NanoBSD 簡介

Daniel Gerzo

修訂: 43126

版權 © 2006 The FreeBSD Documentation Project

FreeBSD 是 FreeBSD基金會的註冊商標

許多製造商和經銷商使用一些稱為商標的圖案或文字設計來彰顯自己的產品。 本文中出現的衆多商標，以及 FreeBSD Project 本身廣所人知的商標，後面將以 '#' 或 '#' 符號來標註。

2013-11-07 由 gabor.

摘要

這篇文件提供了關於 NanoBSD 工具的情報介紹， 這工具可用來建立用於嵌入式環境應用程式的 FreeBSD 系統映像檔， 以便存放到 Compact Flash 卡(或隨身碟)。

內容目錄

1. NanoBSD 簡介 1
2. 如何使用 NanoBSD 2
索引 7

1. NanoBSD 簡介

NanoBSD 是 Poul-Henning Kamp 目前正在開發的一項工具。 它可用來建立用於嵌入式環境應用程式的 FreeBSD 系統映像檔， 以便存放到 Compact Flash 卡(或隨身碟，mass storage medium)。

這一工具也可以用來自製安裝映像檔， 以簡化俗稱為 “嵌入式系統(computer appliances)” 的系統安裝、維護工作。 通常，每個嵌入式系統產品都有限定硬體和軟體， 或者換言之，所有的應用程式都是預先裝好的。 這些設備可以直接放到現有的網路中，而且(幾乎是)立即使用。

NanoBSD 提供的功能包括：

- 可以和 FreeBSD 一樣使用 Ports 和 Packages — 所有的應用程序都可以在 NanoBSD 中直接使用， 而方式與 FreeBSD 完全一樣。

- 功能絲毫未損 — 在 FreeBSD 做的任何工作，都可以在 NanoBSD 中使用， 除非您在建立 NanoBSD 映像檔時， 有指定要拿掉它們。
- 所有東西在運行時都是唯讀的 — 可以安全地拔掉電源插頭。 系統不正常關機的話，不用再跑 fsck(8) 了。
- 可輕鬆編譯、自行打造 — 只需使用一個 shell script 和一個設定檔， 您可以輕鬆依需求來量身訂做適用的映像檔。

2. 如何使用 NanoBSD

2.1. NanoBSD 的設計

一旦將映像檔存入嵌入式硬體，就可以用它來引導 NanoBSD 了。 預設情況下，隨身碟會劃分為三部分：

- 兩個映像檔分割區： code#1 和 code#2。
- 一個設定檔分割區，在運行環境中， 可以將其掛載(mount)到 /cfg 目錄下。

這些分割區，在預設情況下是以唯讀方式掛載。

/etc 和 /var 目錄均為 md(4)(malloc)磁碟。

設定檔的分割區則是在 /cfg 目錄。 它包含了用於 /etc 目錄的檔案，在啓動之後暫時以唯讀方式掛載。 因此，若想要重開機保留新的設定， 那麼要記得從 /etc 把改過的檔案複製回 /cfg 目錄才行。

範例 1. 把修改過 /etc/resolv.conf 設定保存起來

```
# vi -/etc/resolv.conf
[...]
# mount -/cfg
# cp -/etc/resolv.conf -/cfg
# umount -/cfg
```

注意

只有在系統啓動過程中，以及需要修改設定檔的時候，才需要掛載含有 /cfg 的那個分割區。

> 一直都掛載 /cfg 不是一個好主意，特別是當您把 NanoBSD 放在不適合進行大量寫入動作的分割區時 （比如：由於檔案系統的同步化會定期在系統碟內寫入資料）。

2.2. 打造 NanoBSD 映像檔

NanoBSD 映像檔是透過使用非常簡單的 nanobsd.sh shell script 來打造的，這個 script 可以在 /usr/src/tools/tools/nanobsd 目錄中找到。 這個 script 建立的映像檔，可以用 dd(1) 工具來複製到隨身碟上。

打造 NanoBSD 映像檔所需的指令是：

```
# cd -/usr/src/tools/tools/nanobsd ❶
# sh nanobsd.sh ❷
# cd -/usr/obj/nanobsd.full ❸
# dd if=_.disk.full of=/dev/da0 bs=64k ❹
```

❶ 進入 NanoBSD 打造 script 的主目錄。
❷ 開始打造過程。
❸ 進入打造好的映像檔所在的目錄。
❹ 在隨身碟上安裝 NanoBSD。

2.3. 自行打造 NanoBSD 映像檔

這可能是 NanoBSD 最為重要， 同時也是您最感興趣的功能。 同時，在開發 NanoBSD 應用程式時，這也是相當耗時的過程。

執行下面的指令將會 nanobsd.sh 讀取目前所在目錄的 myconf.nano 檔的設定：

```
# sh nanobsd.sh --c myconf.nano
```

自行打造的流程，只需兩個步驟：

・自訂選項

・自訂功能

2.3.1. 自訂選項

透過修改設定，可以設定用於 NanoBSD 打造過程中 buildworld 和 installworld 階段的編譯、安裝選項，以及 NanoBSD 主要打造過程中的選項。 透過使用這些選項可以削減系統的尺寸，使之能夠放入 64 MB 的隨身碟。 您還可以進一步透過這些選項來削減 FreeBSD， 直到它只包含 kernel 以及兩三個 userland 檔案為止。

設定檔案中包含用以代替預設值的設定選項。簡介最重要的幾項設定如下：

・NANO_NAME — 本次打造的名稱(所建立工作目錄的名稱)。

・NANO_SRC — 用以編譯、打造映像檔的 source tree 的位置。

・NANO_KERNEL — 設定用來編譯的 kernel 設定檔檔名。

・CONF_BUILD — 用於 buildworld 打造階段的選項。

・CONF_INSTALL — 用於 installworld 打造階段的選項。

・CONF_WORLD — 用於 buildworld 和 installworld 這兩個打造階段的選項。

・FlashDevice — 定義所用的嵌入式硬體類型。 詳情請參考 FlashDevice.sub 檔。

2.3.2. 自訂功能

透過在設定檔案中使用 shell 函數，可以進一步微調 NanoBSD。 舉例說明一下自行打造函數的基本方式：

```
cust_foo()(
 echo -"bar=topless" > \
  ${NANO_WORLDDIR}/etc/foo
)
customize_cmd cust_foo
```

下面舉更實際點的例子，它會把預設的 /etc 目錄大小，從 5MB 調整為 30MB：

```
cust_etc_size()(
 cd ${NANO_WORLDDIR}/conf
 echo 30000 > default/etc/md_size
)
customize_cmd cust_etc_size
```

除此之外，還有幾個預設的功能定義可以用來自訂：

・cust_comconsole — 在預設 VGA 顯示卡上停用 getty(8) (/dev/ttyv*)並啓用 serial port 的 COM1 以作為系統 console。

・cust_allow_ssh_root — 允許 sshd(8) 可以用 root 帳號登入。

・cust_install_files — 從 nanobsd/Files 目錄中安裝檔案，這包含一些實用的系統管理 script 。

2.3.3. 設定檔案舉例

下面是用於自行打造的 NanoBSD 映像檔的完整例子：

```
NANO_NAME=custom
NANO_SRC=/usr/src
NANO_KERNEL=MYKERNEL
NANO_IMAGES=2
```

```
CONF_BUILD='
NO_KLDLOAD=YES
NO_NETGRAPH=YES
NO_PAM=YES
'

CONF_INSTALL='
NO_ACPI=YES
NO_BLUETOOTH=YES
NO_CVS=YES
NO_FORTRAN=YES
NO_HTML=YES
NO_LPR=YES
NO_MAN=YES
NO_SENDMAIL=YES
NO_SHAREDOCS=YES
NO_EXAMPLES=YES
NO_INSTALLLIB=YES
NO_CALENDAR=YES
NO_MISC=YES
NO_SHARE=YES
'

CONF_WORLD='
NO_BIND=YES
NO_MODULES=YES
NO_KERBEROS=YES
NO_GAMES=YES
NO_RESCUE=YES
NO_LOCALES=YES
NO_SYSCONS=YES
NO_INFO=YES
'

FlashDevice SanDisk 1G

cust_nobeastie()(
 touch ${NANO_WORLDDIR}/boot/loader.conf
 echo -"beastie_disable=\"YES\"" >> ${NANO_WORLDDIR}/boot/loader.conf
)

customize_cmd cust_comconsole
customize_cmd cust_install_files
customize_cmd cust_allow_ssh_root
customize_cmd cust_nobeastie
```

2.4. 更新 NanoBSD

更新 NanoBSD 相對 FreeBSD 而言較為簡單：

1. 和之前一樣打造新的 NanoBSD 映像檔。
2. 將新的映像檔放入正運行的 NanoBSD 中未用的分割區之一。

與之前最初安裝 NanoBSD 的步驟相比， 這一步驟最重要的區別在於：這次不用 _.disk.full 檔(它包含整個磁碟的映像檔)， 而應安裝 _.disk.image 映像檔(這個檔案中，只包含一個系統分割區)。

3. 重新啓動，並從新安裝的分割區中啓動系統。

4. 如果一切順利的話，升級工作就完成了。

5. 如果發生了任何問題，則可以從先前的分割區啓動 (其中包含了舊的、 可用的映像檔)，來盡快恢復系統功能。 接下來可以修正新編譯的版本中存在的問題，並重複前述步驟。

要在正在運行的 NanoBSD 系統中安裝新的映像檔，可以使用位於 /root 目錄的 updatep1 或 updatep2 script ， 實際上要用哪一個 script，則取決於正在運行的系統是位於哪個分割區而定。

隨時提供新 NanoBSD 映像檔所提供的服務， 以及採用的傳輸方法的不同，您可以參考並使用下列三種方式之一：

2.4.1. 使用 ftp(1)

如果傳輸速度是第一要求的話，請採用下面例子：

```
# ftp myhost
get _.disk.image -"| sh updatep1"
```

2.4.2. 使用 ssh(1)

如果想更安全的話，應參考下面例子：

```
# ssh myhost cat _.disk.image.gz -| zcat -| sh updatep1
```

2.4.3. 使用 nc(1)

如果遠程主機既不提供 ftp(1) 服務，也不提供 sshd(8) 服務的話：

1. 首先，在提供映像檔的主機上開啓 TCP listen，並讓它把映像檔傳給 client：

```
myhost# nc -l 2222 < _.disk.image
```

注意

請確認您所使用的 port 沒有被防火牆阻止來自 NanoBSD client 的連線請求。

2. 連到提供新映像檔服務的主機，並執行 updatep1 這支 script：

```
# nc myhost 2222 -| sh updatep1
```

索引

N

NanoBSD, 1