

FreeBSD 概論

dennylin93

<dennylin93@cnmc32.hs.ntnu.edu.tw>

CNMC, HSNU

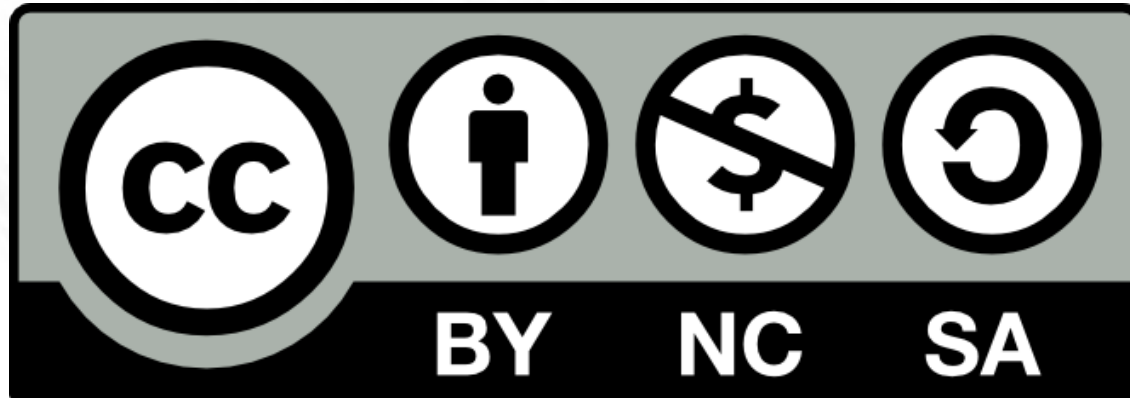
2009/10/30

授權

本文件採用：

Creative Commons

Attribution-Noncommercial-Share Alike 3.0 Unported



爲什麼要用 FreeBSD ？

- 它**很強**！
- 它**非常強**！
- 它**霹靂無敵強**！

屬於 **Unix** 的分枝（**Unix** 有 40 年了）。

講白一點，就是它的穩定度、效能皆非常出色，而且說明文件齊全，易於使用。

我用過不少作業系統，而 **FreeBSD** 大概是我唯一真的喜歡的作業系統。

它可以做什麼？

其實很多人都對 **FreeBSD** 不是很了解，以為它只能當伺服器。大錯特錯！我個人拿它來當 **desktop**，非常好像。看一下 [screenshots](#)。
。總之，它也可以做：

- 伺服器
- 防火牆
- 個人電腦
- etc.

硬體支援

從古老的 Pentium Pro 到最新的 Xeon 處理器、Fast Ethernet 到 Gigabit Ethernet 的 NIC、各式 RAID controllers FreeBSD 都支援。不論是高級伺服器、桌上型電腦或 10 年前的舊古董，幾乎都可以跑 FreeBSD。

FreeBSD 的硬體支援並不會輸給其他作業系統（通常指伺服器方面，但近幾年 FreeBSD 對 desktop 的支援愈來愈好）。

CPU Architectures

- 目前 Tier 1 的 archs : i386 、 amd64 、 pc98
- 其他有支援的包含：
arm 、 powerpc 、 ia64 、 sparc64

目前比較常見的其實都是 i386 、 amd64 ，
而現在市面上的 CPU 幾乎都是 amd64
了。

版本釋出過程 I

每個作業系統都有自己的釋出過程，而 FreeBSD 也不例外。

- **-CURRENT**：所有最新的開發，較不穩定
- **-STABLE**：比較保守一點，大部分都經過 HEAD 才 commit 進來
- **-RELEASE**：這就是穩定的版本！

有時候會看到

HEAD、**RELENG_7**、**RELENG_7_2** 之類的名稱，其實跟上面一樣。

版本釋出過程 II

- 都是先在 `-CURRENT` 開發
- Branch 出 `-STABLE`，進入 code slush、freeze
- 經過 BETA、RC，`-RELEASE` 就但生了
- `-STABLE` 再度進入開發
- `-RELEASE` 釋出會繼續有更新，如 `RELENG_7_2`
- 開發時，怎麼 port 回去：MFC

FreeBSD 內部運作

- 核心、周圍程式都由 FreeBSD 開發團隊維護
- 軟體安裝、管理幾乎都透過 Ports
- Comitters 分爲 doc、src、ports
- 成爲 comitter 的方式：一直送 PR
- 程式碼管理：之前是 CVS，現在可能會慢慢變成 Subversion

檔案系統

檔案系統即是硬碟分割時的格式，隨作業系統而異。常見的如：**FAT32**、**UFS**、**ext3**。

FreeBSD 支援：

- **UFS**（預設，穩定，參見 **Soft Updates**）
- **ZFS**（新，功能非常強，較耗資源）
- **NTFS**、**FAT32**
- **ext3**、**ext2**
- 更多...

虛擬化

隨著資訊的發展，虛擬化已經普遍使用，未來會更受重視。FreeBSD 目前已有 VirtualBox、QEMU，Xen 還在發展中。

若 FreeBSD 都是 host、guest 的話，可以使用 FreeBSD jails。不但可以透過 nullfs 節省空間，效能也絕佳（與 native 相同）。

有誰在用 FreeBSD ?

- Yahoo!
- Cisco
- Juniper
- NetApp
- Apache
- Netcraft
- Sony
- 還有很多

終端機

沒錯，灌好之後看到的的就是終端機！其實也沒什麼好害怕的。用久了就會愛上它。

- **Console** 共有 8 個（預設）
- 用 **Alt + [0-7]** 互換
- 其實直接用 **console** 的機會很少

root

開機後，請不會傻在哪邊：用 **root** 登入！若安裝後沒有設密碼，密碼為空白。

- **root** 是超級使用者，它可以做任何事
- 平常沒事不要用 **root**，有需要的時候用 **sudo** 或 **su** 取得權限

原因？你可能不小心手賤下錯指令。

系統架構 I

FreeBSD 系統架構與 Windows 有很大的差別。

- / 根目錄
- /boot/ 開機有關的檔案，含 kernel
- /dev/ 裝置
- /etc/ 系統設定檔
- /tmp/ 暫存用，很少用到
- /var/ 資料庫、紀錄檔、程序

系統架構 II

- `/bin/` 系統程式
- `/sbin/` 系統程式
- `/usr/` 使用者有關的東西
- `/usr/bin/` 使用者程式
- `/usr/sbin/` 使用者程式
- `/usr/src/` 原始碼
- `/usr/ports/` Ports

系統架構 III

- `/usr/share/` 系統範例檔案
- `/usr/local/` 與 Ports 息息相關
- `/usr/local/share/` Ports 的範例檔案
- `/usr/local/etc/` Ports 設定檔

權限

FreeBSD 權限與 Windows 完全不同，可以有很好的控制。

- `-rwxr-xr-x` `dennylin93` `dennylin93`
- 這裡看到有 `r` (4)、`w` (2)、`x` (1) 三種不同的權限。`r` 代表可以讀取，`w` 可以寫入，`x` 可以執行 (`$ man chmod`)。
- 可以看到有三個區塊 (黑、藍、紅)，分別代表使用者、群組、與其它的權限。
- 後面兩個分別是使用使用者、群組。

基本指令 |

其實，名字只要有「基本」兩個字的東西，一點都不基本！不信的話，下次數學課，問老師：「可以證代數基本定理給我看嗎？」

千萬不要小看這些指令的威力...這些都是必學，且非常有用

順便題一下，**man** 會是你很好的朋友！

基本指令 II

- `man` 還用說嗎！？
- `ls` 看目錄底下的檔案
- `cd` 想換目錄嗎 XD ？
- `pwd` 腦容量不足，忘了自己在哪裡？
- `cp` 別搞錯！這是複製檔案...
- `rm` 刪除檔案，刪資料夾參數 `-r`
- `ee` 編輯檔案
- `vi` 超強檔案文字編輯器！！！！

基本指令 III

- `mkdir` 資料夾不敷使用，怎麼辦？
- `rmdir` 其實可以被 `$ rm -r` 取代
- `touch` 新增檔案
- `cat` 妙用無窮！
- `echo` 簡單來說，就是輸出
- `more` Pager
- `less` Pager
- `(head|tail)` 看檔案開頭、結尾

基本指令 IV

- `clear` 清理螢幕
- `chown` 改變使用者、群組
- `chmod` 改變權限
- `chflags` 更進階的權限
- `ln` 做連結啊
- `exit` 登出
- `df` 看硬碟使用量
- `du` 資料夾大小

基本指令 V

- `su` 想要 `root` 權限嗎？
- `adduser` 新增使用者
- `rmuser` 刪除使用者
- `shutdown` 關機（廢話）
- `reboot` 重新開機（廢話）
- `uptime` 開機多久了？一秒？一年？
- `login` 最近誰登入過，或重開、關機
- `lastlogin` 看使用者最後登入時間

基本指令 VI

- `id` 我是誰？
- `who` 有誰登入？
- `w` 有誰登入？
- `top` 觀看程序，動態
- `ps` 差不多，靜態
- `kill` 控制程序（砍掉、更新等）

基本指令 VII

- `grep` 字串搜尋
- `find` 尋找檔案、資料夾
- `locate` 快速尋找檔案、資料夾
- `whereis` 在哪（限定）？
- `mount` 載入
- `umount` 卸載
- `fsck` 檢查磁碟損毀

基本指令 VIII

- ping 就 ping 啊...
- ifconfig 網路設定
- dig 查詢 DNS
- host 查詢 DNS
- netstat 網路資訊
- sockstat 網路資訊
- arp 查詢 ARP 紀錄

基本指令 IX

這裡先做個小小的歸納。一開始接觸這些指令或許很陌生，但看習慣之後就會發覺他們真的很好用。比起在圖形話介面上除錯、設定容易多了。

不必硬背，用了幾次之後自然而然會記住。

"Unix is simple. It just takes a genius to understand its simplicity."

—Dennis Ritchie

重導 IO

剛剛指令中有 `cat`、`echo` 一類的指令。這些在 `scripts` 當中重導 IO 功能十分強大。

一些例子：

- `$ echo "Hello world!" > foo`
- `$ cat foo > bar`

管道

管道英文稱爲「**pipes**」。這也是一個非常好用的東西。

例子：

- `$ pkg_info -qoa | sort -`
- `$ wc -l * | sort -`

或許一開始不懂它爲什麼好用，不必擔心，以後一定有機會用到。

另外，有些程式預設輸入是 `stdin`（像 `sort`）。

Shell

Shell 任何作業系統之中都扮演非常重要的角色，它等於提供使用者一個適合的工作環境。

目前比較常見的有：**sh**、**(t) csh**、**bash**、**zsh**、**ksh**。

Shell 重要功能之一就是自動完成，另外還有 **shell scripts**。

vi & ee

講到任何作業系統，都比需要文字編輯器。而在非 **Windows** 的作業系統上，更為重要。目前，**vi** 是任何 **OS** 必備的編輯器，另外 **ee** 對初學者比較有親和力。

說實話，**ee** 簡單，但是個人覺得 **vi** 還是藥學，畢竟它太重要了。此外，可以考慮使用 **Vim**，因為有更多選擇。

結論

有一點必須不斷的強調，**FreeBSD** 不是一個很難學的作業系統，只是看你是否用心。我敢說，有努力，一定會有回報，而且投資報酬率很高。

不要因為還怕接觸它的環境而退縮。或許，將來你會欣賞它的 **simplicity**。

FreeBSD 有的是「資源」，最好能利用這個優勢。

參考

- For People New to Both FreeBSD and UNIX
- Explaining BSD
- BSD for Linux Users
- FreeBSD
- FreeBSD 使用手冊（必看！）
- Man pages（千萬不要小看它！）
- NCTU 計算機系統管理
- FreeBSD Forums（有很多高手在上面）
- freenode ##freebsd、#bsdchat