

Linux® 用者的 FreeBSD 快速入门向

摘要

本文旨在快速使那些高 Linux® 用者熟悉 FreeBSD 的一些基础知识。

目

1. 介	1
2. Shell程序：没有Bash？	1
3. Packages和Ports：在 FreeBSD 中添加件	1
4. 系：行在里？	3
5. 网配置	4
6. 防火	5
7. 升 FreeBSD	6
8. procfs：已是去式但仍未被忘	6
9. 常用命令	7
10. 附录	8

1. 介

本文将突出介绍 FreeBSD 与 Linux® 的差别，以使得那些 Linux® 高用者能自己快速熟悉 FreeBSD 的基础内容。这只是技术上的快速入门，并非是描述操作系统之间的"哲学"上的差别。

此文假定您已经安装好了 FreeBSD。如果您没有安装 FreeBSD 或者 FreeBSD 的安装过程方面需要帮助，请参考 FreeBSD 手册的 [安装 FreeBSD](#) 一章。

2. Shell程序：没有Bash？

那些从 Linux® 来的用者通常会习惯于 Bash 不是 FreeBSD 的默认 Shell。事实上，Bash 甚至没有包括在 FreeBSD 的默认安装中。代替的是，FreeBSD 使用 [tcsh\(1\)](#) 作为自己的默认 Shell，尽管如此，Bash 和其他喜欢的 Shell 程序在 FreeBSD 的 [Packages](#) 和 [Ports](#) 套件里都可以得到。

如果您安装了其他的 Shell 可以使用 [chsh\(1\)](#) 来设置一个用者的默认 Shell。通常情况下，强烈建议不要去更改 root 用者的默认 Shell。原因是有些 Shell 没有包括在基本系统中，正常情况下它会被安装到 /usr/local/bin 和 /usr/bin 目录下。万一某天 /usr/local/bin 和 /usr/bin 的文件系统不能被挂载，这种情况下 root 将不能进入自己默认的 Shell，从而 root 将不能登录去。由于这个原因，第二个系统管理工具 [toor](#) 建议使用的是非默认的 Shell。在安全 FAQ 可以了解到关于 [toor](#) 的信息。

3. Packages和Ports：在 FreeBSD 中添加件

除了经典的 UNIX® 安装件的方法（下载源包，解包，编译源，安装）外，FreeBSD 提供了另外方法来安装

用程序：packages 和 ports。可以在 [这里](#) 到一完整可用的 ports 和 packages 的件清。

3.1. Packages

Packages 是好的用程序，在 FreeBSD 中等价于基于 Debian/Ubuntu 的系中的 .deb 件包以及基于 Red Hat/Fedora 的系中的 .rpm 件包。Packages 使用 `pkg_add(1)` 来行安装。例如，下面的命令将用来安装 Apache 2.2：

```
# pkg_add /tmp/apache-2.2.6_2.tbz
```

使用 `-r` 操作将告 `pkg_add(1)` 来自取并安装一个件包，以及解决所有的依系：

```
# pkg_add -r apache22
Fetching ftp://ftp.freebsd.org/pub/FreeBSD/ports/i386/packages-6.2-
release/Latest/apache22.tbz... Done.
Fetching ftp://ftp.freebsd.org/pub/FreeBSD/ports/i386/packages-6.2-release/All/expat-
2.0.0_1.tbz... Done.
Fetching ftp://ftp.freebsd.org/pub/FreeBSD/ports/i386/packages-6.2-release/All/perl-
5.8.8_1.tbz... Done.
[snip]
```

```
To run apache www server from startup, add apache22_enable="YES"
in your /etc/rc.conf. Extra options can be found in startup script.
```



如果正行着 release 版本的 FreeBSD (6.2, 6.3, 7.0等，通常从 CD-ROM 被安装的) `pkg_add -r` 会其下些特定版本建好的件包。些件包可能不是当前最新的程序。可以使用 `PACKAGESITE` 量来覆默的作。例如，把 `PACKAGESITE` 置成 `ftp://ftp.freebsd.org/pub/FreeBSD/ports/i386/packages-6-stable/Latest/` 来下 6.X 系列最新的包。

想了解更多的 packages 信息 FreeBSD 手册的 4.4 小：[使用 Packages 系](#)。

3.2. Ports

FreeBSD 的第二安装用程序的方法就是使用 Ports 套件了。Ports 套件是 FreeBSD 上的一个利用 Makefile 和一些丁文件来特定从源定制安装各件程序的框架。当安装一个 port 系会取程序源，用任何所需要的丁，源，并安装用程序（并依系以同的方式安装解决）。

Ports 套件，常被称作 ports，可以在 `/usr/ports` 下到。假 Ports 套件已在安装 FreeBSD 安装了。如果 Ports 套件没有被安装可以通 `sysinstall(8)` 来行安装，或者使用 `csup(1)` 或 `portsnap(8)` 来从 FreeBSD 的服器上面拉下来。在手册的 4.5.1 小可以到安装 Ports 套件的介。

安装一个 port 就像入 port 的目并始建程一（通常情况下），下面是从 Ports 套件安装 Apache 2.2 的例子：

```
# cd /usr/ports/www/apache22
# make install clean
```

使用 ports 安装软件的最大好处就是能自定义安装。例如，从 ports 安装 Apache 2.2 可以通过置 `WITH_LDAP` `make(1)` 选项来启用 `mod_ldap`：

```
# cd /usr/ports/www/apache22
# make WITH_LDAP="YES" install clean
```

看看 FreeBSD 手册的 4.5 小节，[使用 Ports Collection](#)，以获取更多关于 Ports Collection 的信息。

3.3. Ports 是 packages，我如何使用它们？

Packages 就是很好的 ports，所以从源 (ports) 安装与从二进制 packages 安装两者都有很大不同。它们方法各有自己的特点：

Packages (二进制)

- 更快速的安装（比大的应用程序加起来会花很长时间）。
- 不需要知道如何安装软件。
- 不需要在操作系统上安装设备。

Ports (源)

- 能定制安装。（Packages 通常都是使用标准构建的。使用 ports 能定制各个包的，比如构建附加的模块或是更改安装路径之类的。）
- 如果喜欢的话可以使用自己的补丁。

如果没有一些特殊的需求，packages 可能最好最合用的情况。如果需要定制，ports 是最合用的方法了。（记得，如果需要定制而自己又更倾向于使用 packages，可以使用 `make`package`` 从 ports 构建一个定制的 package，然后移植到其他的服务器。）

4. 系统初始化：行如何在里面？

Linux® 使用 Sysv init 初始化系统，而 FreeBSD 使用的是 BSD 风格的 `init(8)`。在 BSD 风格的 `init(8)` 中没有行和 `/etc/inittab`，代替控制的是 `rc(8)` 应用程序。`/etc/rc` 脚本取 `/etc/defaults/rc.conf` 和 `/etc/rc.conf` 文件来决定哪些服务将被启动。特殊服务在此后由位于 `/etc/rc.d/` 和 `/usr/local/etc/rc.d/` 下的相应服务初始化脚本文件所启动。这些脚本类似于位于 Linux® 系统中的 `/etc/init.d/` 目录下的脚本。

何会有个服务初始化脚本的目录？`/etc/rc.d/` 下的脚本是属于 "基本" 系一部分的
用程序所使用的。（[cron\(8\)](#)，[sshd\(8\)](#)，[syslog\(3\)](#)，以及其他。）`/usr/local/etc/rc.d/`
下的脚本是用安装的程序如 Apache, Squid 等使用的。

"基本" 系和用安装的程序之间的区别是什么？FreeBSD 是一套出来的完整的操作系统，也就是
内核，系，有程序（如 [ls\(1\)](#)，[cat\(1\)](#)，[cp\(1\)](#) 等）全部被做一个整体一起并出。
就是被属于 "基本" 系的程序。用安装的程序并不是 "基本" 系的一部分，如 Apache, X11,
Mozilla Firefox, 等等。些用安装的程序通常是使用 FreeBSD 的 Packages 和 Ports
套件安装上去的。了将些程序和 "基本" 系区分来，用安装的程序通常被安装到 `/usr/local/`
下。因此用安装的二进制文件存在于 `/usr/local/bin` 下，配置文件在 `/usr/local/etc` 下，以此推。

可以通过在 `/etc/rc.conf` ([rc.conf\(5\)](#)) 文件中加之的 `ServiceName_enable="YES"` 配置来用服务。
看一下系默认的 `/etc/defaults/rc.conf` 文件，些默认配置可以使用 `/etc/rc.conf` 文件来改。因此，
当安装附加程序最好回下文来决定到底如何任何的服务。

下面的一小段内容用来在 `/etc/rc.conf` 中用 [sshd\(8\)](#) 和 Apache 2.2。指定了 Apache 通 SSL 方式。

```
# enable SSHD
sshd_enable="YES"
# enable Apache with SSL
apache22_enable="YES"
apache22_flags="-DSSL"
```

一旦服务已在 `/etc/rc.conf` 中用，服务将能从命令行（不需要重新系）：

```
# /etc/rc.d/sshd start
```

如果服务没有被用，可以使用 `forrestart` 来从命令行：

```
# /etc/rc.d/sshd forrestart
```

5. 网配置

5.1. 网接口

代替 Linux® 中所使用的网接口所常用的 `ethX` 格式的是，FreeBSD 使用名字后跟一个数字来。
下面 [ifconfig\(8\)](#) 的输出示了个 Intel® Pro 1000 的网接口（`em0` 和 `em1`）：

```
% ifconfig
em0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    options=b<RXCSUM, TXCSUM, VLAN_MTU>
    inet 10.10.10.100 netmask 0xffffffff broadcast 10.10.10.255
    ether 00:50:56:a7:70:b2
    media: Ethernet autoselect (1000baseTX <full-duplex>)
    status: active
em1: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    options=b<RXCSUM, TXCSUM, VLAN_MTU>
    inet 192.168.10.222 netmask 0xffffffff broadcast 192.168.10.255
    ether 00:50:56:a7:03:2b
    media: Ethernet autoselect (1000baseTX <full-duplex>)
    status: active
```

5.2. IP配置

一个 IP 地址可以使用 [ifconfig\(8\)](#) 来指定到一个网口接口。通常，要保持重启后依然能使用的 IP 配置信息需要包含在 `/etc/rc.conf` 中。下列例子指定了主机名，IP 地址，以及默认网关：

```
hostname="server1.example.com"
ifconfig_em0="inet 10.10.10.100 netmask 255.255.255.0"
defaultrouter="10.10.10.1"
```

使用下面内容来网口接口配置 DHCP：

```
hostname="server1.example.com"
ifconfig_em0="DHCP"
```

6. 防火墙

像 Linux® 中的 IPTABLES 一样，FreeBSD 也提供了一个内核的防火墙；同时 FreeBSD 提供了三个防火墙：

- [IPFIREWALL](#)
- [IPFILTER](#)
- [PF](#)

IPFIREWALL 或是 IPFW（管理 IPFW 的 [ipfw\(8\)](#) 命令）是 FreeBSD 作者同时并行的。IPFW 能与 [dummynet\(4\)](#) 配合使用来提供流量整形功能以及模拟不同网口接口类型的功能。

允许 SSH 进入的 IPFW 规则如下：

```
ipfw add allow tcp from any to me 22 in via $ext_if
```

IPFILTER 是 Darren Reed 所写的防火墙程序。不是 FreeBSD 的，它已被移植到 NetBSD, OpenBSD, SunOS, HP/UX, 还有 Solaris 等一些操作系统之上。

允许 SSH 进入的 IPFILTER 命令例如如下：

```
pass in on $ext_if proto tcp from any to any port = 22
```

最后一个防火墙程序，PF，是 OpenBSD 项目所写的。PF 是被作为 IPFILTER 的一个替代品而被建出的。就特点而言，PF 的语法与 IPFILTER 的非常相似。PF 可以与 [altq\(4\)](#) 配合来提供 QoS 的特性。

允许 SSH 进入的 PF 命令例如如下：

```
pass in on $ext_if inet proto tcp from any to ($ext_if) port 22
```

7. 升级 FreeBSD

共有三种方法来升级 FreeBSD 系统：源码，二进制更新，还有安装光盘。

从源码升级是最慢的升级方法，但是提供了最棒的系统活性。这个过程包含了使用 FreeBSD CVS（并行版本系统）来同步一个本地的 FreeBSD 源代码。一旦本地源码已更新到当前最新，便可以构建新版本的内核以及应用程序。关于源码更新的更多信息可参阅 FreeBSD 手册 [关于如何更新操作系统的章节](#)。

二进制更新类似于使用 `yum` 或 `apt-get` 更新 Linux® 系统。`freebsd-update(8)` 命令会获取新的更新并安装它们。这些更新可以通过 `cron(8)` 使用程序来调度。



如果使用 `cron(8)` 来调度更新，确信在你的 `crontab(1)` 中使用了 `freebsd-update cron` 来控制大数目的机器同时获取更新。

```
0 3 * * * root /usr/sbin/freebsd-update cron
```

最后一个更新的方法，从安装光盘来升级，是个直接的过程。从安装光盘并安装来更新。

8. procfs：已是过去式但仍未被遗忘

Linux® 中，可能会通过看一看 `/proc/sys/net/ipv4/ip_forward` 来判定 IP 是否被启用。在 FreeBSD 中则使用 `sysctl(8)` 来查看和其他方面的系统配置，在当前的 FreeBSD 版本中 `procfs(5)` 已不推荐使用。（当然 `sysctl` 在 FreeBSD 也同样可用。）

在 IP 配置例中，使用下列内容来判定 FreeBSD 系统中是否已启用了 IP 转发：

```
% sysctl net.inet.ip.forwarding
net.inet.ip.forwarding: 0
```

`-a` 选项用来列出所有的系统配置：

```
% sysctl -a
kern.ostype: FreeBSD
kern.osrelease: 6.2-RELEASE-p9
kern.osrevision: 199506
kern.version: FreeBSD 6.2-RELEASE-p9 0: Thu Nov 29 04:07:33 UTC 2007
    root@i386-builder.daemonology.net:/usr/obj/usr/src/sys/GENERIC

kern.maxvnodes: 17517
kern.maxproc: 1988
kern.maxfiles: 3976
kern.argmax: 262144
kern.securelevel: -1
kern.hostname: server1
kern.hostid: 0
kern.clockrate: { hz = 1000, tick = 1000, profhz = 666, stathz = 133 }
kern.posix1version: 200112
...
```



某些 `sysctl` 的参数是只读的。

需要 `procfs` 的情况是，运行一些古老的软件，使用 `truss(1)` 来跟踪系统信号，以及 [Linux® 二进制兼容](#)。（尽管，Linux® 二进制兼容性使用其本身的 `procfs`，`linprocfs(5)`。）如果需要挂载 `procfs` 可以在 `/etc/fstab` 中加入如下内容：

```
proc                /proc              procfs rw,noauto      0      0
```



`noauto` 会防止 `/proc` 在启动时被自动挂载。

然后使用如下命令挂载 `procfs`：

```
# mount /proc
```

9. 常用命令

9.1. 软件包管理

Linux® 命令 (Red Hat/Debian)	FreeBSD 等价命令	目的
<code>yum install package</code> / <code>apt-get install package</code>	<code>pkg_add -r package</code>	从远程安装 <code>package</code>
<code>rpm -ivh package</code> / <code>dpkg -i package</code>	<code>pkg_add -v package</code>	安装 <code>package</code>
<code>rpm -qa</code> / <code>dpkg -l</code>	<code>pkg_info</code>	列出已安装的软件包

9.2. 系管理

Linux® 命令	FreeBSD 等价命令	目的
<code>lspci</code>	<code>pciconf</code>	列出 PCI 卡
<code>lsmod</code>	<code>kldstat</code>	列出已加载的内核模块
<code>modprobe</code>	<code>kldload / kldunload</code>	加载/卸载内核模块
<code>strace</code>	<code>truss</code>	跟踪系统调用

10. 结语

非常希望这篇文档能给予足够的帮助来开始的 [FreeBSD](#) 之路。有必要再去看一下 [FreeBSD 手册](#) 所提到的但并没有被包含在本文档中的那些更深入广泛的主题。