序, 在眞正開機前它將決定機器所要的功能.
- 現在你可以用任何一種 Linux Kernel 開機, 不需再加上修正檔. 亦能加上命令列控制及 ramdisk 映像檔.
- 你可以採用 Unix, NT 或 Radius 伺服器做開機前的使用者身份認證, 並能拒絕非法使用者使用.
- 已包含完整的國際化語系支援.
- 還有許多, 許多其他新的功能...

是否有那個程式能把舊的映像檔格式轉換成新格式 ?

沒有, 因爲核心的格式根本就不相同. 但你可以輕易地自行轉換:

1. 以舊的映像檔開機 (把它解到你的硬碟)
2. 移掉叫用舊的 `unzipreg` 工具程式, 改以適當的 `patch` 指令 (參照以下的步驟, 它相當容易使用)
3. 執行新的 `MrZip` 程式以產生一個新格式的磁碟映像檔.

2.2 ...從 3.0 版以後 ?

3.0 版: 原本是一個測試版本. 在經過一個月的時間, 全球各地使用者的實際測試及尋找程式裡的臭蟲並提供更好的意見. 感謝他們所有人的耐心, 特別是 Maciek Uhlig, Dick Velders 及 Jeff Teeters.

3.01 版: 後開始增加一些細部的功能, 比如無磁碟機的 Linux 開機 (藉由關掉快取功能).

3.10 版: 開始與 Intel 的 *Wired for Management 1.1a* NetPC 標準相容. 這些工具程式已能在現今任何 PXE 相容的 BootPROM (它通常是直接附在主機板上的 BootPROM) 上工作. 感謝 *Bootix GmbH* 提供我們 PXE BootPROM 以發展這項功能. 而我們也成功地完成各項工具程式與我的 Dell 電腦上附於主機板的網路卡 (稱爲 LanDesk Service Agent) 的 PXE BootPROM 測試.

3.11 版到 3.12 版: 增加 UNIX 伺服器端工具程式 (可在 Solaris 及 Linux 下跑的 PXE Proxy DHCP 伺服器, 及一個增強型的 TFTP 伺服器給 Linux 執行), 更詳細的資訊以說明伺服器端的設定方法及 PXE 開機步驟.

3.13 版: 增加對 Advanced Power Management 的支援 (關電源的指令).

3.14 版: 增加細部功能強化及修正. 我們修正一個 RedHat 5.1 terminal 的問題, 及另一個關於 "if" 指令的語法問題. 我們增加一些由 Laboratori de C跳cul de la Facultat d'Inform較ica de Barcelona (LCFIB) 所建議的新功能:

- 一個新的 APM 變數使你知道系統是否支援 Advanced Power Management (即 支援關閉電源的指令).
- "beep" 指令.
- DrawWindow 的新參數, 可以在產生視窗同時加入標題. 你可以用 DrawWindow 200 200 400 200 "視窗標題".

3.15 版: 新增 VESA 的完整支援. BpBatch 現在支援多種顯示模式, 以讓那些不能顯示 800x600 圖形的舊電腦也可以使用. InitGraph 加入一個新參數以指明所使用的顯示模式, 並可由新的 VESA-Modes 變數取得所偵測出可用的顯示模式.

3.16 版: 修正下列的程式臭蟲:

- 當 Fullunzip 處理混亂複雜的影像檔時產生的 "Malloc failed" 訊息. 相當感謝 Christian Meyer 的協助.
- 在 MrBatch 的 Linux 版本有個程式臭蟲使其無法完整的把映像檔解壓縮, 這個程式臭蟲是出現在 MrBatch 的底層函式, 因此會一併修正其他 MrBatch 在 *Linux* 版本下的問題. 相當感謝 Jeff Teeters 的協助.
- 在 codepage 轉換表中的錯誤. 這個程式臭蟲由 Laboratori de C跳cul de la Facultat d'Inform較ica de Barcelona (LCFIB) 發現. 在 BpBatch 討論群組中, 你能找到此程式臭蟲的報告.

3.17 版: 新增的細部功能及修正的程式臭蟲:

- Fullunzip 程式會把延伸記憶體關掉.
- 以 RedHat 開機片開機, 已可以正常工作.
- 當所解開的映像檔含有太多目錄時, FAT 檔案系統會出現的問題已修正了.

- 我們加入以 TFTP 通訊協定傳送資料時的重發功能. BpBatch 會在 "Could not transfer the file" 出現前重送資料三次.
- Timestamps 可以正確地在 FAT 下更新. (感謝 Francis Chan)

3.18 版: 修正 IncrUnzip 功能上的程式臭蟲. 感謝 Gary Pike 的協助.

3.19 版: 修正在 Ext2fs 下 `delete` 指令的錯誤處理方式, 解決在 Linux 下對 A: 的不合宜處理. 並加入下列功能:

- 新的 `if valid 硬碟機:分割區` 語法用以檢查分割區是已否被格式化
- FAT32 磁碟映像檔現已功能完整 (已可正常開機).
- Linux EXT2 已可支援超過 2 GB.
- Linux Swap 分割區以可支援超過 128 MB (此功能需用 2.1.x 以上的 Kernel).
- FullUnzip 現在也可以不用快取分割區, 只要設定 `CacheNever` 為 `"ON"`. 這對每台用户端的安裝都不一至時還滿有用的, 但通常不建議使用, 因為這會使得網路負載加重.

感謝 Ruben Schattevoy 的協助及對此版本的貢獻.

3.20 版: 我們加入 Peter Schmitt 所寫的文件.

3.21 版: 我們已修訂本文的部份章節以吻合 PXE 2.0 的規範. 最主要的改進是 BpBatch 的命令列模式下參數設定改用 DHCP 選項 135 取代原來的 155 (155 仍然支援以向下相容). 3.22 版:引入新的版權條文. 請在使用 BpBatch 之前,詳細閱讀.

3.24 版: 由 Tellurian Pty Ltd, *http://www.tellurian.com.au* , 加入 Watchdog 介面卡的支援. 我們一並修正載入器對名稱的語意法則, 以允許 Intel 及 Bootware TFTP 伺服器的使用. 參考 what's new 檔以了解進一步的細節.

3.25 版: 我們加入一些小技巧 (給 Windows 98 使用的過濾器及如何取消密碼傳輸前加密功能) 來自 BpBatch 迷.

3.26 版: 修正許多印刷的錯誤, 感謝 Herman Bruninckx 長期反複的審閱. 我們也加入一節來說明 Solaris TFTP 伺服器.

3 簡介

這裡所描述的架構是從 1996 夏季開始在 日內瓦大學, CUI 所發展的. 這個資訊系有許多的伺服器, 及一些個人電腦, 可依用途分為兩類:

- 學生使用
- 研究與輔助教學

以下是我們發展這種架構的目標:

- 任何一台電腦都可以執行 Linux, DOS, Windows 3.1, Windows 95/98, 或 Windows NT. 每次開機時可以選擇所需要的作業系統.

- 所有的軟體, 包含作業系統, 可以放在伺服器上, 便於安裝與更新.
- 用户端電腦, 可以在不對伺服器做寫入的狀態下正常工作 (為了安全性), 除了用户自己的 home 目錄.
- 盡可能減少用户端電腦的設定. 用户端電腦可以自動由伺服器取得 IP 組態設定, 這些資訊放在一個小檔案, 可供各種作業系統使用.
- 大部份的電腦都各自擁有硬碟, 用户端電腦可以靠這優點來降低網路負載與存放使用者的暫存檔.
- 所有的使用者都 必須 登入才能使用電腦.
- 登入的資訊必須所有的作業系統都相同. 而使用者可以存取個人的目錄, 並而在各種作業系統都一至.
- 學生 (以及秘書 :-) 所使用的電腦在每次啓動時必須清理一次. 也就是像剛安裝完的樣子.
- 每台電腦可以防毒.

這些限制條件, 使得我們把所有的組態都以 BootPROM 工具程式爲基礎. 我們先是針對來自 *Bootix GmbH* 的 *TCP/IP BootPROM* 發展. 而現在更加入對 preboot execution environments 標準的支援. 我們讓這些工具程式也可以與任何 PXE 相容的 BootPROM 一起使用. PXE BootPROM, 也稱爲 LanDesk Service Agent, 已跟隨著大部份內建網路卡的主機板一起散布. 其他關於 PXE 及 Intel *Wired for Management* 的標準, 可透過 *http://developer.intel.com/ial/wfm/tools/index.htm* 取得.

3.1 BootPROM 及 硬碟機

BootPROM 已出現很長一段時間, 在此之前我們把它單獨用於無磁碟機的電腦(採用網路爲基礎的磁碟模擬或者下載一份 ramdisk). 從 1996 年起, 這份 How-to 提出使用 BootPROM 在自身具有硬碟機的電腦下使用, 將更有趣. 由於這樣將可得到雙重得優點:

- BootPROM 使得此架構更加穩定, 用它可以保證每台電腦都會以同樣方式開機. 不管中了任何病毒或在分割區已毀損的狀況. 就如我們已完成的, 它可以做到在載入作業系統前清理硬碟資料.
- 自身具有硬碟機, 使得此架構更有效率. 因爲這樣可以經由快取方式降低網路負載. 更能獲得較有效率的 swap.

至今, 我們相當高興見到所有的電腦製造商已朝向同樣的目標. 並把 BootPROM 列爲新電腦標準的一部份.

你仍然可以照 舊 的方式, 使用下列所敘述的工具程式. 這些工具程式對無磁碟機的電腦而言, 其實就是一個小的 Kernel/ramdisk 載入器. 但我們不鼓勵如此使用.

3.2 網路

日內瓦大學 擁有 B 級網域, 分割成許多子網路. CUI 共使用了四個子網路, 其中之一供學生使用.

原本,我們的個人電腦使用兩種網路通訊協定: IPX 與 IP. IPX 這一類, 我們由一台 Novell Netware 3 的伺服器共用 DOS 與 Windows的軟體. IP 這一類, 我們由一台 SUN 伺服器靠 NFS 共用 Linux 軟體及個人的分割區.

在我們最新的架構中, 已不再使用 IPX 通訊協定. 改由一台 Unix 伺服器 (可以是 Linux 或 SUN) 來擔任, Linux 用户端可透過 NFS, 而 DOS 與 Windows 則由 SMB (NetBIOS) over TCP/IP 共用軟體及使用者的檔案. 因此, 我們對所有的作業系統都使用單一個 home 目錄.

3.3 它是如何工作

1. 用户端個人電腦開機後, 在 TCP/IP BootPROM 或 PXE Boot ROM 獲得控制權之前先做自我測試.
2. BootPROM 發出 BOOTP/DHCP 要求, 以取得 IP 組態參數.
3. 如果伺服器收到個人電腦所送出的要求, 就會送回 BOOTP/DHCP 回應, 內容包括用户端的 IP 位址, 預設通訊閘道, 及開機映像檔. 否則, 伺服器會忽略這個要求.
4. 而在 PXE Boot ROM 的環境下, 將有更多的訊息在用户端與伺服器之間作應答, 用以決定安裝參數.
5. BootPROM 由 TFTP 通訊協定從伺服器下載開機映像檔, 這個映像檔其實就是一個名爲 `BpBatch` 的小程式, 也就是我們的開機時期批次檔案的解譯器.
6. 啓動批次檔解譯器之後. 幾乎是只有它單獨存在記憶體中. 除了 preboot execution environment (由 Boot ROM 提供)外, 這時作業系統尚未載入.
7. 這個批次檔解譯器會檢查 BOOTP/DHCP 給命令列的所有選項, 特別是 BpBatch 要處理的. BpBatch 以 DHCP 選項 155 取得命令列參數. 如果選項 155 是空的, 它會改由 DHCP 選項 135 取得參數 (PXE 2.0 BootPROM 只能收到 128-135 的選項).
8. 透過批次作業檔, 我們可以做一些事如:
   (a) 載入多國語系鍵盤對應.
   (b) 透過遠端伺服器 (Unix, Radius 或 Windows NT) 做使用者身份認證.
   (c) 讓使用者決定使用, 哪種可用的作業系統.
   (d) 決定作業系統後, 自動重新分割硬碟並快速格式化部份分割區.
   (e) 檢查其硬碟內的資料是否與其選定的作業系統的最新設定一制, 如果不是, 它會透過 TFTP 通訊協定下載最新映像檔.
   (f) 解壓縮所選定的作業系統的映像檔到主分割區.
   (g) 如果選擇 Linux, 載入 Kernel 並啓動它.
   (h) 如果選擇 DOS 或 Windows, 就更新硬碟資料並啓動電腦.

   對 **DOS and Windows 3.1**, 我們使用免費的 Microsoft LanManager for DOS (搜尋網路上與你最近的映射站台; 應該是有四個名爲 `disk1` 到 `disk4` 的檔案) 爲 SMB 用户端. Microsoft LanManager 可以透過 DHCP 動態設定組態. 在登入後, 使用者面對的是 DOS, 且只要打 `win` 指令就可啓動 Windows 3.1 了. 有一點, 在這要特別注意 DOS 及 Windows 3.1 看起來像是安裝在用户端的自身電腦.

   對 **Windows 95**98 及 Windows NT/, 我們也使用 Microsoft SMB 用户端 (稱爲 *Client for the Microsoft Network*), 它也支援透過 DHCP 動態設定組態. 爲了降低網路負載我們採用了 *Shared LAN Cache*, 這是一個強而有力的網路到硬碟之間的快取程式.

學生所使用的電腦可以很暴力的直接關掉電源而不會有任何問題, 因爲硬碟的資料在每次開機時都會重新建立.

對於比較 "安全的" 電腦 (如 助教的電腦), 如果電腦已用上述系統開機過一次, 則開機腳本只會改由本機硬碟開機, 而不再清理硬碟. 這樣可以允許使用者保留資料在硬碟中. 而當整個架構爛掉了, 使用者只需從開機選單挑選所要的系統, 則又能得到一個全新的安裝.

3.4 相關的非商業文件

這種組態已經成功的移殖到世界上的許多地方. 少數人已經開始篆寫一些技巧與提示來完成這份 How-To. 如果您也曾做過, 但在本文中未被引用歡迎送一份 e-mail 給 mvuilleu@bpbatch.org. 如果您在重建這種組態中遇到問題, 請再詳細看一次文件！

- *http://www.ph-ludwigsburg.de/nutzer/schmitt_peter/*, 由在德國, Ludwigsburg 的 Carl-Schaefer-Schule 的 Peter Schmitt 所寫. 此文是 BpBatch 的優質參考資料, 適合德語的使用者 .
- *http://www.br.fgov.be/RESEARCH/INFORMATICS/info/bootp.html*, 由在比利時國家植物園的 Alain Empain 所寫. 有許多有用的範例腳本程序,以及一個很棒的 PERL 程式用來自動產生圖形化的選單並能透過高階的敘述轉成 HTML 文件.
- *http://www.katedral.se/system/elevsyst*, 由瑞典 Uppsala 教會學校的 Johan Carlstedt 所寫. 至今, 在這的組態設定還停留在舊版的遠端開機工具程式. 大致上, 只要一點點修正, 所有的東西都還可以適用.
- *http://vitoria.upf.tche.br/~fred/*, 葡萄牙文, 由 Brasil 的 Passo Fundo 大學的 Frederico Goldschmidt.
- *http://www.etse.urv.es/~larinyo*, 西班牙文, 由西班牙, Escola Tecnica Superio d'Enginyeria 的 Lluis Arino 所寫.

你也可以寄給我你的 BpBatch 腳本程序檔, 如果你想讓它放在範例腳本程序精選中.

4 如何設定

首先, 調整機器位置使手邊有下列兩台機器:

- 伺服器, 我們是用一台 Unix 主機或 Windows NT.
- 用户端, 一台已經啓動 BootPROM 的個人電腦, 硬碟内不要有重要資料.

如果你想要測試您的組態設定而您還沒有 BootPROM, 你可以到 *http://www.bootix.com* 下載測試版磁片. 這個磁片可以使您的電腦就好像是插上 TCP/IP BootPROM 一般.

如果你已有 Boot ROM, 你要先啓動它. 如果您使用 Bootix 的 TCP/IP BootPROM, 你可以用網路卡製造商所提供的特殊程式來啓用它. 如果你用 PXE BootPROM, 你只要經由 BIOS 設定, 改變預設的開機裝置 (default boot device) 就可以了.

學生所使用的電腦, 我們設定由網路開機, 而不能透過硬碟及軟碟開機. 助教的電腦, 我們設定先由網路開機, 但也允許硬碟與軟碟開機.

4.1 伺服器端的設定

在伺服器端, 只需要下列的服務:

1. 一個 BOOTP/DHCP 伺服器.
2. 或許也要一個 Proxy DHCP 伺服器.
3. 一個 TFTP 伺服器.

使用 **PXE Boot ROM** 必須注意**:** 你要確定是使用最新版的 PXE. 在寫本文的時候, 最新的版本是 PXE 2.0 但 PXE 0.99x bootrom 也支援, 不過有部份的 DHCP 選項將不會透過 DHCP 伺服器送出. 特別是, 如果 DHCP 選項 135 對 BpBatch 失效 (BpBatch 於啟動時, 顯示 "Cannot load file"), 你就要升級到 PXE 2.0 . 如果你的網路卡屬於 Intel EPRO100 族系, 你可以取得最新的 Intel PXE Product Development Kit (PDK) 評估版, 它還包含一個用來更新 BootPROM 的快閃記憶體工具程式. 這個 PDK 放在 *http://developer.intel.com/ial/wfm/tools/pxepdk20/index.htm*.

4.1.1 設定 DHCP 伺服器

DHCP 伺服器所伴演的角色是傳送 IP 位址給用户端並讓它自 TFTP 伺服器下載, 名爲 `bpbatch.P` 的檔案. DHCP 是重疊於 BOOTP 的通訊協定. 如果你用 Bootix 的 TCP/IP BootPROM, 你可以不用 DHCP 伺服器 (用舊的 BOOTP 伺服器).

在 Windows NT, 你也需要類似的原生 DHCP 伺服器. 如果你用 Bootix 的 TCP/IP BootPROM, 你則要用一個小技巧來定義可開機檔的檔名 (可由 Bootix 的 WWW 網頁獲得更多資訊). 如果你用 PXE BootPROM, 你還需要一個用 DHCP 選項 67 定義可開機檔的檔名. 還要設定 VendorClass (DHCP 選項 60) 爲 "PXEClient" 及 VendorOptions (DHCP 選項 43) 爲 `01,04,00,00,00,00,ff` (參考下文).

在 Linux, 最佳的選擇是來自 網際網路軟體協會 (Internet Software Consortium) 的標準 DHCP 伺服器. 如果你用 PXE BootPROM, 則在常見的設定外, 你還要多加下面兩個設定:

- `option dhcp-class-identifier "PXEClient"`.
- `option vendor-encapsulated-options 01:04:00:00:00:00:ff;`.

如果你使用 ISC DHCP 3.0, 則語法上有些微出入. 以下是 DHCP 3.0 相容的設定檔:

```
# DHCP ISC 3.0 & BpBatch 用的 DHCP 設定檔
#

# 全域選項
option subnet-mask 255.255.255.0;
default-lease-time -1;

# 定義 PXE 規範的選項
# Code 1: 指到開機檔所在位置機器的 IP 位址
# Code 2: 用户端電腦監控 MTFTP 回應的 UDP 埠號
# Code 3: MTFTP 伺服器用來監聽 MTFTP 回應的 UDP 埠號
# Code 4: 用户端電腦在試著啓動一個新的 MTFTP 傳輸之前的等待秒數
# Code 5: 用户端電腦在試著重新啓動一個新的 MTFTP 傳輸之前的等待秒數
```

```
option space PXE;
option PXE.mtftp-ip    code 1 = ip-address;
option PXE.mtftp-cport code 2 = unsigned integer 16;
option PXE.mtftp-sport code 3 = unsigned integer 16;
option PXE.mtftp-tmout code 4 = unsigned integer 8;
option PXE.mtftp-delay code 5 = unsigned integer 8;

# 規範子網域的選項
subnet 192.168.1.0 netmask 255.255.255.0 {
  option routers 192.168.1.1;

# 規範主機的選項
  host pctest {
        hardware ethernet 00:01:02:03:04:05;
        filename "bpbatch.P";
        next-server 192.168.1.2;
        fixed-address 192.168.1.100;
        # BpBatch 命令列參數: -i == 交談式
        # 你也可以定義一腳本程式檔名 (不要把副檔名 .bpb 寫進去)
        option option-135 "-i";
        # PXE 規範的選項
        class "pxeclients"
        {
         match if substring (option vendor-class-identifier, 0, 9) =
                "PXEClient";
         option vendor-class-identifier "PXEClient";
         # 至少設定一個 vendor-specific 選項.
         # 我們設定 MCAST IP 位址爲 0.0.0.0 也就告訴 BootPROM 繼續使用
         # TFTP 通訊協定 (位址 0.0.0.0 意謂著 "沒有位址")
         option PXE.mtftp-ip 0.0.0.0;
         vendor-option-space PXE;
        }
  }
}
```

在 Solaris 上, 你依然可以使用 網際網路軟體協會 的 DHCP 伺服器 (可以在 Web 上找到), 或使用 Solaris 的 DHCP 伺服器 (Solaris 2.5 版以後都有附). 然而, 用 Solaris DHCP 伺服器似乎無法在 DHCP 設定中提供, 用户類別的辨識碼. 你必須安裝一個 Proxy DHCP 伺服器. 不過, 這個 Proxy DHCP 伺服器必須安裝在另一台電腦上, 因爲 Solaris DHCP 伺服器會鎖住 DHCP 埠.

我們建議您設定用户端電腦的 DHCP 參數爲無限制的使用週期. 別忘了 BOOTP/DHCP 的要求訊號受限於子網域內傳送. 如果用户端與伺服器不在同一個子網域你就必須安裝一個 BOOTP/DHCP Relay 代理程式在兩個子網域之間. 但現在, 先假設兩台機器都在同一個子網域.

如果您使用 PXE 2.0 BootPROM, 您只能使用 site-specific 選項第 128 至 135 以傳遞訊息給 BpBatch. 大於 135 的選項會被 PXE 2.0 BootPROM 忽略掉. 選項 135 用來傳遞 BpBatch 的命令列參數. 所以你還有 6 個選項可以用來自訂.

4.1.2 設定 Proxy DHCP 伺服器

Proxy DHCP 伺服器所伴演的角色是用來克服 DHCP 伺服器的限制並提供 PXE 規範的延伸訊息. Proxy DHCP 伺服器只爲了與 PXE BootPROM 搭配使用. 因此, Proxy DHCP 伺服器不是必要的. 如果你的 DHCP 伺服器能回應所有 PXE BootPROM 要求的選項, 則 PXE BootPROM 就不會去尋找 Proxy DHCP 伺服器 (aka BINL). 而我們所說的 "所有 PXE BootPROM 要求的選項", 是指 IP 選項 (IP 位址, 閘道器, 子網域, 主機名,...) 以及 PXE vendor-specific 選項 (vendor-class – 選項 60 – 用以設定 "PXEClient" 而 vendor-encapsulated-options 設定一些有意義的資訊給 PXE BootPROM. 你可以用 01:04:00:00:00:00:ff 如果你不知道對 PXE BootPROM 而言那一個資訊是有意義的). 更完整的敘述請看下面的說明.

BpBatch 本身功能就很完整, 你不須使用任何 PXE 規範的 DHCP 延伸參數 (選單, 等.). 然而, 如果您的 DHCP 伺服器不能提供最低限度的 PXE 相容, 則您就需要一台 Proxy DHCP 伺服器, 否則您的 PXE BootPROM 無法繼續使用. 最低限度的相容, 是指允許設定 VendorClass 及 VendorOptions (選項 60 及 43).

在 Windows NT 下, 您可以試著用 Intel WfM PDK (可在他們的網頁找到), 但它並不容易操作. 我們建議您找一台在子網域中, 跑 Linux 的機器安裝我們的小型 Proxy DHCP 伺服器. 我們的 Proxy DHCP 伺服器對 BpBatch 的最大優點是, 讓你可以自訂選項 155 vendor 標籤讓 BpBatch 解釋爲命令.

在 Linux 及 Solaris 下, 您可以使用我們的 Proxy DHCP 伺服器程式, 你只需設定好 TFTP 伺服器的 IP 位址, 可開機檔檔名及其它附加參數, 就可以完成所有的工作. 如果在伺服器上的 DHCP 通訊埠已被其它伺服程式使用, 則 Proxy DHCP 伺服器會改在 4011 埠上執行. 在這種狀況下, 使用 DHCP 通訊埠的程式必須回應 `PXEClinet` 給 client class. 這樣 PXE 用户端才會知道要去試著在 4011 埠上通連.

如果你想更了解 PXE 對 DHCP 的延伸功能, 可以在 Intel 的網頁上找到更多說明. 但是這些文章有點怪怪的, 這個通訊協定竟然設定在額外的位置號碼上, 用以得到更大的彈性. 對這個問題唯一的解釋是 PXE 用户端需要完整的 增強型的 *DHCP* 回應. 如果它只收到標準的 DHCP 要求, 它會一直等到收到才停住.

1. 一個 client class (T60) 設定爲 `PXEClient`.
2. vendor encapsulated options (T43). (不可以是空的字串, 但你可以設爲 16 進制的 `01:04:00:00:00:00`, 這可以被 PXE BootPROM 所接受)
3. 一個非空白的可開機檔檔名.
4. 一個有效的主機名.

PXE 規範協調的工作會在所有的資訊都取得後才結束, 如果缺了某一部份的資訊, 將引發相當複雜的作業程序 (尋找安裝伺服器, 等問題.).

4.1.3 設定 TFTP 伺服器

TFTP 伺服器是相當簡單的檔案伺服器. 最基本的版本中, TFTP 通訊協定乃是採用每個區塊 512 位元組, 這個作法相當沒效率. Bootix 的 TCP/IP BootPROM 及 PXE BootPROM 允許使用較大的區塊來傳送 (1408 位元組), 此法可以大幅加速傳輸速度. 但, 這只能在增強型的 TFTP 伺服器上工作.

在 Windows NT 環境下, 我們建議採用 Intel 的 TFTP 伺服器, 可以在 Intel PXE Product Development Kit for Windows 找到. 你可以在這找到這個 PDK *http://developer.intel.com/ial/wfm/tools/pxepdk20/index.htm*. 極重要的問題: 如果你計畫在 Windows NT 上使用 Intel 的 TFTP 伺服器, 別忘記必須重新命名你的 BpBatch 載入器名稱, 由 bpbatch.P 改爲 bpbatch.B. 這將告知 BpBatch 採用與 Intel 相容的 TFTP 選項.

Intel 的 PXE PDK 還包還含了評估版的 PXE BootPROM 程式碼, 給 Intel 的 EPRO100 族系網路卡使用, 亦包含了用來更新快閃記憶體的工具程式以升級爲 BootPROM 內容爲最新版的 PXE.

在 Linux 環境下, 你可以使用大部份套件中所含的 tftp daemon, 但不具備增強的 TFTP 模式 (1408 位元組的封包). 我們建議使用 Bootix 的增強型 TFTP 伺服器, 可以在我們的目錄下找到 (*http://cuiwww.unige.ch/info/pc/remote-boot/soft/*), 檔名爲 `incomtftpdlx.tar.gz`.

在 Solaris 環境下, 你可以使用 Bootix 增強型 TFTP 伺服器, 也可以在我們的目錄下找到 (檔名爲 `incomtftpdsun.tar.gz`). 另一種可用的選擇: John Martin (`j.martin@abdn.ac.uk`) 已把 Linux 上的 `tftpd` 移殖到 Solaris 上. 你可以在這找到 *http://www.csd.abdn.ac.uk/~jmartin/tftpd.tar.gz*.

如果你較喜歡使用標準的 **TFTP** 伺服器, 把所有可開機映像檔檔名的副檔名 `P` 拿掉, 這樣可以告知 **BootPROM** 採用標準的 **TFTP** 通訊埠通訊. (這個小技巧是 **Bootix** 給 **TCP/IP BootPROM** 使用. 我們也可以用這個簡單的方法來選擇 預設的 **TFTP** 通訊埠給 **PXE BootPROM** 使用).

摘要來說, 如果你計畫使用標準的 TFTP 伺服器, 則把 BpBatch 載入器檔名的副檔名 移去 (也就是改爲 bpbatch). 如果你計畫使用 Bootix 支援在 59 埠, 以 1408 位元組封包的 TFTP 伺服器, 則你的載入器檔名必須有副檔名 `.P` . 最後, 如果你計畫使用 Intel 的 TFTP 伺服器 (或 Bootware 的那個伺服器), 則把載入器命名爲 `bpbatch.B`.

4.2 用户端的設定

首先, 我們把所有作業系統相同的部份設定好. 如: 批次檔的解譯器. 然後, 再針對各種作業系統來完成, 我們需要下列的步驟:

1. 設定單一的用户端.
2. 儲存其設定到伺服器.
3. 測試它是否可以爲遠端開機的用户端電腦所使用.
4. 調整它的設定以適用於其他硬體配備類似的用户端機器.

如果這些都完成了, 你要加一台用户端, 則只要插上 Boot ROM (或買台具 Wired for Management 的電腦...) 然後再加一行設定到 DHCP 伺服器的設定檔. 譯註: Wired for Management 應該是指本身具備 PXE 功能的品牌電腦.

我們的範例是假設你的用户端電腦的硬碟空間至少有 1.4 G 以上. 如果低於這個值, 那就要縮小分割區, 但請記住你要保留幾百 MB 空間不被對應到 (也就是, 不要連最後一個分割區都已畫分到最後一個磁柱) 以保留空間給特別的快取分割區. 然而, 快取區通常接在最後一個對應到的磁柱開始對應, 如果你不想用相同大小來做測試, 你將下載同樣的檔案數次 (快取區會被自動清除).

別灰心. 如果你一直無法讓它正常工作, 先參考 問題集 那一節, 搞不好你的問題已被解決過了 (請自網頁取得最新版本). 再不行, 查一下 BpBatch 討論區. 或許有人跟你遇到同樣的問題, 而解答可以在討論區找到. 討論區的網址是: *http://cuiwww.unige.ch/info/pc/remote-boot/forum/*. 如果還是無法正常工作, 考慮看看監控一下網路流量是否是網路本身的問題 (在 Linux 環境下, 可以用 `tcpdump` 而 Solaris 環境, 可以用 `snoop`). 如果你依然無法讓它正常工作, 送一封 E-mail 到 `support@bpbatch.org`. 如果你的問題與遠端開機的設定有密切關係, 我們會試著解決你的問題.

4.3 設定開機流程

取得 BpBatch 軟體, 不管是 `.zip` 或 `.tar.gz` 格式. 可執行檔都放在

- *http://cuiwww.unige.ch/info/pc/remote-boot/soft/bpb-exe.zip*.
- *http://cuiwww.unige.ch/info/pc/remote-boot/soft/bpb-exe.tar.gz*.

在伺服器上的 `/tftpboot` 目錄下, 放入下面三個特別的可開機映像檔. 他們是用來完成開機前的批次檔案解譯功能 (下面我們用 `bpbatch.P` 爲載入器的檔名. 如果你不是用 Bootix 的 TFTP 伺服器, 則變更載入器檔名爲 `bpbatch` (標準型 TFTP 伺服器) 或 `bpbatch.B` (Intel/Bootware 的 TFTP 伺服器):

- `bpbatch.P`, 動態載入器 (注意大小寫並參考 設定 TFTP 伺服器那節, 以選擇適當的附檔名).
- `bpbatch.ovl`, 中斷重導檔
- `bpbatch.hlp`, 線上說明檔

然後爲你的用户端加一些 DHCP 的設定工作, 其中: 可開機檔檔名設定爲 `"bpbatch.P"`. 定義 vendor 選項標籤 155 (十進制) 爲 `"-i"` (在標準型的 DHCP 伺服器, 可以用以下的指令完成: `option option-155 "-i";`). 它會被 `bpbatch` 解釋爲命令列, 而 `-i` 意謂著 "交談模式".

啓動用户端電腦. 你應該看到

- BootPROM 的版權宣告.
- 當用户端在等待 DHCP 回應時, 會出現 `DHCP` 字串.
- 當用户端在等待第一個 TFTP 封包時, 會出現 `TFTP` 字串.
- 當載入器下載解譯器時, 會出現 `Loading BpBatch` 字串.
- 最後是我們的旗幟, 隨著一個 極大的 提示符.

恭喜！你以經啓動了批次檔解譯器... 如果你很急於知道在這環境下你可以做舍事, 那就繼續閱讀下一節. 如果你不急, 跳過它直接安裝你所選擇的作業系統. 如果你有任何關於解譯器指令的疑問, 輸入 `help`.

記住你可以在 DOS 及 Linux 下執行相同的解譯器, 只要執行 `MrBatch` 程式. 只有一點點的差異 (Linux 版本不支援圖形化的功能, 而 DOS 版本如果在 BootPROM 未被作業系統隱藏之前, 只能送出 BOOTP 及 TFTP 的要求訊號).

現在是閱讀 BpBatch 語法 的時候了, 尤其是 參考檔案 及 快取檔案系統 這兩段. 這將幫助你了解範例.

當所有的作業系統都設定好, 你要製作一個選單讓使用者去選擇它所要的是哪一種作業系統. 你可自行探究如何去產生那種選單. 而所需的指令都在本文末端.

4.3.1 探討 BpBatch

試著輸入 `LogVars`. 你應該可以看到大約三十幾個變數被列出來. 概括來說, 先出現 BpBatch 的變數設定, 然後是自 BOOTP/DHCP 通訊協定取得的回應, 最後是硬碟大小的變數, 以百萬位元組爲單位.

輸入 `GetPartitions part`, 再輸入 `LogVars` 一次. 就應該看到在你第一個硬碟機內的分割區定義資訊. 假設第一個分割區有 BIGDOS, FAT32 或 LINUX-EXT2, 試著打 `LogDir "{:1}"` 以取得根目錄的內容, 然後輸入 `LogDir "{:1}/usr"` 如果有 `usr` 目錄的話. 你也可以試著打 `LogTree "{:1}/etc"` 以取得樹狀目錄.

放一個 GIF 檔 (格式是 GIF-87a, 交錯或非交錯的都行, 但不可是 GIF-89a) 到你的 TFTP 伺服器. 我們假設檔名是 `image.gif`. 你可以用下列的指令把它複製到任何地方: `copy "image.gif" "{:1}/temp/image.gif"`.

或者, 你可以自伺服器上直接使用. 現在輸入 Logvars "V*" 並檢視 VESA 變數值. 如果它顯示 On, 那就相當可能意謂著, 你的顯示卡有 VESA相容的功能. 你可以用 Echo "$VESA-Modes" 列出有效的顯示模式. 要秀出你的圖檔只要試著輸入以下的指令: DrawGif "image.gif". 這個圖檔應該顯示在螢幕左上角. 你可以在檔名後標定 X 及 Y 軸, 把它擺在其他地方. 你也可以用 DrawText 200 200 "Hello world" yellow 畫出文字. 或用 DrawWindow 200 200 300 150 畫出一個空的視窗. 而要在產生新視窗時插入主題, 可以試著用 DrawWindow 200 200 300 150 "My Window". 如果你對圖形化模式感到厭煩只要打 CloseGraph.

在圖形化模式下要注意: 預設情況下, 所有的圖形功能在 800x600 VESA 模式下工作 (有 256 色階), 這是在 VESA-Modes 變數下的第一個欄位. 如果你要使用不同的顯示模式, 則變更第一個欄位爲你所要的顯示模式.

現在用文字編輯器, 在 tftpboot 目錄下, 建一個名爲 test.bpb 的檔案內容是:

```
:again
DrawWindow 150 200 400 160 "Identity check"
TextAttr Black LightGray
At 15,20 Print "Username : "
Input username 8
At 17,20 Print "Password : "
Getpasswd userpass 8
if "$username" != "smith" goto again
if not "$userpass" match-passwd "BpR8oiIlRR9bo" goto again
#
clear
DrawWindow 200 200 150 100 green blue "Congratulations"
DrawText 220 250 "You got it !" yellow
WaitForKey 3
CloseGraph
interact
```

在你的 BOOTP/DHCP 伺服器的設定檔中, 變更 選項-155 由 "-i" 爲 "test", 並把用户端電腦重新開機. 這個小型的腳本程序可以自動執行, 並會尋問你的使用者名稱 (username) 及 判行密碼 (password). 如果你不輸入 smith 及 justdoit, 你就不能使電腦開機. 接著你將學得如何用 Unix, NT 或 Radius 伺服器以辨證有效的使用者.

4.4 設定 Linux

爲了安裝 Linux, 你需要 RedHat Linux 套件所提供的可開機軟碟. BpBatch 含有一個可以指向軟碟開機的指令: FloppyBoot.

安裝 *RedHat Linux* 到你的用户端電腦, 必需具備網路支援功能. 以及你所需要的套件軟體. 你或許需要重新編譯一個 Kernel 以合於你的硬體配備, 但這不是必需的.

4.4.1 設定用户端電腦

在 Kernel 中加入對 BOOTP 通訊協定的支援將會是一個好主意, 這樣你就不用手動去設定用户端的 IP 位址.

爲了降低網路負載, 你應該安裝 file cache 以快取由 NFS 所載入的檔案到硬碟中. 簡單的說, file cache 的功能是當有來自 cache 子目錄下的符號連結形成時, 它會被它的目標所取代. 如果目標本

身是子目錄, 則在子目錄下的所有東西都會變成符號連結, 並指向外在檔案系統的原始資料. 這個 file cache 由 Unifix GmbH 所寫, 已成爲 Unifix Linux 2.0 的一部份. 它可以自由散布, 你可以在 *http://cuiwww.unige.ch/info/pc/remote-boot/soft/filecache.tar.gz* 取得必要的檔案. 爲了使用這個 file cache, 你必須要

- 對 Kernel 加入一些修正 (檔案是 `patch-filecache`), 透過 `make config` 或你所喜歡的方式來啓用這個功能, 並重新編譯 kernel.
- 複製 file cache 的可執行檔到 `/sbin`.
- 建一個可以掛上的點叫 `/mnt/nfs` (使用 `mkdir` 指令).
- 複製 `filecache.conf` 到 `/etc`. 此檔的內容包含以下幾行:

```
Max 100 MB 50 % #
Cache /mnt/nfs/usr /usr
Cache /mnt/nfs/opt /opt
```

- 複製 `/usr` 及 `/opt` 內的資料到伺服器, 以 `anon=0` (允許root 存取) 設定爲唯讀屬性的分享並把它掛在 `/mnt/nfs` 下 (加入一行到 `/etc/fstab`)
- 更名 `/usr` 爲 `/usr.orig`.
- 建立連結 `/usr` 指到 `/mnt/nfs/usr`.
- 更名 `/opt` 爲 `/opt.orig`.
- 建立連結 `/opt` 指到 `/mnt/nfs/opt`.
- 確定 `/usr` 及 `/opt` 目錄不是空的並應該包含正確的目錄資料.
- 以遞迴的方式移除 `/usr.orig` 及 `/opt.orig`.
- 複製 `filecache.init` 到 `/etc/rc.d/init.d`.
- 最後建立連結 `/etc/rc.d/rc3.d/S35filecache` 指到 `/etc/rc.d/init.d/filecache.init`.

如果你完成上述的步驟, 你應該在下次開機時已具備 file cache 功能, 只要你沒忘記修正你的 Kernel.

4.4.2 測試設定檔

複製你的壓縮 Kernel 映像檔 (`zImage`, `bzImage`, `vmlinuz` 或其它的命名) 到伺服器的 `/tftpboot` 目錄下, 更名爲 `linux.krn`. 如果你已經把 BootPROM 從你的 PC 拔起來了, 那現在必須重插回去. 當 `BpBatch` 啓動, 輸入 `LinuxBoot "linux.krn" "root=/dev/hda1 BOOT_IMAGE=linux"` (假設 根的 ext2 檔案系統在第一個分割區). 另一種狀況, 如果你要在無 BootPROM 的電腦上設定你的組態, 只要用你所安裝的載入器 (lilo, ...) 來開機就可以. 但在後面的情況下, 如果你要使 file cache 正常工作, 你就必須把 file cache 的支援功能正確的安裝到你的 Kernel.

等到系統開機完成. 如果你已安裝 file cache, 你可以檢查 `/usr` 目錄下, 已經直接變成一些目錄型式的符號連結到已劃分好的目錄. 現在啓動使用者常用的程式, 幫他們把所需的資料載到硬碟中.

你還可以調整你的設定, 讓它更像其他獨立運作的 Linux 工作站.

4.4.3 建立磁碟映像檔

當你覺得你的設定用起來很棒了, 就可以用 `root` 登入, 並到 `/tmp` 目錄下, 執行我們的 `mrzip` 程式. `MrZip` 是一個像 `BpBatch` 的命令解譯器, 但它可以了解比 `BpBatch` 更多的命令. 特別是, 它可以了解下列的命令:

```
showlog
filter -"tmp/*"
filter -"var/log/*"
fullzip "/" "/tmp/linux.imz"
```

這將產生一個磁碟映像檔到 `/tmp/linux.imz`. 把它移到伺服器的 `/tftpboot` 目錄下. 然後複製下列的批次執行檔到 `/tftpboot/linux.bpb`:

```
hidelog
setpartitions "linux-ext2:992 linux-swap:32"
fullunzip "linux.imz" 1
clean 2
linuxboot "linux.krn" "root=/dev/hda1 BOOT_IMAGE=linux"
```

這個 `BOOT_IMAGE` 參數是用來表示相容於 RedHat 5.1 的 `lilo` 及之後的 `rc.sysinit`.

你的 Linux 遠端開機設定組態已經就續! 你現在只要設定 BOOTP-option-155 爲 `"linux"`, 或在 BpBatch 下輸入 `include "linux.bpb"` 就可以測試它了.

4.4.4 系統維護與升級

如果你以後要更新軟體, 安裝程式臭蟲修正或安全性的修正, 處理程序如下:

- 以遠端開機方式啓動一台用户端電腦以得到一個全新的 Linux.
- 完成你需要的改變.
- 重製磁碟映像檔.
- 複製新的磁碟映像檔到伺服器以取代舊的.

這就意謂著, 你可以就像在使用一台普通的電腦一樣來更新你的伺服器爲基礎的組態設定.

4.5 設定 DOS 6 及 Windows 3.1

到用户端電腦上, 用你所喜好的 DOS 磁片開機 (可以移去 BootPROM 或在 BpBatch 環境下, 輸入 `FloppyBoot`). 用 `/S` 選項, 格式化磁碟中的 DOS 分割區, 以安裝作業系統. 分割區的大小並不重要. 因爲以 `MrZip` 產生磁碟映像檔案時, 是以檔案來認定的. 而與分割區大小無關.

建一個 `DOS` 子目錄, 把 DOS 複製進去. 安裝你所喜好的網路用户端軟體 (比如 Microsoft LanManager), Windows 3.1, 及其他你要的程式. 如果你要用 Microsoft LanManager, 千萬別使用它的 DHCP 通訊協定來組態 IP 設定, 因爲它的實作方法相當爛, 在正常的網路負載下, 也絕對有可能失效. 但要得到相同效果, 可以加入以下幾行到 `protocol.ref` 檔, 在載入 `tcptsr` 的同時 (它會把 `xxx` 以你給定的眞確 IP 參數值取代):

```
IPADDRESS0 = xxx xxx xxx xxx
SUBNETMASK0 = 255 255 xxx xxx
DEFAULTGATEWAY0 = xxx xxx xxx xxx
DISABLEDHCP = 1
```

不用怕使用 EMM386 最佳化您所使用的記憶體, 您可以把網路卡 ROM 所佔的區域納入管理, 因爲它在這時已經用不到了. 但要記得把網路卡所使用的 RAM 區塊不納入管理, 否則您將無法連接到您的伺服器. 用 NOEMS 參數.

如果你要確認用户端電腦只允許已認可的使用者所使用, 可以把我們的 nobreak 虛擬裝置驅動程式 (可在這下載 *http://cuiwww.unige.ch/info/pc/remote-boot/soft/nobreak.zip*) 放在您 config.sys 檔的開頭處, 並把以下這些東西放到您的 autoexec.bat:

```
rem -- we use the dummy file c:\logged as a flag
del c:\logged >nul
:loginneeded
cls
echo Please type in your login name and password
echo.
net logon *
rem -- the login script should have created c:\logged
if not exist c:\logged goto loginneeded
del c:\logged
rem -- now enable break again
echo Yes >NOBRK
```

重開機, 確認你的用户端電腦一切正常, 並試著在 BpBatch 的交談模式下輸入以下指令:

```
HideBootprom
HdBoot
```

如果你沒看到 BpBatch 則 PC 會採用硬碟來開機.

4.5.1 建立磁碟映像檔

回到伺服器上, 建立一個分享名稱爲 admin 的目錄, 用來放一些系統管理者用的東東. 如果伺服器是跑 Unix, 建議您把 /tftpboot 子目錄用軟式連結方式放一份到 admin 目錄, 這樣您可以直接由用户端傳回開機映像檔. 在 admin 目錄內, 建一個 /utils 子目錄並放入如下的工具:

- mrbatch.exe, DOS 版的 BpBatch.
- mrzip.exe, DOS 版的工具程式用來建立磁碟映像檔.
- bpbatch.hlp, 線上說明檔.

您也需要放一些簡便的 MrZip 腳本程序檔名爲 zipdos.mrz 的檔案, 內含用來建立 DOS 磁碟映像檔的指令, 就像下面這樣:

```
showlog
filter -"lanman.dos/lmuser.ini"
```

```
filter -"temp/*"
filter -"*.swp"
fullzip "c:/" "L:/tftpboot/dos.imz"
```

現在回到你的用户端電腦, 把 admin 掛到 L:, 到你的 utils 子目錄下, 輸入以下指令:

```
mrzip -b zipdos
```

一分鐘之後, 你會在伺服器的 /tftpboot 子目錄下看到一個名爲 dos.imz 的檔案, 它就是你用户端電腦硬碟的壓縮磁碟映像檔. 把以下的批次檔內容複製到 /tftpboot/dos.bpb:

```
hidelog
setpartitions "bigdos:1024"
setbootpart 1
fullunzip "dos.imz" 1
hideBootprom
hdboot :1
```

你的 DOS 遠端開機設定組態已經就續！你現在只要設定 BOOTP-option-155 爲 "dos", 或在 BpBatch 下輸入 include "dos.bpb" 就可以測試它了.

4.5.2 調整設定以適用其他機器

如果你要針對某一些機器做一些特定的設定, 最典型的是 IP 位址值的設定不過因爲 Micro$oft 的 DHCP 做的很爛, 你可以用 BpBatch 在開機時做一些修正. 首先到 lanman.dos 目錄下並執行

```
copy *.ini *.ref
```

接著編輯副檔名爲 .ref 檔案, 把所有固定的參數用 BOOTP 的變數取代, 下面是範例:

```
computername = ${BOOTP-Host-Name}
ipaddress0 = ${MS-IPAddress}
subnetmask0 = ${MS-IPSubnet}
defaultgateway = ${MS-IPRouter}
```

接著用先前的方式重建磁碟映像檔. 注意一下 IP 參數, 我們不能直接使用 BOOTP 變數給 LanManager 使用, 因爲它需要以空格分隔而不是以點來分隔數字. 把 dos.bpb 改成如下的樣子:

```
hidelog
setpartitions "bigdos:1024"
setbootpart 1
fullunzip "dos.imz" 1
set MS-IPAddress="$BOOTP-Your-IP"/.= /
set MS-IPSubnet="$BOOTP-Subnet-Mask"/.= /
set MS-IPRouter="$BOOTP-Routers"/.= /
patch "{:1}lanman.dos/protocol.ref" "{:1}lanman.dos/protocol.ini"
patch "{:1}lanman.dos/tcpputils.ref" "{:1}lanman.dos/tcputils.ini"
patch "{:1}lanman.dos/lanman.ref" "{:1}lanman.dos/lanman.ini"
hideBootprom
hdboot :1
```

如果你喜歡, 你也可以把 .ref 的檔案放在伺服器上的 /tftpboot 目錄下, 而不放在磁碟映像檔裡.

我們可以在不重建映像檔的狀態下, 輕鬆的變更電腦的設定. 作法是, 複製你的 autoexec.bat 及 config.sys 爲 autoexec.ref 及 config.ref 到伺服器的 /tftpboot 目錄, 把下面兩行加到批次檔中:

```
patch "autoexec.ref" "{:1}autoexec.bat"
patch "config.ref" "{:1}config.sys"
```

你可以透過 BOOTP 通訊協定, 自由的變更檔案內容. 即使是做一些根據機器而來的特殊化設定.

在對用户端電腦的設定檔做任何修改後, 如果你要預先做好修正. 別忘了以 MrZip 重新製作磁碟映像檔.

4.5.3 系統維護與升級

如果你以後要更新軟體, 安裝程式臭蟲修正或安全性修正的程式, 其處理程序如下:

- 以遠端開機方式啓動一台用户端電腦以得到一個全新的 Linux.
- 完成你需要的改變.
- 重製磁碟映像檔.
- 複製新的磁碟映像檔到伺服器以取代舊的.

這就意謂著, 你可以就像在使用一台普通的電腦一樣來更新你的伺服器爲基礎的組態設定.

4.6 設定 Windows 95/98

在此份文件先前的版本, 我們使用過微軟公司的 Windows 95 以伺服器爲基礎的安裝方式, 但是用這方法太過痛苦, 又不是那麼的有價值:

- 它就眞的是非常的虛擬.
- 大部份的軟體套件都不支援它而且如果您試著去安裝最後的結果通常是失敗的. 包括, Microsoft Internet Explorer, OnNet 32, Novell's Protected-mode client (這較 Microsoft Client for Netware安全).
- 因爲微軟公司不提供 Windows 95/98 在眞實模式下的 TCP/IP 相容的驅動程式所以您無法使用 Microsoft Network client over TCP/IP, 而這也就意味著您不能使用 Samba.
- 因爲當用户端一啓動就會鎖住一些動態連結檔造成許多程式幾乎無法更新, 如果您試著去更新它則會出現 違背分享 *(sharing violations)* 的錯誤.

因此幾個月前, 我們放棄這份本身就有問題的資料 (您仍然可在這找到 *http://cuiwww.unige.ch/info/pc/remote-boot/win95old/win95old.html*) 然後換到我們這個新的以硬碟機爲基礎的遠端啓動概念. 基本上, 在這種環境下 Windows 95/98 安裝方法就跟安裝 DOS 一樣容易.

4.6.1 建立一台獨立的用户端

安裝一台尋常的 Windows 95/98 用户端, 可以用我們先前所提的 DOS 網路開機組態模式啓動電腦, 再透過網路安裝 (我們都這樣做). 你也可以用已經安裝 Windows 的機器來做, 但若這樣你對硬碟裡有舍東西就比較不了解了.

把它視爲先前所提的 DOS 用户端來處理. 但在 Windows 95/98 是不需使用 EMM386. 如果你使用 Windows 98 或 Windows 95 OSR2 (別名爲 MSWIN 4.1, 也就是 Windows 95 service pack 1, 亦是 Windows 95 with Internet Explorer) 你應該加入以下幾行到 `MSDOS.SYS` (沒錯, 他是文字檔) 的 `[Options]` 那一節:

```
AUTOSCAN=0
```

這樣可以使 Windows 知道您開機時, 不自動啓動 ScanDisk.

當所有的軟體都放在伺服器時, 如果您要減輕網路及伺服器的負擔 (這將增進您的系統效率), 您可以考慮安裝這套相當不錯的 Shared LAN Cache, 由 Measurement Techniques, Inc 生產的 (參考 *http://www.lancache.com*).

這套軟體在每台用户端電腦執行, 它會把所有從網路上取得的資料放在區域性的硬碟中快取. 即使像 MS-Office 這種怪物, 在您第二次執行時也會更快... 您的每台客户端電腦都要有一份版權, 不過它還不算太貴, 這家公司對於大專院校有特惠價格. 更棒的是, 您可以到他們的網頁下載免費的評估版.

4.6.2 建立磁碟映像檔

在這個地方, 如果你是使用 Samba 檔案伺服器, 你應該需要先關掉 Windows 98 的密碼傳輸前加密功能. 這樣才能把 `admin` 掛上去當網路磁碟機. 你只要把下面的登錄組態 `EnablePlainTextPassword` 節的 DWORD 值設爲 1

`[HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\VNETSUP]`

你的 MrZip 腳本程序檔可以命名爲 `zipwin.mrz` 並包含以下內容:

```
showlog
filter -"windows/schedlog.txt"
filter -"windows/msimgsiz.dat"
filter -"*/index.dat"
filter -"*.swp"
filter -"*.tmp"
filter -"temp/*"
fullzip "c:/" "L:/tftpboot/win.imz"
```

爲了建立映像檔, 把 `admin` 掛到磁碟機 `L:`, 並到你的 `utils` 目錄下並輸入以下指令:

```
mrzip -b zipwin
```

幾分鐘過後, 你可以在伺服器的 `/tftpboot` 子目錄下, 找到名爲 `win.imz` 的檔案, 它就是你硬碟的壓縮磁碟映像檔. 如果你的映像檔超過 87 MB, 最好把它分割成兩個或更多的檔案. 這些檔案會在我們需要它時自動地一個接著一個載入. 記住, 如果你的映像檔超過 87 MB 則系統花在解壓縮的時間會超過一分鐘而這會激怒你的使用者. 我們的 Windows 95/98 映像檔止有 70 MB 大, 因爲大部份的軟體都擺在伺服器, 且只有在要求時才會載入. 只有極少部份的軟體需要安裝到你的硬碟中. 如果你跟我們一樣在乎這問題, 你應該讓解壓縮這個映像檔及完整回存到硬碟的時間低於一分鐘.

把以下的批次檔內容複製到檔案 `/tftpboot/win.bpb`:

```
hidelog
setpartitions "bigdos:1024"
setbootpart 1
fullunzip "win.imz" 1
hide‘
hdboot :1
```

你的 Windows 95/98 遠端開機設定組態已經就續！你現在只要設定 BOOTP-option-155 爲 "win", 或在 BpBatch 下輸入 include "win.bpb" 就可以測試它了.

4.6.3 調整設定以適用其他機器

Windows 95/98 與 Windows 3.1 最大的不同是加入了即插即用的程式碼, 即自動偵測您的硬體設備. 它本身無舍壞處, 但問題是出在它太過靈敏, 而且有時後會出鏈.

如果您試著用同樣的開機映像檔去啓動另一台用户端機器, 您將在收到一堆 Windows 偵測到新的硬體設備的訊息: 一片新的音效卡, 一顆新的硬碟機, 一片新的網路卡, 或者一隻新的滑鼠... 會這樣的原因有二:

- 這些設備可能使用不同的硬體資源 (比如, 滑鼠不是接在相同的連接埠, 音效卡不是接在同樣的擴充槽 -.- 沒錯, 這會被偵測到).
- 這些設備可能通知 Windows 95/98 它們的序號 (比如, 每台跑 Windows 95/98 的網路卡, 從最根本的乙太網路位址 就每台都不一樣了).

事實上 , 如果 Windows 95/98 的即插即用功能可以正常工作的話, 自動偵測變更的硬體設備並不會出問題, 所以如果即插即用功能失效時, 問題就來了. 例如, Windows 95/98 對上我們的羅技 PS2 滑鼠時就不能即插即用, 更糟的是滑鼠就消失了. 要解決這類問題方法就是, 盡可能使全部電腦有相同的硬體設備.

您無法避免的問題是每台電腦的網路卡. 很不幸的, 在搭配我們所用的 SMC EtherEX 網路卡時, 即插即用的程式碼會使電腦當機. 唯一的解法是使 Windows 95/98 相信它已經知道有這片網路卡, 且不需觸動即插即用機制. 這個技巧就是透過開機自動執行檔 autoexec.bat, 自動的把關於這片網路卡的設定值加入 Windows 95/98 的系統登錄資訊.

要注意的是大部份 PCI 的網路卡並不需要這個技巧.

到您的用户端電腦, 編輯開機自動執行檔 autoexec.bat 並加入下列幾行:

```
rem --- Patch Windows registry in order to avoid plug-and-play detection
regedit /L:c:\windows\system.dat /R:c:\windows\user.dat c:\temp\patch.reg
```

regedit 是標準 Windows 95/98 的程式. 如果您在 Windows 環境下執行它, 它可以幫您流覽您的系統登錄資訊, 或者如果您在 DOS 下執行它, 它可以對您的系統登錄資訊做一些簡單的修正. 在視窗環境下執行 regedit, 尋找你的網路卡, 通常在

```
HKEY_LOCAL_MACHINE\Enum\ISAPNP
```

用 登錄 功能選單的 匯出登錄檔 功能. 這個動作將建立一個文字檔, 你可以把它命名爲 patch.ref 放在伺服器的 /tftpboot 目錄下. 編輯這個檔案並找到這個網路卡的乙太網路位址放置的位置 (如果你找不到它, 可以在不同機器上產生這兩個檔案, 比較他們就可以找到). 用 ${MACID} 的樣式來取代它. 接著在 win.bpb 腳本程序檔中加入以下幾行:

```
set macid = "$BOOTP-Client-ID"
patch "patch.ref" "{:1}temp/patch.reg"
```

(如果它不是用户的正確乙太網路位址, 就必須對 MACID 字串設定值做必要的修正). 就這樣, 你的用户端電腦就不會再試著去自動偵測網路卡了.

再提一次, 這整個技巧在使用 PCI 網路卡時, 是不需要的. 另外, 當 Windows 不能在登入時透過 DHCP 通訊協定設定主機名 (hostname), 我們也可以用相同的手法來達到. 只要把下列幾行加到我們的 patch.ref 檔:

```
[HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\VNETSUP]
"ComputerName"="${BOOTP-Host-Name}"

[HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\MSTCP]
"HostName"="${BOOTP-Host-Name}"

[HKEY_LOCAL_MACHINE\System\CurrentControlSet\control\ComputerName\ComputerName]
"ComputerName"="${BOOTP-Host-Name}"
```

使用登錄編輯程式的小技巧, 你的組態設定可以順利的轉移到有類似配備的機器上. 如果你無法避免 Windows 去偵測你的某些硬體配備, 可以試著在那機器上重建一個磁碟映像檔. 這樣會把這台機器上的組態設定都含入這個映像檔中, 希望這能解決問題.

4.6.4 系統維護與升級

如果你以後要更新軟體, 安裝臭蟲修正或安全性修正的程式, 處理程序如下:

- 以遠端開機方式啓動一台用户端電腦以得到一個全新的安裝.
- 完成你需要的改變.
- 重製磁碟映像檔.
- 複製新的磁碟映像檔到伺服器以取代舊的.

這就意謂著, 你可以就像在使用一台普通的電腦一樣來更新你的伺服器爲基礎的組態設定.

4.7 設定 Windows NT

我們並沒有使用 Windows NT 來當遠端開機用户端電腦的作業系統, 但我們已經測試過, 它確定可以正常工作. 並且沒問題.

由於我們的工具程式還沒支援到 NTFS 檔案系統 (我們既沒有相關文件也沒有時間去做這件事, 但我很樂意去幫助任何對這工作有興趣的人) 所以你必須把 NT 安裝到 FAT16 檔案系統上 (只要在安裝時, 不把你的分割區轉換爲 NTFS 檔案系統就可以).

把 win.bpb 開機腳本程序複製到 winnt.bpb. 改變 winnt.bpb 中 setpartitions 那一行爲如下:

```
setpartitions "BIGDOS:512 BIGDOS:512"
```

接著用這個腳本程序檔使 Windows 95/98 開機, 把 NT 用户端安裝到 C 槽. 先別管第二個分割區的問題. 別安裝太多東西進去, 不然會產生一個又大解壓縮又很慢的映像檔. 把 Windows 95/98 從 C 槽移除, Windows NT 的映像檔不需要它 (開機選單是由 BootPROM 所控制, 不再是 NT 的開機載入器).

在不覆寫硬碟資料狀況下, 重新啟動你的電腦, 就是千萬別去執行 `winnt` 腳本程序檔. 只要做:

```
hideBootprom
hdboot
```

你的 NT 工作站就可以正確的啟動. 然後就可以做其他的特殊修正了.

4.7.1 建立磁碟映像檔

Windows NT 麻煩的地方是出在: 磁碟直接存取的功能會被 Kernel 所禁止. 意謂著, `MrZip` 無法讀取開機磁區. 製作映像檔的最佳方式是以 Windows 95/98 開機, 在 DOS 視窗下, 執行 `MrZip`. 要這麼做就要變更 `winnt.bpb` 腳本程序檔內容. 使得 Windows 95/98 映像檔不回存到第一個分割區, 而是第二個:

```
hidelog
setpartitions "BIGDOS:512 BIGDOS:512"
setbootpart 2
fullunzip "win.imz" 2
hidebootprom
hdboot :2
```

(如果你有其他補充的修正, 在把 "{:1}" 變爲 "{:2}"). 用這個腳本程序開機; 應該會讓 Windows 95/98 執行, 而新磁碟機 `D:` 槽會出現, 裡頭放的就是 Windows NT.

就如以往的方式建立你的磁碟映像檔 (只是改對 `D:` 做, 當然了), 把它存爲 `winnt.imz` 並放在伺服器上的 `/tftpboot` 目錄下. 最後編輯 `winnt.bpb` 腳本程序檔改成如下:

```
hidelog
setpartitions "BIGDOS:512 BIGDOS:512"
setbootpart 1
fullunzip "winnt.imz" 1
clean 2
#fullunzip "win.imz" 2
hidebootprom
hdboot :1
```

你的 Windows NT 遠端開機設定組態已經就續. 當然, 如果你不想有兩個分割區, 你可以設定成單一個分割區. 但當你要重建映像檔時, 你還是需要再設定第二個分割區以 Windows 95/98 開機.

4.7.2 系統維護與升級

如果你以後要更新軟體, 安裝程式臭蟲修正或安全性的修正, 處理程序如下:

- 以遠端開機方式啟動一台用户端電腦以得到一個全新的安裝.
- 完成你需要的改變.
- 編輯 `winnt.bpb`: 把 `clean` 及 winnt `fullunzip` 改成註解, 把 win95 `fullunzip` 改成可執行.

- 重製磁碟映像檔.
- 複製新的磁碟映像檔到伺服器以取代舊的.

就這樣, 搞定

4.8 問題集 (FAQ)

本節列出我們所統計最常遇到的問題.

不停地下載映像檔

可能是因爲你所使用的是標準型 TFTP 伺服器, 而它不能處理超過 65535 個 512 位元組大小的封包 (而 Solaris 伺服器, 在超過 32767 個封包就有這個問題). 因爲這樣, 你的映像檔必須拆成許多個不超過的檔案 30 MB (Solaris 的話, 是 15 MB). 參考 複製檔案*(CopyArchive)* 的機制, 它會教你如何分割已存在的映像檔. 如果你目前還在使用標準型 TFTP 伺服器, 你應該愼重考慮改用 Bootix 或 Intel 的 TFTP 伺服器. 參考下一個問題以探討這些伺服器的功能.

BpBatch 顯示 **"Fragment is not complete"** 的錯誤訊息

這是 TFTP 通訊協定的問題. 你的載入器檔名的副檔名一定要設對 (參考 TFTP 伺服器那節). 如果你用 Bootix 的 TFTP 伺服器並用了 `-s 1408 59` 的選項, 則你的載入器檔名應該是 `bpbatch.P`. 如果你是用 Intel 或 Bootware 的 TFTP 伺服器有支援 blksize 的 TFTP 選項功能, 則重命名你的載入器檔名爲 `bpbatch.B`. 最後, 如果你用標準型 TFTP 伺服器, 則重新命名你的載入器檔名爲 `bpbatch` (沒有副檔名). 這個狀況下 BpBatch 會用 512 位元組大小的封包取代 1408 位元組大小. 而這時分割檔的大小限制在 32MB (TFTP 內部限制) 不是 87MB. 參考上一個問題, 探討 512 位元組大小的封包.

檔案解壓縮時直接失效

有三種可能原因. 在伺服器上的映像檔葛屁了 (可以試著用 MrZip 檢查是不是這個問題), 或是 TFTP 傳輸延誤以致檔案檔案傳送失敗, 或是通訊協定不相容.

當網路負載太重, 就會發生 TFTP 傳輸延誤的狀況 (比如, 你想一次把一個極大的映像檔下載到超過四台的用户端電腦). 在這種情況下, `BpBatch` 不會一直重複傳輸. 因爲那樣作毫無用處. 關掉幾台電腦並試著在不超過四台用户端電腦同時啓動的環境下測試 (或者不超過三台). 如果你常常需要下載映像檔到許多台電腦, 你可以試用我們所使用的特殊廣播型 TFTP 伺服器 (參考討論它的那一節內容).

因標準型 TFTP 伺服器 (通常內建於你的 UNIX 伺服器) 所導致的通訊協定不相容. 因爲當 `BpBatch` 要以增強型的 TFTP 協定工作時, 伺服器以標準型 TFTP 協定來對應. 如果你要用標準型 TFTP 伺服器, 你應該移去副檔名 **.P** (參考下一個問題的解說會更清楚).

下載 / 解壓縮時**(1)** 電腦當機

如果你用 Bootix 的 TFTP 伺服器, 試著在啓動電腦時於命令列上加入 `-s 1408 59` . 如果你不是採用增強型 TFTP 伺服器, 則移去在伺服器上 BpBatch 檔名及 `bootptab` 中的副檔名 `.P` .

細部解說: 這個問題發生在你沒有架設增強型 TFTP 伺服器但卻又使用了 `bpbatch.P` 爲你 DHCP/BOOTP 的可開機檔檔名的標籤. 當可開機檔檔名以 `.P` 爲副檔名時, BpBatch 將試著去連接到一台 Bootix 的 TFTP 伺服器. 要解決這個問題, 你可以移去可開機檔檔名中的副檔名 `.P` (這就等於告訴 BpBatch 改用標準型 TFTP 協定) 或者安裝一台增強型 TFTP 伺服器. 還有其他選擇 , 你可以安裝 Intel 的 TFTP 伺服器並變更載入器檔名爲 `bpbatch.B`. 記住: 要使 Bootix 的 TFTP 伺服器正確的在增強型 TFTP 特性下工作, 你必須在命令列中加上 `-s 1408 59`.

下載 / 解壓縮時(2) 電腦當機

能是你的電腦對 VESA 的支援有問題. 試著在命令列參數中加入 -v 或設定 VESA 變數為 "OFF".

VESA 捲頁功能故障

我們使用 VESA 1.1 的捲頁功能. 如果你的顯示卡不支援 VESA 1.1, 就別用這個功能. 如果只捲一頁, 就會發生奇怪的條紋, 別急. 這是一個已知的程式臭蟲, 我將在有空時修正 (VESA 捲頁功能不是那摸重要啦...).

快取中有壞掉的檔案

如果一個存於快取區中的檔案, 被外在的程式弄壞了. 它會自動地從快取中被移除. 如果一個正在寫入快取區的檔案沒完整寫入 (可能是電腦在檔案傳輸中被關機), 則它也會被自動清除. 但如果伺服器傳送一個已損壞的檔案. 或者發自伺服器的資料傳送取消. 則這類已毀損的檔案仍然會存在快取區中. 你只要在 `BpBatch` 第一次存取快取區時, 同時按下兩個 shift 鍵, 就可以把快取區內的資料清空. 還有一個方法, 在交談模式下輸入 `clean -1` .

在批次檔中的 EXIT 指令失效

這不是一個程式臭蟲. Exit 並不是一個指令. 沒有 exit 或 quit 指令是因為如果不開機, 則從開機腳本程序中退出是毫無意義的. 而 MrBatch 完全跟 BpBatch 一樣是同一支程式. 你可改以叫用 HdBoot 取代 exit. 這才有意義, 且在 DOS 版下, 真的會取代重開機而完全退出. 記住你可以在 DOS 版任何情況下按下 Ctrl-Break 然後退出. 這將在離開前, 回存所有被攔截的中斷.

Print 指令失效

如果你試著要印一些東西並直接進入交談模式, 你有可能來不及看到那些文字. 這是因為你所印的字是顯示在 執行時期 顯示幕而 `Interact` 指令會切換到 記錄 顯示幕. 因此只要放一個 `GetKey` 指令在 print 指令之後你就可以看到你要的文字了.

MrZip 顯示 `Malloc failed`

MrZip 需要大量的傳統記憶體才能執行. 如果你遇到這個問題, 先確定你已經用 `HideBootprom` 指令或 Bootix 的 `bputil` 卸載了 BootPROM 所占的記憶體. 如果你在單純的 MS-DOS 下執行 (非 Windows 95/98 DOS 視窗), 你應該使用 EMM386 把網路卡驅動程式載到高層記憶體以取得更多的傳統記憶體. 在 Windows 的 DOS 視窗下執行, 通常是沒舍問題的 (只要你在安裝 Windows 時, 沒留下任何 16 位元的東西在你的 `autoexec.bat` 檔中).

MrZip 在讀取目錄時失效

這個程式臭蟲已修正. 取得最新版 `MrZip`. 如果還有這個問題, 試著在把變數 `Trace` 設為 `"ON"` (且把 `PauseLog` 設為 `"OFF"`) 下, 重建你的映像檔; 這可以讓你發現導致問題的原因. 請回傳詳細的程式臭蟲報告.

MrZip 無法存取某些檔案

MrZip 會試著去讀取已鎖住, 已開啓或其他特別的檔案, 如 Windows swap 檔. 這類檔案應該不被含在映像檔中並應被過濾出來 (使用 `filter` 指令). 當然也可能是作業系統在耍你. 如果 MrZip 沒告訴你哪個檔案導致問題, 試著把變數 `Trace` 設為 `"ON"` (且把 `PauseLog` 設為 `"OFF"`) 下, 重建你的映像檔. 你也可以試著用直接磁碟存取 (也就是, 不把來源分割區當成 `"C:"` 或 `"/"` 改以 `"{:1}"` 表示). 使用直接磁碟存取通常比較慢, 因為我們的緩衝區不若作業系統大, 但它有些時候比較可靠.

磁碟映像檔總是一直由伺服器上重新載入

磁碟映像檔被存放在特殊快取區中, 如果伺服器上的沒變更, 照理說應該不需再下載一次. 但是, 由於快取區是在最後一個使用到的分割區之後啓始, 更動分割區的大小將移動快取區的位置並損

毀其內容. 另一個使檔案自快取區中消失的可能因素是先前的檔案已經變得比原來的大一倍半. 則這個檔案就應該被覆寫並重新下載一次. 但這應該不會發生. 第三個可能因素是快取區過小. 如果分割區以外所剩的可用空間小於所有壓縮映像檔的一倍半, 則只有最常使用的映像檔會在快取區中, 其他的就必需再重載一次.

Red Hat Linux 5.1 不能正常開機

這個套件假設 Linux 都用 `lilo` 開機並會檢查 `BOOT_IMAGE` 命令列參數 (在 `/etc/rc.d/rc.sysinit`). 只要把它加在 `linuxboot` 呼叫中, 或者變更你的 `rc.sysinit`.

廣播型 **TFTP** 的 **ramdisk** 當掉 (***Got in bound state***)

Linux 的 dhcp 用户程式可以透過 DHCP 回應, 動態地改變用户端的 IP 位址. 如果提供的位址值是永久的 (無限制使用時間), 則 DHCP 用户端程式只設定位址值後就結束返回 (這就是我們所期望的). 但是, 如果位址的使用時間有限制, 則用户端程式必須一直存在並每一分鐘就尋問新的位址值. 而如果 DHCP 用户端程式沒有結束返回, 則 MrBatch 就永遠不會被載入... 這問題的解法就是設定無限期的位址使用時間 (有時候就當成是 -1).

在 **BpBatch** 下會檔案存取當機, 但在 **MrBatch** 不會

這個問題發生在 AMI BIOS 日期爲 94/07/25. 我們研究後, 發覺無解. 這個問題好像是因爲 BIOS 本身有程式臭蟲 (部份暫存器或記憶體位置會被毀壞).

對分割過的檔案解壓縮失效 **(Malloc failed)**

這個問題是發生在 PXE 相容性上, 已被修正. 請取得最新版.

MrBatch 及 **MrZip** 不相容於 **RedHat 5.x** 的 **terminal**

這個問題已在 MrBatch/MrZip 八月份第 9 個版本修正. 與 RedHat 5.1 一起發行的新版 ncurses 函式庫有問題.

在 **Linux** 下出現 **"libncurses.so.3.0: cannot open shared object file"**

MrZip 已改在 libncurses 3.0 版下連結. 你只能使用其它比 3.0 版更新的 libncurses 才行. 要用較新版的 libncurses, 而你所需要去做的只是建立一些軟性連結由 libncurses.so.3.0 到你新的 libncurses.so.xx 檔案. 在 RedHat 5.1, 你可以使用下列的指令: `cd /usr/lib ; ln -s libncurses.4.2 libncurses.3.0` 你也可以下載 mrzip/mrbatch 最新的版本. 自 10/25/98, MrBatch 已在 RedHat 5.1 環境下編譯.

MrBatch 及 **MrZip** 不能在 **Linux** 下執行**(file not found)**

這個問題與前一個問題相反. 現在這個套件是預設在 libc6 環境下, 它已不能再與 libc5 一併使用. 如果你遇到這個問題, 只需要升級你的 Linux (當然, 如果我們聽到太多抱怨, 我們或許會試著保留兩種套件...).

除了預設的 **800x600 VESA** 模式, 我不能使用其他模式

你應該先檢視 `VESA-Modes` 變數的內容, 看看你的硬體設備是否可以支援你要使用的模式. 然後, 試著用下列兩種方法之一來選用你要的 VESA 模式:

- `InitGraph "mode"`: 試著用 InitGraph "1024x768", 然後執行你所喜好的初始劃圖形 (即用 `DrawGif` 指令).
- `VESA-Modes`: `VESA-Modes` 變數的第一欄是預設模式. 如果你改變 VESA-Modes 變數, 所有的初始化圖形會依循你所設定的模式.

在回存多個分割映像檔時, BpBatch 顯示 "Malloc failed" 訊息

我們已修正 BpBatch 一個記憶體對分配的問題. 你應該確定你使用的 BpBatch 版本在 1998 年九月 22 日之後.

Linux 版的 MrBatch 使用 Fullunzip 總會失效

們已於 09/22/1998 的版本中, 修正這個問題.

Scandisk 顯示我的硬碟機已損毀

10/25/98 的版本已修正一個與大映像檔有關的問題. 試著去下載最新版的 BpBatch.

我的 RedHat 開機磁碟片不能用 FloppyBoot 功能

這個軟體臭蟲已經在 10/25/98 版本中修正.

我的 FAT32 磁碟映像檔無法正常開機

這個軟體臭蟲已經在 02/09/99 版本中修正.

當你啓動 BpBatch 總是出現 "Cannot load file"

你的 DHCP 伺服器不能送出有效的 option-135 到用户端. BpBatch 使用 option 135 爲它自身的命令列參數. 如果你想要進入交談模式, 你可以設定爲 "-i". 或者你可以把它設定爲一個腳本程序檔名, 如果你想要 BpBatch 按照你所選的腳本程序執行 (別把 .bpb 的副檔名納入名稱中). 如果你使用 PXE BootPROM, 確定你使用 option-135 爲命令列參數. PXE 2.0 BootPROM 已不再支援 Option-155.

5 遠端啓動工具程式參考手冊

本節將提供我們於 日內瓦大學, CUI 爲遠端啓動架構, 所發展的工具程式的詳細說明資訊.

5.1 BpBatch, MrBatch 及 MrZip

這三個名稱是爲了給同一個程式賦予三種變化, 分別有下列的特性:

- `BpBatch` 是一個特別的程式, 可以在作業系統載入之前自 BootPROM 啓動. 它是由兩個部份來完成: `bpbatch.P`, 動態的載入器, 及 `bpbatch.ovl`, 程式本體. `BpBatch` 已可以透過我們自行實作的 FAT16, FAT32 及 Ext2fs 檔案系統達成完整的磁碟存取功能, 更可透過 BootPROM TFTP API 達到遠端網路 I/O 存取能力. `BpBatch` 是在 DOS 下以 Borland C 5.0 及 Turbo Assembler 3.2 編譯.
- `MrBatch` 是 `BpBatch` 在 DOS/Linux 下的版本. 所有 `BpBatch` 可以辨識的指令都可以在 `MrBatch` 下使用. 如果你要在 DOS/Linux 下測試你的批次檔腳本程序, 這會相當有用. 在 DOS 下, `MrBatch` 可以作業系統爲基礎, 模擬遠端 I/O 功能, 如果 BootPROM 不存在的話. 在 Linux 下, BootPROM 亦無法被偵測到, 但 `MrBatch` 可以用 Linux 的 IP 支援功能以模擬它, 或採用作業系統爲基礎的檔案存取來模擬. `MrBatch` 在 Linux 下, 用 GCC 2.7.2.1 編譯; 而 DOS 下,則用 Borland C 5.0 及 Turbo Assembler 3.2.
- `MrZip` 是一個解譯器用來辨識 `MrBatch` 語言的擴增功能, 並能建立磁碟映像檔. 在 `MrZip`, 原本有限制的遠端檔案 I/O 已由完整的作業系統爲基礎的檔案存取功能取代. `MrZip` 並不支援 VESA. `MrZip` 在 Linux 下, 採用 GCC 2.7.2.1 編譯; 而 DOS 下, 採用 Borland C 5.0 及 Turbo Assembler 3.2.

5.1.1 命令列參數

所有的程式都可以接受相同的參數用法. MrBatch 及 MrZip 可在命令列下取得參數, 而 BpBatch 則藉由參考 BOOTP 通訊協定的 option 155 (十進制) 取得. 這是參數用法 :

```
[-x] [-l] [-b] [-v] [-w] [-i] [script-basename]
```

說明:

- -x 不使用延伸記憶體.
- -l 不使用 ISO-latin-8859-1 爲預設字元集.
- -b 取消對 BootPROM 的偵測 (在 DOS 環境下, 會變成搜尋軟碟機).
- -v 取消對 VESA 的偵測 (在 Windows 95/98 環境下, 會切換成全螢幕).
- -w 啓用直接磁碟寫入 (在 DOS 及 Linux 下, 預設是不使用).
- -i 啓用交談模式, 即使有設定腳本程序檔名也一樣.

script-basename 是附加的選項. 如果有設定, MrBatch 及 BpBatch 會載入以 .bpb 爲副檔名的檔案, 而 MrZip 會載入以 .mrz 爲副檔名的檔案. 如果沒設定, MrBatch 及 MrZip 會執行於交談模式, 而 BpBatch 會載入與 BOOTP 可開機檔名同主檔名, 但以 .bpb 爲副檔名的檔案.

5.1.2 語法

以下的規則用於 BpBatch 分析輸入行.

- 所有指令都是一行一行解釋. 各行間以 CR 及/或 LF 爲分隔.
- 每行最長爲 255 個字元.
- 關鍵字及變數都要區分大小寫.
- " 用以當作字串識別字元
- 當使用了 ${變數} 或 $變數, 則會以變數的值來取代, 若未定義則會是一個空字串. 這個方法也可以在字串中使用. 另外, 如果是用在字串值時, 必須在字串前後加上雙引號 (即 這裡稱作 巨集式運算 會比稱作 變數 來得貼切).
-
    - \a 是代表會發出 嗶聲 (bell) 字元 (ASCII 7).
    - \b 是代表 倒退鍵 (backspace) 字元 (ASCII 8).
    - \n 是代表 換行 (newline) 字元 (ASCII 10).
    - \r 是代表 歸位 (return) 字元 (ASCII 13).
    - \t 是代表 跳位 (tabulation) 字元 (ASCII 9).
    - \v 是代表 横向跳位 (vertical-tab) 字元 (ASCII ...).
    - \nnn n 是三位數的十進位數值在 000 到 377 代表 ascii 碼的設定值.
    - \X X 是任何上面沒列出的字元就由 X 代表. 特別是:

    - \" 是代表常見的 雙引號 (double-quote) (不是字串識別字元).
    - \$ 是代表常見的 錢字號 (dollar sign) (不是變數標示).
    - \\ 是代表常見的 反斜線符 (backslash) (不是特殊字元).

- 字元 "end of string" (ASCII 碼 0) 不能在任何地方使用. 因它是程式內部用來當作字串結束的識別字元.
- 字元 "floating diaeresis" (ASCII 碼 十進制 249, 十六進制 F9, 八進制 371) 不能被用在任何字串中. 因爲它是程式內部用來辨識輸入機制的識別字元.
- 字元 "block space" (ASCII 碼 十進制 255, 十六進制 FF, 八進制 377) 不能被用在任何變數中. 因爲它是程式內部變數的識別字元.

空行會被忽略. 該行由 (#) 啓始則會被當成註解, 不做任何解釋. 該行由 (:) 啓始則會被當成標籤, 不做任何解釋.

字串表示

字串由雙引號括起來的可被識別到:

```
"Hello world"
```

如要在字串中放入雙引號, 則必須用, 反斜線符 (backslash) 括起來:

```
"I said: \"Hello world\""
```

字串可以用一些運算子來做預先修正.

- 字元取代運算子:

```
"Hello world"/o=u/      ==      "Hellu wurld"
"198.76.54.32"/.= /     ==      "198 76 54 32"
```

- 文字選擇運算子 (以零爲起始):

```
"Hello world"{0}        ==      "Hello"
"198 76 54 32"{1-3}     ==      "76 54 32"
```

- 子字串選擇運算子 (以零爲起始):

```
"Hello world"[4]        ==      "o"
"Hello world"[4-7]      ==      "o wo"
```

運算子可以一再使用以修正出所需的字串樣式. 如果想知道關於字串長度及文字個數的運算子, 可以參考下面 "數值表示法" 一節.

數值表示法

數值表示是以 32 位元整數來運算 (由 -2,147,483,646 到 2,147,483,647). 十六進制, 八進制, 及二進制 數值則無法辨別. 如果一遇到數值表示, 則必要經過下面的辨識程序:

- 是 正整數 或是 負整數
- (*expr1 op expr2*) 這種形式的 *op* 可以是 +, -, * (乘), / (除) 或 % (餘數) 而 expr 是數值表示. 要注意的是每個運算都要用括號包起來 :

```
((3 * 5)+2)              == 17
```

- 字串長度運算子 (@), 後面接著一個字串：

```
@"Hello world"           == 11
```

- 文字計算運算子 (#), 後面接著一個字串：

```
#"Hello world"           == 2
```

延遲

部份指令須要延遲來當作參數. 延遲時間是以秒爲單位, 精度到十分之一秒:

Delay 3	等待 3 秒
Delay 0.3	等待 3/10 秒

顏色

需要設定顏色時, 你可以用數值或顏色名 (區分大小寫) 來代表顏色. 可以使用下列顏色:

Black	0
Blue	1
Green	2
Cyan	3
Red	4
Magenta	5
Brown	6
LightGray	7
DarkGray	8
LightBlue	9
LightGreen	10
LightCyan	11
LightRed	12
LightMagenta	13
Yellow	14
White	15

參考檔案

檔名是一種字串. 他們必需以雙引號括起來. 檔案名稱在大小寫有別的檔案系統時, 就有大小寫的分別; 反之, 於無大小寫之別的檔案系統時, 就有無大小寫的分別. 斜線符 與 反斜線符 可以自由的擇一使用. 但別忘記 要用到溢位字元 反斜線符 必需用 雙反斜線 來表示.

有兩種類型的檔案參考:

- 直接參考到本機磁碟機上的檔案.
- 外部的檔案.

直接參考到本機磁碟機上的檔案用法:

```
"{磁碟機號碼:分割區}/絕對路徑/檔案名稱"
```

可以省略磁碟機號碼預設值是零. 比如, "{:1}/usr/bin" 是指在第一個分割區有 /usr/bin 這目錄. 直接參考到本機磁碟上檔案的 I/O 功能完全是我們自行實作出來的 (我們不使用作業系統所提供的功能).

有兩種 特別的 分割區. 分割區零 是指磁碟機上的主開機記錄 (MBR) 並提供一個虛擬檔案系統裝置以讓你存取開機識別碼. 分割區負一 (-1) 是指快取檔案系統 (參考下面敘述).

BpBatch/MrBatch 環境下, 如果有 BootPROM 存在, 則外部的檔案是指到遠端 TFTP 伺服器上:

"help.bpb"	是指在 /tftpboot 下的 help.bpb 檔案
"images/MyImage.pcx"	是一個在 /tftpboot/images 下的檔案

如要參考到其他的 TFTP 伺服器可以用:

```
"198.76.54.32:help.bpb"
```

如果伺服器要繞過閘道器可以用 :

```
"198.70.0.1/198.76.54.31:help.bpb"
```

亦可以自行決定特殊埠號作 TFTP 連結 :

```
"198.76.54.32@89:getpasswd/smith"
```

在同一時間只能有一台機器開啓遠端檔案. 如果 BootPROM 不存在, 遠端的檔案會透過作業系統的檔案 I/O 功能來模擬, 但同樣的限制依然存在.

在 MrZip 環境下, 遠端的檔案是透過作業系統存取到該檔. 這裡就沒有任何限制, 遠端的檔案可被當成直接參考到本機磁碟檔案一樣來使用. 而遠端的檔案通常會比直接參考到本機磁碟檔案還快, 因爲作業系統本身有較多的暫存區. 如果作業系統有支援, 則遠端檔案可被當成網路上的檔案.

```
"C:\\autoexec.bat"
"C:/config.sys"
"/mnt/net/usr"
```

5.1.3 快取檔案系統

爲了降低網路負載並加快 開機速度, 磁碟檔案存取速度, Linux Kernel 及其他檔案可以由硬碟裏快取得到. 磁碟快取的區域是在硬碟機的末端, 介於分割表中最後一個對應到的磁柱與磁碟最後一個理論磁柱之間 (於任何對應要使用的分割區之外). 要使快取檔案系統作動, 就必需在最後一個分割區到磁碟機末端之間保留空間. 而如果你要回存一個磁碟映像檔, 則更要讓快取檔案系統作動才行.

這個磁碟快取是由一個具 CRC 確認檢查的變易性檔案系統所組成: 每個目錄 及 每 32 KB 的資料區塊都以一個 32 位元的 CRC 來檢查確認. 每當一個目錄或資料區塊無預警的改變後, 則該檔案會自動由快取區移除; 並在下次存取要求時, 再次下載新的資料以吻合現況.

你可以透過 BpBatch, MrBatch 及 MrZip 自由存取快取檔案系統, 用直接存取本機磁碟檔案功能存取特別分割區 "{:-1}" . 要檢視快取區內容, 只需輸入 :

```
logdir "{:-1}"
```

如果快取區資料毀損且不會自動清理時 (這狀況應該不會發生), 你只要輸入 (在交談模式下) :

```
clean -1
```

或在 BpBatch 第一次存取快取時, 同時按下兩個 shift 鍵.

5.1.4 特殊變數

有些變數在初值設定時具有特別的意義. 一部份是存在於所有的程式, 另一部份只可以在 MrZip 執行時才存在, 但仍有一部份只能在 BOOTP/DHCP 回應收到的時候才存在.

通用變數

- `$Program` 在 BpBatch, 是設成 "BpBatch", 在 MrBatch, 是設成 "MrBatch", 在 MrZip, 是設成 "MrZip".
- `$Basename` 用來設定給批次檔解譯器使用的腳本程序檔的主檔名.
- `$HelpFile` 當呼叫到 `Help` 時, 所要載入的檔案名稱. 預設: `"${Basename}.hlp"`.
- `$BOOTP-...` 是經由 BOOTP/DHCP 回應所取得的設定. (參考 BOOTP/DHCP 變數那一段, 已取得更細部的資料).
- `$DHCP-...` 是經由 DHCP 回應而取得的設定. (參考 BOOTP/DHCP 變數那一段, 以取得更細部的資料).
- `$Disks` 是用來設定各磁碟機所劃分的大小空間分布. 意謂著, 以 `#"$Disks"` 象徵硬碟機個數; 而 `"$Disks"`{0} 則是指第一台硬碟機的大小.
- `$Keypressed` 是指鍵盤暫存區中, 下一個準備讀取的輸入鍵值 (如果有輸入的話).
- `$LBA` 控制是否使用 LBA 來存取 $>$ 2Gb 的硬碟機. 預設: "ON".
- `$FDA` 控制是否使用 快速磁碟存取 (透過磁柱寫入). 預設: "ON".
- `$VESA` 控制是否使用 VESA 圖形化模式. 預設: "ON" 如果系統有此功能.
- `$VESA-Modes` 列出所有可供使用的 VESA 模式. 第一行即爲預設模式, 用於不具任何參數的 InitGraph 指令時, 選用. 記住: 如果 VESA="OFF", 此變數會是空的.
- `$APM` 如果你的電腦支援 Avanced Power Management 會設成 "ON". 如果 $APM 設成 "ON", 你就可以使用 PowerOff 指令來關掉電腦電源. 預設: 與你的硬體設備有關.
- `$Trace` 控制是否在執行前, 顯示每個指令. 它也可以控制是否要顯示所建立的新檔檔名. 預設: "OFF".
- `$AutoShowLog` 控制是否在 ESC 鍵按下後, 自動切換到文字記錄. 預設: "ON".
- `$PauseLog` 控制是否在顯示記錄時有跳頁暫停功能. 預設: "ON".
- `$CacheDisk` 用來設定磁碟是否有快取遠端檔案的功能. 預設: empty == 0, 第一台硬碟機.
- `$CacheAlways` 控制是否自動快取遠端複製過的檔案, 或已修正過的 或 製出來的映像檔. 預設: "OFF".
- `$CacheNever` 防止任何檔案被快取. 在無磁碟機的 Linux 開機時請把這個變數啓動. 預設: "OFF".
- `$CacheReserve` 防止在快取分割區中, 使檔案超過所需空間的百分之 25. 如果你的硬碟空間短少, 關掉這個變數. 預設: "ON".
- `$ExtMemory` 控制是否使用延伸記憶體 (或稱爲 XMS). 一經選用, 延伸記憶體就不能被再利用. 預設: "ON" 如果有延伸記憶體的話.
- `$IsoLatin` 控制在含入檔與修正檔中, 高層 ASCII 碼的解釋方式.IsoLatin 的設定是在檔案載入後才處理的, 不是在檔案處理之後. 預設: "ON".
- `$ProgressX` 及 `$ProgressY` 控制檔案下載與解壓縮正在處理中的視窗, 於 VESA 圖形模式下的顯示位置. 預設: 200 200.

- `$EXT2-Backup` 控制 Linux Ext2 檔案系統的 superblock 備份的更新. Superblock 備份需花掉幾秒鐘在現行的 Kernel 中不曾被使用 (只靠 e2fsck).
- `$WatchdogPort` 設定 I/O 埠 (十進制值) 用以存取 watchdog 介面卡. 預設此變數是 0, 也就是關掉 watchdog 的支援. 設定這個變數爲非零值時, 會啓用 watchdog 支援.
- `$Security-Gateway` 控制是否使用 安全認證閘道伺服器 作使用者身份認證. 我們的特殊身份認證閘道器必須在目標電腦上執行才行. 預設: `"${BOOTP-伺服器-IP}@89"`. (即. TFTP 伺服器, 在 89 埠).
- `$Security-Check` 內含來自 安全認證伺服器 最後一次檢查的結果, 不是 PASSED 就是 FAILED. 預設: "FAILED".
- `$Security-Passwd`, `$HelpTopic`, `$OnExit`, `$OnKey-...`. 只在內部使用.

參考 BOOTP 變數 與 MrZip-規範的變數.

MrZip-規範的變數

以下是只能在 MrZip 使用的變數.

- `$TempPath` 控制暫存檔所存放的目錄位置 預設: <empty> == 現行目錄.
- `$DumpFormat` 當有這種必要時, 控制檔案傾印到記錄檔的方式. 它是一個含下列內容的字串
  - "h"/"H" 顯示檔頭.
  - "b"/"B" 用以 總結(summarize)/傾印(dump) 開機磁區.
  - "s"/"S" 用以顯示一個 短/長 的對應摘要.
  - "d"/"D" 用以顯示一個 短/長 的目錄列表.
  - "f"/"F" 用以 總結(summarize)/傾印(dump) 檔案.

  預設: "hbD".
- `$FragmentSize` 控制分割檔案的大小. 如果你不是使用 Bootix 的 增強型 TFTP 伺服器, 你應該設定這個爲 "30 MB". 預設: "87 MB".
- `$SourceArchive`, `$DestArchive`, `$Filter...` 只在內部使用.

BOOTP 變數

以下的 BOOTP-... 及 DHCP-... 變數在 BOOTP/DHCP 回應時才具有實際的意義 (前提是 TCP/IP BootPROM 必需已被偵測到):

```
$BOOTP-Client-ID
$BOOTP-Your-IP
$BOOTP-Server-IP
$BOOTP-Gateway-IP
$BOOTP-Bootfile
$BOOTP-Server-Name
$BOOTP-Subnet-Mask
$BOOTP-Time-Offset
$BOOTP-Routers
$BOOTP-Time-Servers
$BOOTP-Name-Servers
$BOOTP-Domain-name-Servers
$BOOTP-BOOTP-Log-Servers
```

```
$BOOTP-Cookie-Servers
$BOOTP-Lpr-Servers
$BOOTP-Impress-Servers
$BOOTP-Resource-Location-Servers
$BOOTP-Host-Name
$BOOTP-Boot-Size
$BOOTP-Merit-Dump
$BOOTP-Domain-Name
$BOOTP-Swap-Servers
$BOOTP-Root-Path
$BOOTP-Extensions-Path
$BOOTP-IP-Forwarding
$BOOTP-Interface-MTU
$BOOTP-All-Subnets-Are-Local
$BOOTP-Broadcast-Address
$BOOTP-NIS-Domain
$BOOTP-NIS-Servers
$BOOTP-NTP-Servers
$BOOTP-Font-Servers
$BOOTP-X-Display-Manager
$DHCP-IP-Address-Lease-Time
$DHCP-Message-Type
$DHCP-Server-Identifier
$DHCP-Message
$DHCP-Renewal-Time
$DHCP-Rebinding-Time
$BOOTP-NIS+-Domain
$BOOTP-NIS+-Servers
$BOOTP-Server-Name
$BOOTP-Bootfile
$BOOTP-Mobile-IP-Agent
$BOOTP-SMTP-Servers
$BOOTP-POP3-Servers
$BOOTP-NNTP-Servers
$BOOTP-WWW-Servers
$BOOTP-Finger-Servers
$BOOTP-IRC-Servers
$BOOTP-StreetTalk-Servers
$BOOTP-STDA-Servers
```

其他的 BOOTP/DHCP 變數可以用以下的名稱儲存

```
$BOOTP-Option-n
```

n 是十進制數值代表 BOOTP 選項號碼.

別把 `BOOTP-Gateway-IP` 搞混了, 它是 TFTP 通訊協定的閘道器, 如果用户端與 TFTP 伺服器在同一個子網域, 則應設爲 0.0.0.0 , 而 `BOOTP-Routers`, 內含 預設 IP 閘道器. TCP/IP BootPROM

有時會透過 `BOOTP-Routers` 設定 `BOOTP-Gateway-IP` 的值, 乃因爲每個 TFTP ack 封包都會先送往 路由器. 爲了避免這種行爲, 如果你的 TFTP 伺服器與用户端在同一個子網域, 則強迫把 `BOOTP-Gateway-IP` 設成 `0.0.0.0` (感謝 Maciek Uhlig 指出這個問題).

5.1.5 監控指令

本節列出用來監視系統狀態的指令. 額外的參數也將列入 (我比較喜歡使用 方括 來註明, 但 LaTex 不喜歡在這看到他們...)

Interact

顯示記錄資料並進入交談模式直到 QUIT 或 EXIT 輸入. 在離開交談模式之前先輸入HideLog, 以防止記錄的訊息干擾批次檔的執行.

Help (主題)

載入線上說明檔 (`bpbatch.hlp`) 並顯示所給定主題的內容. 如果沒設定主題, 或給定的主題找不到, 顯示說明索引.

Log "字串"

顯示字串到記錄檔. 不加入 游標 返回/換行 字元.

Echo "字串"

顯示字串到記錄檔並跳至下一行. 相同於

```
Log "text\r\n".
```

LogVars ("pattern")

記錄 (即. 顯示到記錄檔中) 所有合於所給定樣式的變數. 樣式可以使用萬用字元 (? 及 *).

範例: `LogVars "BOOTP-*"` 列出所有 `BootP` 變數

LogDir "path/pattern"

記錄 (即. 顯示到記錄檔中) 所有合於所給定樣式的檔案. 樣式可以使用萬用字元 (? 及 *).

範例: `LogDir "/usr/g*p"` 列出所有檔名爲 `g...p` 的檔案

LogTree "路徑"

記錄以給定路徑爲根目錄的樹狀目錄結構.

LogFile "檔案名"

記錄檔案的內容. 檔案大小必需小於 64 KB.

ShowLog

使隱藏的記錄檔顯示. 在 "$AutoShowLog" == "ON" 並按住 ESC 狀況, 或交談模式下, 會自動啓動.

HideLog

防止記錄的訊息出現在螢幕上. 當 BpBatch, MrBatch 及 MrZip 以執行腳本程序檔時爲預設狀態.

CaptureLog

錄下所有的記錄訊息到 64 KB 的暫存區直到 EndCapture 出現. 記錄輸出超過 64KB 會清除暫存區資料. 這個指令用來產生制式內容的文字檔. EndCapture 指令必須在同一個腳本檔案中出現.

EndCapture ("檔名")

束截取記錄訊息. 如果有給定檔名, 會把截取到的文字存到該檔. 否則, 會忽略掉它.

Beep

發出一個聲音. 這個指令是同於 Echo "\007".

5.1.6 控制指令

本節列出批次執行的控制指令. 額外的參數也將列入.

Include "filename"

載入所給定的檔案並對它做語法分析. 在此引入的檔案執行結束後返回此載入點. 在此引入檔中, ASCII 127 以上的字元的意義, 由載入此檔時 $IsoLatin 的值來決定.

OnExit *command*

設定一個外部處理器. 它會在現行批次檔終了時自動執行.

Goto *label*

移到所指的標籤 (label) 執行 (即. 以 :*label* 爲起始的行)

Eval "指令"

執行 "指令" 所指的替代命令並對其做語法分析.

If ...

```
If (not) <expr1> (==|!=|<|>|>=|<=|=>|=<|<>) <expr2> <command>
If (not) (ci) "str1" (==|!=|<|>|>=|<=|=>|=<|<>) "str2" <command>
If (not) (ci) "str1" Match-Expr "樣式" <command>
If (not) (ci) "str1" Match-Passwd "unix 密碼" <command>
If (not) (ci) "str1" in "wordlist" <command>
If (not) (ci) "str1" in-file "檔名" <command>
If (not) exist "檔名" <command>
If (not) valid <硬碟機>:<分割區> <command>
```

如果條件測試成立; 這些指令用來執行指令 *command;* . 第一種格式用來比較兩個以數字來表示的運算式. 第二種格式用來比較兩個字串, 可選擇不區分大小寫. 第三種格式用來測試 "str1" 是否合於樣式 (允許使用萬用字元). 第四種格式用來測試明碼 "str1" 是否經 Unix 的編碼系統後結果,能吻合. 第五種格式用來測試 "str1" 是否被引入在文字列表中. 第六種格式用來測試 "str1" 是否被引入在文字檔案中. 第七種格式用來測試所給定的檔案是否存在. 第八種格式用來測試所給定的分割區是否有效 (即. 是否已格式化過). 這個格式只在 BpBatch 1999年 二月後的版本才有.

Set ...

```
Set variable = "string-value"
Set variable = <expr>
```

設定變數值. 如果給定的是數值運算式, 它將間接轉化爲字串形式. 所定義的變數可以在任何地方以 $variable 或 ${variable} 的形式來參考使用. 如果參考的結果要當成字串, 則必須用雙引號把變數括起來: "$variable" 或 "${variable}".

Delay 週期時間

等待直到所指定的時間耗盡 (以秒爲單位). 參考時間表示格式那一章節.

GetTime 變數**, GetDate** 變數

取得 CMOS 的時間資訊並把它存到 變數 中, 格式爲 HH:MM:SS. 取得 CMOS 的日期資訊並把它存到 變數 中, 格式爲 YY/MM/DD. 這樣就可以依時間或日期來調整你的開機程序內容.

SetTime "HH:MM:SS", SetDate "YY/MM/DD"

設定用户端電腦的 CMOS 時間或日期爲所給定的值. 如果你有一台安全認證閘道器 (security gateway) (或我們的特殊型 TFTP 伺服器), 你可以透過下列指令在每次開機時, 自動調整 CMOS 時間與日期資訊:

```
include "$Security-Gateway:gettime"
```

如果你想知道這個指令會做哪些事, 只需輸入:

```
logfile "$Security-Gateway:gettime"
```

Poweroff

關掉電腦電源. 這個指令只能在具 Advanced Power Management (APM) 相容的電腦上使用.

5.1.7 與鍵盤相關的指令

本節列出一些指令讓你可以監控鍵盤的輸入. 額外的參數也將列入. 亦可參考後面提到的 國際化語系支援 一節.

GetKey (變數**)**

一直等到鍵盤按下某鍵並把它存到 變數 中.

WaitForKey 週期時間 **(**指令**)**

於所指定的 週期時間 內等待鍵盤按下某鍵單位爲秒. 如果時間內沒按下任何鍵, 則執行 指令. 否則, 保留鍵值在鍵盤暫存區中. 亦可參考時間格式那一段.

Input (變數 **(**最大字串長度**))**

自鍵盤讀取一個以 return 爲結尾的字串, 並把它存在 變數 (不會把 return 存進字串). 如果有給定 最大字串長度 , 則不允許使用者輸入字元超過這個長度.

亦可參考 與安全性有關的指令 一節中, 的 `GetPasswd` 指令.

OnKey "c" 指令

設定 當 "c" 某個單一字元被按下時, 會自動去執行所給定的 指令 (例外情況爲 正使用到 GetChar 或 某個 Input 指令, 而系統處於等待輸入的狀態). 如果用 `"default"` 字串來取代單一字元 "c", 則指令會在按下任意鍵後就執行.

5.1.8 文字輸出指令

本節列出用來輸出格式化文字的指令. 所有提到的指令都可以在圖形模式下使用, 可以得到相同的功能 (在文字環境下可以提供 80x25 字元模式, 而圖形化環境則可題供 100x37 字元模式, 因爲圖形化環境每個字元大小是 8x16). 額外的參數也將列入. 亦可參考下文的 國際化語系支援.

Print "字串"/運算式

在目前游標所指的位置印出字串/運算式, 並採用目前的文字屬性並移動游標. 加入到字串中 "\r\n" 以換行.

TextAttr 前景顏色 背景顏色

設定文字屬性. 也可以用一個數值來設定顏色方法是 16*背景顏色+前景顏色.

如果你還想更炫, 可以用 `LoadFont` 指令. 參考下文的 國際化語系支援.

At ***line*,*col*** **(*command*)**

移動游標位置到所指定的位置並執行命令. 如果有設定的話.

範例範例: `At 10,20 Print "Gnats and rats !"`

Clear (顏色 **(**文字樣式 **(**頂部,左限,底部,右限**)))**

以所給定的 文字樣本 填滿所定義的文字區域 (可以是字串或十進制的 ascii 碼). 預設的文字區域爲全螢幕, 而文字樣式爲 全黑 (ASCII 十進值 219) 且顏色是黑色 (清除螢幕). 移到游標到所清除區域的左上角.

與舊版的 **BpMenu** 相容的指令

```
.ATT (<屬性>)
.CLS (<屬性>)
.DEF <鍵值> (<等待時間>)
.KEY <鍵值> <檔名>
.POS ((< x 軸位置 >) < y 軸位置 >)
.PWD <鍵值> <編碼過的密碼>
.WLN (<字串>)
.WRT <字串>
```

參考 Bootix 的使用手冊可以獲得更多訊息. 我們先前寫了一個程式 *program* 用來編輯選單檔案. 就是用這個語法, 但使用新的語法來建立選單是比較好. 需注意 .PWD 指令並沒被實作出來. 因爲我們根本不知道 Bootix 的編碼法則.

5.1.9 圖形輸出顯示指令

本節將列出在圖形模式下, 如何輸出圖形到螢幕上. 這章節中所列出的功能與所給定的像素相關. 當然你也可以在圖形模式下, 使用所有文字輸出的指令 (參考前文). 額外的參數也將列入.

需注意圖形模式會在使用到與圖形相關的指令時自動啓用, 除非設定 `VESA` 變數爲 `"OFF"`.

InitGraph ("模式")

啓用 VESA 圖形模式. 原點在螢幕左上角 (0 0). 在 Windows 95/98 下使用 VESA 模式可能會當機. 在使用 MrBatch 時, 加上 -v 選項以避免發生這個問題.

你可以要求使用某個特定的顯示模式, 如果有給定 "模式" 這項參數. 這個參數是附加的: 如果你沒做任何定義, 則會從選用 VESA-Modes 變數中第一行的顯示模式.

用的模式有 :

- 640x480 => 640 乘 480 相素, 256 色.
- 800x600 => 800 乘 600 相素, 256 色 (預設模式).
- 1024x768 => 1024 乘 768 相素, 256 色.
- 1280x1024 => 1280 乘 1024 相素, 256 色.

VESA-Modes 變數列出你的硬體設備能支援的模式.

範例: `InitGraph "640x480"`

CloseGraph

關掉 VESA 圖形模式返回文字模式.

DrawBar ***x*-軸位置 *y*-軸位置 寬度 高度 顏色**

VESA 繪圖功能. 以給定的大小與顏色畫出一個實心長條圖.

DrawWindow ***x*-軸位置 *y*-軸位置 寬度 高度 (背景顏色 (長條圖顏色)) ("主題文字" (主題文字顏色))**

VESA 繪圖功能. 以給定的大小與顏色畫出一個視窗. 預設的背景顏色是 淡灰色 (LightGray) 而主題部分是藍色 (Blue). 如果你引入一個主題字串及顏色, 則文字會顯示在 主題那部份.

Drawtext ***x*-軸位置 *y*-軸位置 "字串文字" (前景顏色)**

VESA 繪圖功能. 在給定位置顯示字串文字並採用透被明背景. 預設爲前景文字的顏色.

DrawPcx "pcx-圖檔名" (*x*-軸位置 *y*-軸位置 (顏色用法))

VESA 繪圖功能. 載入所給定的 256-色 PCX 圖檔並顯示到螢幕上. 預設的顯示位置在螢幕左上角的 (0 0).

顏色用法 定義當使用超過 256 種顏色時, 所對應的調色盤 (比如要同時顯視兩個 256 色的圖檔):

- `Best-Colors` 對最後載入的圖檔顯示最佳顏色
- `Spare-Colors` 試著避免顏色對應, 只改變現在使用的顏色
- `Share-Colors` 試著避免顏色對應, 延用現在使用的顏色
- `Reuse-Colors` 不對應新的顏色, 只使用現在可用的顏色

預設是使用 `Best-Colors`.

DrawGif "gif-圖檔名" (*x*-軸位置 *y*-軸位置 (顏色用法))

VESA 繪圖功能. 載入所給定的 256-色 GIF-87a 圖檔並顯示到螢幕上. 這個檔案可以是交錯的, 但必需是 GIF-87a 格式 (非 GIF-89a). 圖檔的大小會修正到合於目前所使用的顯示模式. 但你不能載入一個 1024x768 GIF 圖檔到 640x480 圖形模式下. 預設的顯示位置在螢幕左上角的 (0 0).

因爲 LZW 專利問題, 這項功能在商業版的 BpBatch/MrBatch 不提供. 請改用 DRawPcx 指令.

顏色用法 定義當使用超過 256 種顏色時, 所對應的調色盤 (比如要同時顯視兩個 256 色的圖檔):

- Best-Colors 對最後載入的圖檔顯示最佳顏色
- Spare-Colors 試著避免顏色對應, 只改變現在使用的顏色
- Share-Colors 試著避免顏色對應, 延用現在使用的顏色
- Reuse-Colors 不對應新的顏色, 只使用現在可用的顏色

預設是使用 Best-Colors.

5.1.10 與安全性有關的指令

本節將列出用來認證使用者身份的指令. 額外的參數將列在段落間.

部份的功能可與 安全性閘道器 *(security gateway)* 一併使用, 所以你必須先安裝它. 參考 特殊 *TFTP* 伺服器 一節以獲得更多資訊.

GetPasswd (變數 (密碼最大長度))

就跟 Input 指令一 樣, 但會在輸入字元時改顯示星號.

Crypt "字串" "salt" 變數

以 Unix 編碼功能對所給定的 8 個字元加密, 並把加密後結果以字串形式存到 變數. 而 "salt" 通常是兩個可以在加密後字串中找到的前兩個字.

要注意 Unix 編碼系統是單向的. 無法對已編碼過的字串解碼還原. 只能試著用相同的 salt 比較其編碼過的字串是否一致.

這個功能是用 BSD 實作的 DES 編碼系統. 這是 BSD 版本的標頭: `Copyright (c) 1989 The Regents of the University of California. This product includes software developed by the University of California, Berkeley and its contributors.`

DESCrypt "字串" "鍵值" 變數

對給定的字串加密用給定的 8 個字元 鍵植來編碼並以十六進制格式的字串存入 變數.

這個功能是用 BSD 實作的 DES 編碼系統. 這是 BSD 版本的標頭: `Copyright (c) 1989 The Regents of the University of California. This product includes software developed by the University of California, Berkeley and its contributors.`

DESDecrypt "hexcode" "key" 變數

用給定的 8 個字元來對十六進制字串解密並把結果存到 變數.

這個功能是用 BSD 實作的 DES 編碼系統. 這是 BSD 版本的標頭: `Copyright (c) 1989 The Regents of the University of California. This product includes software developed by the University of California, Berkeley and its contributors.`

MD5 "字串" 變數

用 MD5 對給定字串編碼後的檢查值以十六進制的字串格式存到 變數. 對於密碼大於 8 個字元的, 可採用的 Unix 編碼功能.

這個功能是 MD5 標準的實作. 這是它的版權頁標頭: `RSA Data Security, Inc. MD5 Message-Digest Algorithm`

CheckUser "使用者名稱" "密碼" "領域"

連接到 $Security-Gateway 並檢查所給定的使用者是否存在所給定的 radius 領域並檢查密碼是否吻合. 如果領域是 "Unix", 則使用於 安全認證閘道器上, Unix 的 使用者/密碼. 若在其他領域, 使用安全認證閘道器領域定義檔, 以偵測正確的 Radius 或 NT 領域.

設定 $Security-Check 的值爲 "PASSED" 或 "FAILED". 密碼不會以明碼在網路傳輸.

5.1.11 與磁碟相關指令

本節將列出用來調製硬碟資料的指令. 額外的參數也將列入.

GetPartitions 變數 (磁碟機名)

讀取所給定的磁碟機的分割表並以字串格式存到所給定的 變數. 所存的字串是硬碟內空間分佈的列表 檔案形式:空間大小, 其中

- 檔案形式 有 FAT16, EXT, BIGDOS, NTFS, FAT32, FAT32-LBA, BIGDOS-LBA, EXT-LBA, LINUX-SWAP, LINUX-EXT2 或其它未知的十進值檔案系統識別碼.
- 空間大小 分割區空間大小以 megabytes 爲單位.

參考 SetPartitions 以獲得更多與分割區有關的資訊.

SetPartitions "分割區" (硬碟機名)

依字串內容設定分割表. 個式同於 GetPartitions 指令. 這個指令也同時會重置所有的開機旗標 (提示: 使用 SetBootPart).

在主開機紀錄 (MBR) 中的主要分割區只能有四個. 而且, DOS 及 Windows 在主分割表中, 只能允許一個 FAT 分割區 (稱爲 主要分割區, C:). 其他的 FAT 分割區只能分佈在一個延伸分割區裡 (而他們稱爲邏輯分割區). 如果我們把 MBR 分割表中的分割區編號爲 1-4 , 則 5-8 爲存於第一個延伸分割區的邏輯分割區, 定義兩個 FAT 分割區可以如此做: 分割區 1 爲 FAT, 分割區 2 爲 EXT, 分割區 5 爲 FAT. 分割區 3,4,6,7 及 8 則爲 UNUSED. 同樣的劃分方法可以採遞迴的方式設定出兩個以上的 FAT 分割區: 在分割區中設定一個延伸分割區 6 並加入一個 FAT 邏輯分割區 9 .

在大部份較嚴緊的 DOS 規範中, 分割表第 3 和 第 4 個分割區將不被使用. 事實上, 部份 DOS 版本及其他作業系統都能允許分割表中超過兩個分割區, 但並沒有相當明確的規定. 就這方面的問題而言, BpBatch 對分割表的意義有較彈性的解釋, 它可以了解作業系統無法知道的部份.

一個通用的規則就是不要在分割表中設置超過兩個延伸分割區, 否則分割區編號的劃分方式將被破壞.

如果你想測試一些有趣的設定, 以增加經驗. 那就不要抱怨作業系統無法分辨你的分割區. 保證可以正常工作的劃分方式是以主要分割區存放作業系統的開機分割區, 把其他分割區一併擺到延伸分割區.

延伸分割區設定的範例如下:

```
SetPartitions "BIGDOS:100 EXT:400 EMPTY EMPTY BIGDOS:400"
```

GetBootPart 變數 (硬碟機名)

取得具可開機旗標的分割區號碼 (DOS 的說法是: 已作用的主分割區) 並把它存到 變數 中. 第一個分割區編號是 1. 如果沒有具可開機旗標的分割區, 則回應 0.

SetBootPart 分割區 (硬碟機名)

設定可開機旗標到所給定的分割區. 可開機旗標可以使主開機記錄 (MBR) 決定哪一個分割區可以開機. 第一個分割區編號是 1 .

Blank 分割區 (硬碟機名)

在分割區內資料以零填滿. 這對大的分割區會花掉不少時間. 別把這個分割區格式化爲任一種作業系統使用. 亦可參考 `Clean`.

Clean 分割區 (硬碟機名) ("標籤")

以分割表內所指定的格式快速格式化所給定的分割區. 如果有給定標籤且檔案系統有支援, 則會去設定分割區的標籤. 如要更徹底的格式化, 那就叫用 `Blank` 指令來處理分割區.

Clean 現在支援 (FAT16) BIGDOS, FAT32, EXT, LINUX-EXT2 及 LINUX-SWAP 格式的分割區. 如果要清理主開機記錄 (MBR), 使用 `Clean 0`.

Clean 可以用在資料分割區及 MBR/EXT 分割區. 而如果你在使用 `FullUnzip` 指令之前想要先用 clean 指令來清理資料則是無意義的.

FullUnzip "完整的磁碟映像檔" 分割區 (硬碟機名)

解壓縮一個完整的磁碟映像檔到所給定的分割區, 覆寫任何已存在的檔案 (快速清除).

FullUnzip 支援 (FAT16) BIGDOS, FAT32 及 LINUX-EXT2.

這個指令會啓用 VESA 繪圖功能來顯示程式正進行中的旗幟, 除非 `VESA` 被設爲 `OFF`.

IncrUnzip "增量型的磁碟映像檔" "目的路徑"

解壓縮一個增量型的磁碟映像檔到所給定的目的路徑. 這會取代掉目的路徑中與映像檔中所含檔名相同的檔案, 而其他的檔案則會不會被刪除.

IncrUnzip 支援 (FAT16) BIGDOS, FAT32 及 LINUX-EXT2 檔案格式. 這個指令的效率遠低於 FullUnzip 指令, 因爲它必須能保護原來的檔案系統不受損, 且它可藉由儲存多個映像檔的差異處, 以免去保存多個磁碟映像檔.

FileUnzip "來源檔名" "目的檔名"

解壓縮先前由 `MrZip` 的 FileZip 指令所壓縮的檔案. 這個檔案會經 32-位元 CRC 檢查.

Copy "來源檔名" "目的檔名"

以位元組對位元組的方式, 複製來源檔案到目的檔案. 可以在 FullUnzip 之後使用以更新在伺服器上的設定檔而不需重新建立映像檔. `FileUnzip` 指令, 最好使用在大而易於壓縮的檔案下.

Append "來源檔名**-1**" "來源檔名**-2**" "目的檔名"

以位元組對位元組的方式, 複製第一個來源檔案, 接著是第二個 到 目的檔案. 可用於任意大小的檔案. 目的檔名不可含有任一個來源檔名.

Patch "來源檔名" "目的檔名" ("先期修正" ("後期修正"))

讀取來源檔並以在寫入目的檔前, 針對變數作取代. 對於 ASCII 碼大於 127 的解釋由 `$IsoLatin` 變數來決定.

預設, 可以辯識到以 "${" 及 "}" 所括起來的變數. 且可以把他變換成非空的字串. 記住, 如果你要在修正內容中擺入錢字符則必須用溢位字元或者讓它可以在巨集中被辨識到. 例如, 如果你要明確的使用預設的先期與後期修正, 使用:

```
Patch "來源檔名" "目的檔名" "\${" "}"
```

MkDir "路徑"

遞迴地產生由根目錄啓始到所指路徑的目錄. 如果路徑已存在, 則指令無效.

Delete "檔名", Del "檔名"

移除所給定的檔名. 檔案必須存在.

DelTree "路徑"

遞迴地移除由根目錄到所指路徑的所有檔案與目錄, 並把所給定目錄移除.

5.1.12 開機指令

本節將列出用來延伸開機流程的指令. 額外的參數將也列入.

HideBootProm

回存 BootPROM 所對應的記憶體與中斷向量. 在叫用這個指令之後所有想要使用 TFTP 傳輸的功能會失效. 在 HdBoot 指令叫用之前使用這個功能是個好主意, 否則你在 DOS/Windows 下會導致記憶體不夠用. FloppyBoot 指令會直接叫用這個功能.

記住雖然這個功能會回存所有 BootPROM 會使用到的中斷向量 (正式公布的), 但目前看來似乎並沒把所有的都回存. 不過這對 DOS 及 Windows 已經足夠了.

LoadRamDisk "ramdisk的檔名"

載入一個軟碟的磁碟映像檔到延伸記憶體並重導 BIOS 的磁碟服務, 使得與軟碟有關的呼叫都用這個映像檔取代. 這個指令會直接叫用 `HideBootProm` 指令. 叫用 `FloppyBoot` 指令以用你剛剛所載入的 ramdisk 開機.

這種形式的 ramdisk 可能不若你用 TFTPBoot 指令來的穩定. 唯一的優點是可以由 TCP/IP BootPROM 所預留的 >64KB 中, 多偷出幾百位元組的傳統記憶體. 但也要警告你, 在 ramdisk 所存放的延伸記憶體區並不具任何安全性. 也沒有任何方法可以移除這種 ramdisk.

LoadZRamDisk "ramdisk 的檔名"

跟 LoadRamDisk 作相同的事, 但是是針對已用 `MrZip` 檔案壓縮過的映像檔. 被壓縮過的 ramdisk 採用 32 位元的 CRC 檢查以防止資料錯誤 (及 不完整下載).

TFTPBoot "遠端開機檔名"

連結到另一個可開機檔 (比如 一個由 Bootix 的 BpShell 程式所產生的軟碟映像檔). 參考對存取 TFTP 伺服器上檔案的參考轉換的方式.

FloppyBoot

隱藏 Boot ROM, 載入軟碟機的開機磁區並用它來開機.

HdBoot (磁碟機名)(:分割區)

載入所給定的可開機磁區並用它開機. 預設磁碟機爲 零, 第一顆硬碟機, 而分割區預設爲 零, 也就是主開機記錄. 你可以從任一個分割區來開機, 但必須注意 Windows 95/98 不能讓你在沒設定爲可開機分割區的分割區開機 (題示: 使用 SetBootPart).

這個指令不會直接叫用 HideBootProm, 所以你要先叫用它.

LinuxBoot "Kernel檔名" ("命令列" ("ramdisk 的檔名"))

載入所給定的 Kernel 及 ramdisk 到高層的記憶體, 設定命令列並用這個 Kernel 開機. 至少放個根檔案系統的位置在命令列中會是個好主意 (如 "root=dev/hda1"/). 如果你使用的系統相當依賴 `lilo` (如 RedHat Linux 5.1), 則在命令列中加入像 `BOOT_IMAGE=linux` 這樣的東西會比較好. 記住 Kernel 可以透過 TFTP 協定來載入 (自動從硬碟中快取) 或者直接由目的根分割區取用.

這個指令可以與小型的或大型的 Kernel (`zImage` 與 `bzImage`).

5.1.13 國際化語系支援

本節將列出與國際化語系有關的指令. 額外的參數將也列入.

RemapKeys "原始鍵" "重新對應的鍵"

國際化鍵盤支援. 重新對應所給定的鍵爲另一個字元. 比如, 把 Y 跟 Z 兩鍵交換, 使用:

```
Remapkeys "yzYZ" "zyZY"
```

在使用的字元不含在最小 ASCII 字元集中時, 使用八進制的表示法, 以避免必須依賴著 iso-latin 形式的設定.

對國際化的鍵盤而言, 在非對應過的模式 (US) 下, 有兩種鍵來產生反斜線符. 而它們都可以各自地重新對應, 這要感謝 `BpBatch` 會把他們分別回應成 ASCII 碼 252 (八進制) 或 ASCII 碼 335 (八進制) 當 shift 鍵按下時.

如果你要傳給我針對你們國家的鍵盤對應腳本檔, 我會把它擺在 *http://cuiwww.unige.ch/info/pc/remote-boot/soft/sample-scripts* 爲了幫助你設定你的鍵盤對應, 我建議你按下所有未對應到的 特殊功能 鍵並記下他們所產生的字元.

把它們擺在 原始鍵 這一欄. 而 重新對應的鍵 就是你想要它讓你看到的樣子. 如果部份按鍵 (不管是原始的或重新對應的) 產生超過 ASCII 十進位的 127, 則使用 八進制表示法. 你可以由 HelpPC (HelpPC 是一個由 David Jurgens 所寫的, 超文字線上說明共享程式) 的 ASCII 表中, 取得所給定任意字元的八進位碼.

RemapAltkeys "原始鍵" "重新對應的鍵"

國際化鍵盤支援. 重新對應所給定的鍵在 ALT 鍵按下時, 爲另一個字元. 例如, 對應 Alt-2 爲小老鼠符, 使用:

```
RemapAltKeys "2" "@"
```

記住重新開機的鍵並不支援.

LoadCodePage "cpxxx.bin"

載入並啓用所給定的二進制 Codepage 檔. Codepage 是用來轉換 Unicode 字元 (在 VFAT 的磁碟標籤中所實作的) 爲 8 位元的字元. 如果你沒有載入正確的 Codepage, 當用到特殊字元來存取檔案系統時, 會出現 FAT 警告.

所有的二進制 codepage 檔可以在這找到 *http://cuiwww.unige.ch/info/pc/remote-boot/soft/codepage.zip*

預設的 codepage 是 850, 是有記載的字集 ISO-Latin-1. 如果你要載入一個外國的 codepage, 你就要把變數 `$IsoLatin` 設爲 `"off"` 否則你會發生無意義的轉換. 此外, 如果你要顯示外國的字元, 你必需載入適當的螢幕字型 (使用 `"LoadFont"`).

LoadFont "字型檔名"

載入並啓用一種 VGA/VESA 字型, 可以在文字及圖形模式下使用. 字型檔必須是二進位檔 16 位元組/字元 (8x16 點陣). 這個指令可以在國際化語系中使用以得到更夢幻的支援.

有多個夢幻字型的檔案的壓縮檔在這 *http://cuiwww.unige.ch/info/pc/remote-boot/soft/fonts.zip*. 這個檔案還包含用來匯出你所用的 codepage 下的 DOS .CPI 檔中的字型的程式.

5.1.14 MrZip 指令的規格

Source...

```
Source (i)archive "檔名"
Source path "路徑"
```

設定要處理的來源檔案 (或增量形式的) 檔名與路徑.

Dest...

```
Dest (i)archive "檔名"
Dest (i)dump
Dest path "路徑"
```

設定要處理的目的檔案 (或增量形式的) 檔名, 路徑, 或 頃印. 如要控制頃印時所要顯示的資料量, 可以用 $DumpFormat 特殊變數來控制.

FileZip "來源檔名" "目的檔名"

壓縮一個先前用 FileUnzip 或 ZRamDisk 所解壓縮的檔案. 這個檔案會以 32 位元的 CRC 方式做檢查.

Filter...

```
Filter -"樣式"
Filter +"樣式"
```

避免/允許 合於所給定樣式 (允許萬用字元) 的檔案及目錄被包含進映像檔中. 樣式吻合於完整的路境名. 預設, 所有的檔案都會包含進映像檔中. 你只須要明確表示出哪些檔案必須由過濾器過濾. 每個負的過濾器都有相對應的正過濾器 (被允許的).

在 DOS/Windows 的映像檔, 你只要用

```
Filter -"*.swp"
Filter -"temp/*"
```

而對 Unix 的映像檔, 你則需要

```
Filter -"var/log/*"
Filter -"tmp/*"
```

CopyArchive

根據所提供的來源, 目的檔案, 及過濾器等設定開始處理映像檔. 只有在極少數的情況, 不然你通常都可以直接使用以下所列出的快捷用法以取代叫用 `CopyArchive` 指令. 其中一種情況你會用到 `CopyArchive` 就是當你要變更分割的映像檔大小時, 如下:

```
set FragmentSize="30 MB"
Source archive "original.imz"
Dest archive "refragmented.imz"
CopyArchive
```

FullZip "路徑" "完整的映像檔名"

是下述的快捷用法

```
Source path "路徑"
Dest archive "完整的映像檔名"
CopyArchive
```

你通常都要先設定過濾器.

IncrZip "路徑" "增量型映像檔名"

是下述的快捷用法

```
Source path "路徑"
Dest iarchive "增量型映像檔名"
CopyArchive
```

FullDump "完整的映像檔名"

是下述的快捷用法

```
Source archive "完整的映像檔名"
Dest dump
CopyArchive
```

IncrDump "增量型映像檔名"

是下述的快捷用法

```
Source iarchive "增量型映像檔名"
Dest dump
CopyArchive
```

Xcopy "來源路徑" "目的路徑"

是下述的快捷用法

```
Source path "來源路徑"
Dest path "目的路徑"
CopyArchive
```

5.2 NoBreak.sys

Nobreak.sys 相當小 (大概只有 350 位元組) 的驅動程式, 它將加在您 config.sys 中的開始處. 它的目的是保護開機程序不被中斷, 直到使用者登入. DOS 爲這提供一種設定 (名爲 BREAK=OFF), 但它不夠暴力, 並且在執行到 autoexec.bat 時一點用處也沒有. 我們的驅動程式工作原理是當中斷要求產生時修改鍵盤的掃描碼, 直接在 BIOS 層工作. 以這方法, 所有的程式在中斷未致能前都收不到中斷訊息.

這個驅動程式必須放在 config.sys 檔中(或者使用 *Undocumented DOS* 的 devlod 程式). 接著, 可以透過傳送 Yes 到 NOBRK 這個虛擬裝置, 並且傳送 No 而禁能 (事實上, 只需要第一個字元, Y 或 N 才是關鍵).

因爲這個驅動程式依靠 BIOS, 所以它只能在 DOS 跟 Windows 3.1 下工作, 而 Windows 95/98 則有它自己的低階鍵盤處理機制.

Assembler 語法的原始程式碼在這.

6 特殊型的 TFTP 伺服器

由於 TCP/IP BootPROM 只能支援 TFTP 通訊協定, 而這與增強型的 TFTP 伺服器新功能息息相關.

6.1 Bootix 的增強型 TFTP 伺服器

Bootix 隨著 TCP/IP BootPROM 一起發行的增強型 TFTP 伺服器, 可以傳送每個封包達 1408 位元組的資料以取代原標準的 512 位元組. 這個是個很棒的功能你應該多加利用. 這個伺服器程式可以在 TCP/IP BootPROM 工具程式磁片中找到, 有 Solaris, Windows 及 Netware NLM 版本.

6.2 Linux 的增強型 TFTP 伺服器

在 Linux 下, 可以用 Bootix/Incom 所寫的 TFTP 伺服器. 供 Linux 使用的版本已編譯好,. 放在這 *http://cuiwww.unige.ch/info/pc/remote-boot/soft/incomtftpdlx.tar.gz*

6.3 Solaris 的 TFTP 伺服器

原始的 Solaris TFTP 伺服器程式有嚴格的檔案大小限制, 因爲它是使用 16-位元的有號整數來計算封包大小. Bootix 所提供的給 Solaris 的增強型 TFTP 伺服器, 就可以解決這個問題, 以提供更快的傳輸速度.

不過, 不份使用者回應 (特別是搭配 SMC9432 卡使用時) 會在啓動它時產生 lock-ups 的問題. 所以, John Martin (j.martin@abdn.ac.uk) 已把 Linux 的 tftpd 移殖到 Solaris. 你可以在這找到 *http://www.csd.abdn.ac.uk/~jmartin/tftpd.tar.gz*.

6.4 安全閘道器

我們寫了一個特殊的 TFTP 伺服器用來當成認證使用者的安全閘道器. 這個伺服器可以在 Linux 或 Solaris 下執行, 並可以透過 Unix 的 通行密碼資料庫 (NIS 及 shadow 通行密碼都已能支援) 或 Windows NT (或 Samba) 伺服器 或 Radius 伺服器. 它放在這 *http://cuiwww.unige.ch/info/pc/remote-boot/soft/stdtpd.tar.gz*, 有原始程式及預先編譯好的可執行檔. 預先編譯好的可執行檔不包含 NT 通行密碼的加密功能因爲我們不能散佈 libdes 即使有滿多人抱怨也一樣.

爲了要使用安全閘道器, 你只要對 *security domains* 設定檔作一些設定, 內容是指每個邏輯安全性領域是對應到哪一台認證伺服器 (Unix 領域直接對應到 Unix 通行密碼資料庫伺服器). 以下是一個設定範例檔:

```
#
# STFTPD 設定檔
#
# 這個檔案用來敘述伺服器的 "security domains".
# 已支援兩種認證伺服器 : Radius 或 winnt (winnt 包含 NT 伺服器 和 Samba)
#
# Radius 伺服器的設定格式
# radius      <領域名>      <伺服器網路位址>      <secret>
#
# secret 是指在你的 /etc/raddb/clients 檔裡頭的密字組
#
# SMB 伺服器的設定格式
# winnt       <領域名>      <伺服器網路位址>      <netbiosname>
#
# netbiosname 是指伺服器的 NETBIOS 識別名稱
#
# 範例
radius        sec-dom-rad     radiusserver    testing123
winnt         sec-dom-nt1     192.168.1.1     NTserver1
winnt         sec-dom-smb     samba           SAMBA1
```

記住 如果你使用了 Samba, 你必須設定 `security = user`.

你也可以提供一個哪些使用者禁止登入 (會使他們的無法通過檢查) 的列表給安全認證伺服器.

6.5 廣播型 TFTP 伺服器

我們撰寫了一個特殊型 TFTP 伺服器以實現自訂的 TFTP 廣播功能. 用這種伺服器, 我們可以在高負載的 10 Mb 乙太網路環境下, 以 6 Mb/s 的傳輸率下載映像檔到 25 台用户端電腦 (這比傳統的 TFTP 傳輸協定來得有效率多了, 因爲它完全不須要對每個封包都做一次已收到的回應). 這種伺服器程式可以在 Linux 或 Solaris 環境下執行. 可以在這取用 *http://cuiwww.unige.ch/info/pc/remote-boot/soft/btdtpd.tar.gz*, 含原始程式碼及以編譯好的可執行檔.

因爲 TCP/IP BootPROM 本身不支援這種通訊協定, 我們的解決方法是以一個極小的 ramdisk 爲基礎的 Linux 系統開機, 然後再搭配本文所談到的工具程式: Linux 版本的 MrBatch, 它本身就內建支援廣播型 TFTP 通訊協定了. 一個簡易的批次檔可以同時使所有的用户端電腦在幾分鐘內下載完所有的檔案. 要使用這個功能你並不需要自行安裝 Linux, 除非你有相當奇特的硬體配備, 以致無法直接使用這個套件所提供的 Kernel.

程序如下. 首先你要手動啓動廣播型伺服器, 並給定參數以設定預期要開機的用户端電腦數量 (記住, 這個程序不需要天天做. 只有在你變更了映像檔內容後, 想確定它可以直接載入到你所有的用户端電腦時才要用到). 接著, 你把所有的用户端電腦都打開, 他們就會執行下列的 BpBatch 腳本程序:

```
#
# 這個批次檔是由 BpBatch 所執行, 用以啓動一個 內建 ramdisk 的 mini-linux,
# 然後在此 Linux 環境下執行 MrBatch.
#
# 廣播型 TFTP 通訊協定只能在 Linux 版本的 MrBatch 作動, 因爲 BootPROM 本身不支援.
```

```
#
# 1. 設定一個極小的分割區, 用來當快取.
setpartitions "BIGDOS:50"
# 2. 清理 MBR.
clean 0
# 3. 執行一個具 initrd (Initial Ramdisk) 支援的 Linux Kernel, 並用
#    bcastrd.gz 爲 初始 ramdisk (將會被掛在 root 並透過 /linuxrc 執行).
#    參考 initrd.txt 以獲得更多關於 初始 ramdisk 的詳細資料.
#    你不需要去設定 root 裝置 (第二個參數是無用的) 到 Kernel,
#    它會自行使用 初始 ramdisk.
linuxboot "linux.krn" "" "bcastrd.gz"
# 4. 初始 ramdisk 會執行 dhcpd 以透過 DHCP 通訊協定來設定網路.
#    接著它會執行 mrbatch -w bcastlx
```

初始 ramdisk 內含:

- `dhcpcd`, 一個 DHCP 用户端程式用來設定網路.
- `mrbatch`
- `linuxrc`, 一個會被 initrd 啓動的檔案, 用來延續啓動 `dhcpd` 及之後要用的 `MrBatch`.
- `usr/lib/terminfo/l/linux`, 由 MrBatch 所使用.
- `dev/*`, 執行 Linux 及 MrBatch 所需的裝置.

所有的程式都去蕪存菁並編譯成靜態連結, 以避免使用到 `libc.so` 因爲它眞的很大. 最後以 Gzip 壓縮的 ramdisk 大小低於 300 KB. 而 Kernel 本身用掉 450KB (並支援多數的網路卡及 initrd). 當 Linux 一啓動執行, MrBatch 會被叫用並依下列的腳本程序來工作 (你也可以根據你所需來修改):

```
# 這個檔案是在 MrBatch 於 初始 ramdisk bcastrd.gz 啓動後執行
# 他主要的功用是 "採廣播的方式複製" 檔案到快取裡頭
#
# 1. 看看詳細資訊.
showlog
# 2. 不需要 "press a key".
set pauselog="OFF"
# 3. 設定分割區爲他們最終值.
#    重點: 當你要複製檔案到快取區以供以後的開機使用時,
#    你需要在以後所用到開機設定都採用一樣的分割區.
setpartitions "BIGDOS:1024"
# 4. 清理 快取分割區.
clean -1
# 5. 透過 廣播型 TFTP 通訊協定 (埠號 99) , 複製檔案到快取區.
#
# 你可以照我們所寫的腳本程序來用, 但你百分之百確定需要修改下面這一行 !
# 在我們的範例下, 我們下載的檔案是 mblinux.imz, 它是我們安裝的 Linux 的映像檔.
copy "$BOOTP-Server-IP@99:mblinux.imz" "{:-1}mblinux.imz"
```

如果傳輸完成, 你只須要關掉所有用户端電腦並把它們的初始開機程序腳本設定成你所喜號的選單.