

LINUX POCKET GUIDE

第2版

LINUX

口袋书



DANIEL J. BARRETT 著
欧阳立博、严小商、任桥伟 译

O'REILLY®



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

O'REILLY®

Linux 口袋书 (第2版)

Linux Pocket Guide, Second Edition

Daniel J.Barrett 著

欧阳立博 严小商 任桥伟 译

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书是一本精简的 Linux 入门指南,简要阐述了 Linux 的基本观念,涵盖 Linux 的许多命令和功能,并以精辟的范例示范如何利用它们进行日常工作,能够让你在短时间内有效提升工作效率。

©2012 by O'Reilly Media,Inc. Simplified Chinese Edition, jointly published by O'Reilly Media,Inc.and Publishing House of Electronics Industry, 2013.Authorized translation of the English edition,2012 O'Reilly Media,Inc.,the owner of all rights to publish and sell the same. All rights reserved including the rights of reproduction in whole or in part in any form.

本书简体中文版专有出版权由O'Reilly Media,Inc.授予电子工业出版社。未经许可,不得以任何方式复制或抄袭本书的任何部分。专有出版权受法律保护。

版权贸易合同登记号 图字: 01-2013-1793

图书在版编目(CIP)数据

Linux 口袋书 / (美) 巴瑞特 (Barrett,D.J.) 著; 欧阳立博, 严小商, 任桥伟译. —2 版. —北京: 电子工业出版社, 2013.6
书名原文: Linux pocket guide,2nd edition
ISBN 978-7-121-20307-7

I. ①L… II. ①巴… ②欧… ③严… ④任… III. ①Linux 操作系统 IV. ①TP316.89

中国版本图书馆 CIP 数据核字(2013)第 092350 号

责任编辑: 孙学瑛

封面设计: Karen Montgomery 张健

印 刷: 涿州市京南印刷厂

装 订: 涿州市京南印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 850×1168 1/40 印张: 5.5 字数: 176 千字

印 次: 2013 年 6 月第 1 次印刷

定 价: 29.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

译者序

昨天参加一个 career talk 的活动，有邀请几位嘉宾作一些有关职业规划的讲座，其中一位前辈着重强调了 Mindset（心态）的问题：当我们是一个普通 engineer 的时候，我们这么写自己的 report “I worked on …”，而当我们 lead 一个 team 的时候，变成这么写 “We worked on …”，我们一直在简单地罗列自己参与的工作，而实际上不管是 I 或者是 We，我们都应该这么写 “I identified …, I contributed …”，去强调我们的贡献及我们的工作对他人的影响。

回顾一下自身，貌似在面对类似的场景时，我也一直在进行着各种重复：博客访问量……，写了……，写了……，进行了多少公益或非公益的讲座，……。从未或者说很少真正地沉下心去想想自己这些工作为他人贡献了些什么，又产生了多少有益的影响。

Anyway，回到本书，我相信它完全有能力成为每个人进入 Linux 的第一步，也相信它会成为很多 Linux 爱好者的随身伙伴，更希望在我以后的生活里，不会仅仅多

了这么一条“翻译了……”，而会是多了“翻译的……让 Linux 能够帮助到更多的人”。

最后，感谢集美大学的欧阳立博、我的同学严小商，他们完成了第一稿的翻译，还要感谢孙学瑛编辑、郑柳洁编辑，以及博文视点各位参与过本书出版的朋友，没有她们的努力，这本书将不可能去帮助更多需要的人。

任桥伟

2013 年 4 月 20 日于上海

目录

Linux 口袋书（第 2 版）	1
本书内容	1
获取帮助	6
初识 Linux	8
文件系统	13
Shell	21
基本文件操作	36
目录操作	41
显示文件内容	43
文件的创建与编辑	53
文件属性	58
文件位置	68
文件文本操作	77
文件压缩和打包	90
文件比较	95
打印	100
拼写检查	101
磁盘和文件系统	103

备份和远程存储	108
查看进程	113
进程控制	117
任务调度	120
登录、注销与关机	125
用户和其操作环境	126
用户账号管理	130
成为超级用户	134
组管理	135
主机信息	137
主机位置	141
网络连接	145
电子邮件	149
网络浏览	154
Usenet News	158
即时消息	160
屏幕输出	162
数学计算	167
日期和时间	170
图像和屏幕保护程序	174
音频	177
视频	181
软件安装	182
Shell 脚本编程	187
后记	200
索引	203

欢迎进入 Linux 的世界！对于 Linux 新人，本书可以作为一本入门指南，同时也可以作为一本常见和实用命令的快速索引。如果你已经拥有一定的 Linux 使用经验，则可以跳过前面这些介绍性的内容。

本书内容

本书只是一本精简的 Linux 入门指南，而不是一本完整全面的参考书。本书只介绍有助于你提高工作效率的重要内容，而不是逐条逐句地去详细解释每条命令及其参数（如果你没有找到自己想要的内容，请见谅），也不会去深入细致地探究操作系统的内部细节。精简、贴切和必要，是我们奉行的宗旨。

对于命令 (*command*) 的介绍是本书的重点。使用这些命令，你只需要输入很少的单词或字母就可以去指示一个 Linux 系统完成你所期望的工作。下面是一个命令的例子，用于统计文件 *myfiles* 的行数：

```
wc -l myfile
```

本书内容会涵盖对于一般用户而言最为重要的那些 Linux 命令，比如 *ls* (列出文件)，*grep* (在文件中查找指定文本)，*amarok* (播放音频文件)，以及 *df* (计算空闲的磁盘空间)。限于篇幅，对 GNOME 和 KDE 之类的图形窗口环境，我们将只作简单介绍。

2) 为了提供一条流畅的学习路线，本书以功能为依据来编排内容。例如，我们将所有涉及文件内容显示的命令集中在一起进行介绍，`cat` 命令用于显示短文本文件，`less` 命令用于显示长文本文件，`od` 命令用于显示二进制文件，`acro read` 命令用于显示 PDF 文件等。然后，我们依次对每个命令进行解释，并提供其典型用法和常用选项的说明。

我们假设你有一个 Linux 系统的账号，并且知道如何使用你的账号及密码登录系统。否则，请咨询你的系统管理员；如果是你自己的系统，那么可以使用你在安装 Linux 时所创建的账号。

何谓 Linux

Linux 是一个流行的开源操作系统，可以与微软的 Windows 和苹果的 Mac OS 相提并论。有两种操作 Linux 系统的方式。

- 图形用户接口，由窗口、图标、鼠标控制组成；
- 命令行接口，又称作 *shell*，可以输入和运行各种命令，比如前面曾提到过的命令 `wc`。

同样，Windows 和 Mac OS 也可以使用命令行（Windows 使用其 `cmd` 和 PowerShell 命令工具，OS X 使用其终端应用程序），但是它们的用户大多并不使用命令行方式操作。然而，在 Linux 系统中，*shell* 是非常关键的，如果使用 Linux 而不用 *shell*，那么你正在错过一些很重要的体验。

何谓发行版

Linux 拥有非常高的可配置性，包含了数以千计的应用程序。因此，出现了各种不同版本的 Linux，来满足不同人的需求和偏好。虽然它们都具有相同的核心，但是却包含不同的程序和文件，外表看起来也各不相同。每种 Linux，也被称为一个 *distro*（发行版，“distribution”的简称）。流行的发行版有：Ubuntu Linux、Red Hat Enterprise Linux、Slackware、Mint 等。本书涵盖了应用

3) 于每个发行版的那些核心部分。

何谓命令

一条 Linux 命令通常由程序名 (*program name*) 及其后的选项 (*option*) 和参数 (*argument*) 组成。命令需要在 shell 环境里输入，比如：

```
$ wc -l myfile
```

程序名 (`wc`, 统计字数) 指向位于磁盘上的一个程序, shell 可以根据该名称定位程序位置并执行。选项通常以“-”开头, 用来影响程序的行为。上述命令中,“-l”选项告诉程序 `wc` 对行数而不是字数进行统计。参数 `myfile` 指定了程序 `wc` 应读取和处理的文件。最前面的美元符号 (\$) 是 shell 的提示符, 表明 shell 正在等待命令输入。

同一个命令可以有多个选项和参数, 它们可以被分开指定：

```
$ wc -l -w myfile (有两个分开的选项)
```

也可以被合并在一起：

```
$ wc -lw myfile (与 -l -w 功效相同)
```

然而, 并不是所有的程序都能够接受合并的选项。多参数也是可行的, 比如：

```
$ wc -l myfile1 myfile2 (统计两个文件的行数)
```

选项的含义并没有被标准化。相同的选项符号 (如 `-l`) 对于不同的程序可能具有不同的含义：同样的选项 `-l`, 对于程序 `wc`, 其含义是“统计文本行数”, 而对于程序 `ls`, 其含义则改变为“更长的输出”。另一方面, 两个不同的程序可能使用不同的选项, 来表达相同的含义。比如 `-q` (run quietly) 与 `-s` (run silently)。

同样, 参数也没有被标准化。它们通常是输入或输出文件名, 但也可以是其他东西, 如目录名或正则表达式。

相比一个单纯带选项的程序, 命令要更为复杂和有趣。

- 同一个命令可以同时运行多个程序，这些程序可以被依次执行，也可以使用上一个命令的输出作为下一个命令输入的管道操作方式同时执行。Linux 高手往往善于使用管道方式。
- Linux 命令行接口，即 *shell* 有一套内置的编程语言。因此，你的命令不仅可以这样写：“运行这个程序”，也可以这样写：“如果今天是星期二，则运行这个程序；否则，对每个扩展名为 *.txt* 的文件，运行另外一个程序 6 次”。

如何阅读本书

在这本书中，我们将描述许多 Linux 命令。每个命令的描述都有一个标准格式的标题。

图 1 显示了 `ls`（列出文件）命令的标题，它用一种简单的格式描述了 `ls` 命令的一般用法：

```
ls [options] [files]
```

这意味着你必须首先输入“`ls`”，是否提供选项及文件名可以视情况而定。方括号“`[`”和“`]`”并不需要实际进行输入，它们只是用来强调其所包含的内容是可选的。斜体字部分表示应该被替换成实际的特定值，比如实际的文件名。如果你看到两个选项或参数之间用 `|` 符号隔开，同时在它们外边可能还有一对圆括号：

```
( file | directory )
```

这意味着你将不得不面临选择：使用文件名作为参数，或者使用目录名称作为参数。

图 1 显示的命令标题包含了一个命令的 6 个属性，深色表示支持，浅色表示不支持。

stdin

命令从标准输入读取数据，例如，默认为键盘。参见第 11 页的“输入和输出”。

```
ls                               stdin  stdout  -file  --opt  --help  --version  
ls [options] [files]
```

图1：标准的命令标题

stdout

命令将数据送到标准输出，例如，默认为屏幕。参见第 11 页的“输入和输出”。

-file

当用“-”符号代替输入文件名时，命令将从标准输入读取数据；同样，如果用“-”代替输出文件名，则意味着命令将向标准输出写入数据。例如，下面的例子中，`wc` 命令依次从文件 *file1*，文件 *file2*，标准输入，文件 *file3* 中读取数据。

```
$ wc file1 file2 - file3
```

--opt

如果选项包括“--”符号，则意味着选项到此结束；其后出现的任何命令行部分均不能作为选项。在处理文件名以“-”符号开始的文件时，这种方式十分必要，否则这样的文件名将很容易被误认为是选项。例如，如果一个文件名为 *-foo*，那么直接运行命令 `wc -foo` 将会失败，因为 `-foo` 将会被误认为是选项，而 `wc -- -foo` 则可以正确运行。如果命令不支持“--”方式，你还可以在文件名前面加上当前路径“./”，如此一来，“-”符号将不会再被误认为是选项首字符，比如：

```
$ wc ./-foo
```

--help

选项 `--help` 告诉命令打印帮助信息，显示该命令的正确用法而后退出。

--version

选项 `--version` 告诉命令打印版本信息，而后退出。

6 Shell 提示符

本书中某些命令只有使用超级用户 (*superuser*) 才能成功运行。超级用户是可以在系统里进行任何操作的特定用户。这种情况下，我们使用“#”作为 shell 的提示符：

```
# superuser command goes here
```

否则，对于普通用户，我们将用“\$”作为 shell 提示符：

```
$ ordinary command goes here
```

按键

本书使用特定的符号表示按键。像许多其他的 Linux 文档一样，我们使用“^”符号表示“按住 Ctrl 键不放”。例如，“^D”（发音“control D”）表示“按住 Ctrl 键不放，然后再按下 D 键”。我们也以 Esc 表示“按下 Esc 键”。回车键和空格键之类的按键就不再解释了。

你的朋友，echo 命令

在本书的许多例子中，我们使用 echo 命令（详见第 162 页的“屏幕输出”）打印屏幕信息。echo 是最简单的命令之一：它仅仅将 shell 处理过的参数显示到标准输出。

```
$ echo My dog has fleas
My dog has fleas
$ echo My name is $USER // 环境变量 USER
My name is smith
```

获取帮助

如果本书提供的内容不能满足你的需求，你可以做以下几件事情来获得帮助。

7 使用 man 命令

man 命令用于查看指定程序的在线说明，即所谓的 *manpage*。例如，若想查看 ls 命令的用法，可以运行：

```
$ man ls
```

对于某些不明确的命令，可以通过 `-k` 选项指定关键词来查找某个特定主题的说明：

```
$ man -k database
```

使用 `info` 命令

`info` 命令提供了比 `man` 命令更为详尽的说明内容，很多 Linux 程序都对其进行支持。

```
$ info ls
```

当 `info` 命令正在运行时，我们可以使用一些按键进行辅助。

- 按 `h` 键获取帮助
- 按 `q` 键退出
- 使用空格键和回退键进行前、后翻页
- 按 `Tab` 键在超级链接之间转跳
- 按 `Enter` 键进入超级链接

如果某个给定程序没有提供相应的 `info` 文件，`info` 命令将显示该程序的 `manpage`。单独运行 `info` 命令可以列出所有可用的文档。要了解如何操作 `info` 命令，可以运行 `info info`。

使用 `--help` 选项（任何）

许多 Linux 命令都会响应 `--help` 选项，并输出一些简短的帮助信息。

```
$ ls --help
```

如果输出超出一屏，则将帮助信息通过管道方式传递给 `less` 程序进行多页显示（可以按 `q` 键退出）：

```
$ ls --help | less
```

检查 `usr/share/doc` 目录

这个目录包含了许多程序的说明文件。这些文件通常按程序名字及版本进行组织。例如，文本编辑器 `emacs` 23 版的

说明文件可能会在目录 `/usr/share/doc/emacs23` 下找到（依赖于具体的发行版）。

GNOME 和 KDE 帮助

访问 <http://www.gnome.org> 或 <http://www.kde.org> 可获取 GNOME 或 KDE 的帮助。

版本说明网址

大多数 Linux 发行版都有官方网站，上面有文档、用来提问与解答的论坛，以及其他资源。只要在任何流行的搜索引擎中输入发行版的名字（如，“Ubuntu”），都可以找到它的网站。也可以访问本书的网站：<http://shop.oreilly.com/product/0636920023029.do>。

Linux 帮助网站

有许多专门的 Linux 网站可以供我们寻找问题的答案，比如：<http://www.linuxquestions.org>，<http://unix.stackexchange.com>，<http://www.linuxhelp.net>，以及 <http://www.linuxforums.org>。

网页搜索

针对一些特定的 Linux 错误信息，通过在搜索引擎里进行查找，你很可能找到一些比较有帮助的结果。

初识 Linux

Linux 有四个主要的部分。

内核

涵盖底层操作系统、文件管理、磁盘、网络，以及其他我们认为必需的部分。大多数用户都几乎不会注意到内核。

支撑程序

包括数以千计的程序用于文件操作、文档编辑、数学计算、网页浏览、音频、视频、计算机编程、排版、加密、DVD 刻录……应有尽有。

Shell

用于输入命令、执行命令及显示命令结果的用户接口。Linux 有多种 shell : Bourne shell、Korn shell 和 C shell 等。本书内容基于 bash, 即 Bourne-Again Shell, 它通常也是用户的默认 shell。不管怎样, 所有 shell 都具有相似的基本功能。

X

提供窗口、菜单、图标、鼠标支持, 以及其他常见 GUI 组件的图形系统。其他更加复杂的图形环境都是基于 X 的基础上建立的, 其中最流行的是 KDE 和 GNOME。我们将会讨论几个打开 X window 并运行的程序。

本书内容集中在上面的第二、第三部分: 即支撑程序和 shell。

图形化桌面

登录 Linux 系统时, 欢迎你的可能是如图 2 所示的图形化桌面^{注1}。它包含:

- 主菜单或任务栏。它有可能位于屏幕的顶部、底部或者边上, 这要取决于你的发行版及系统设置。
- 表示计算机的电脑图标, 表示个人根目录的文件夹、垃圾桶, 以及其他更多的东西。
- 应用程序图标, 如 Firefox 网页浏览器及 Thunderbird 电子邮件程序。
- 开、关窗口及同时运行多个桌面的控件。
- 一个时钟, 以及其他的零星小图标。

10

注 1: 如果是通过网络远程登录, 则看到的是命令行环境。

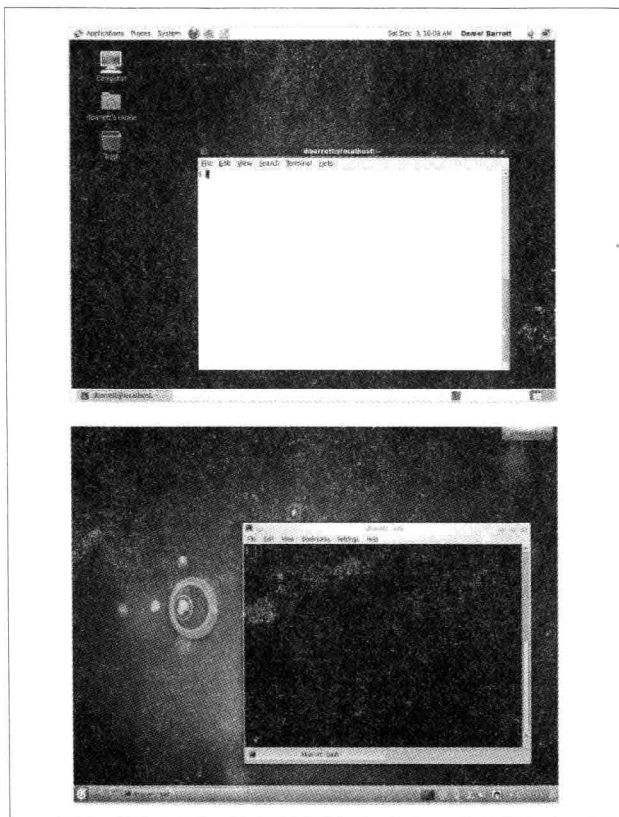


图2：图形桌面（CentOS Linux的GNOME，Ubuntu的KDE）。桌面的样式取决于你的发行版本及系统设置

- 11 Linux 系统有好几个图形界面，GNOME 和 KDE 是其中最为常用的。可以通过如下方式确定你的系统正在使用哪种图形界面：点击系统中相当于主菜单或开始菜单的地方，查找 GNOME、KDE、Kubuntu（基于 Ubuntu Linux 的 KDE）或类似的词。