红旗Linux系统开发技术

北京中科红旗软件技术有限公司　编著

石油工业出版社

红旗 Linux 系统开发技术

北京中科红旗软件技术有限公司　编著

石油工业出版社

内 容 提 要

本书全面介绍了 Linux 系统的开发方法，主要内容包括 Linux 的软件开发环境，包括编译、调试、维护工具以及广泛应用的集成开发环境；Linux 的文件系统、I/O 系统调用、系统数据文件；Linux 进程模型，包括进程关系、进程控制、进程间通信、信号处理；终端 I/O 编程；网络套接字编程；守护进程、系统服务；多线程编程等。本书结合大量实例，概念清晰，内容丰富翔实，特色鲜明，实用性强。

本书可作为从事 Linux 操作系统开发人员的参考用书。

图书在版编目（CIP）数据

红旗 Linux 系统开发技术/北京中科红旗软件技术有限公司编著．
北京：石油工业出版社，2013.1
ISBN 978-7-5021-9388-1

Ⅰ．红…
Ⅱ．北…
Ⅲ．Linux 操作系统
Ⅳ．TP316.89

中国版本图书馆 CIP 数据核字（2012）第 293260 号

出版发行：石油工业出版社
（北京安定门外安华里 2 区 1 号　100011）
网　址：http://pip.cnpc.com.cn
编辑部：（010）64255590　发行部：（010）64523620
经　　销：全国新华书店
印　　刷：北京中石油彩色印刷有限责任公司

2013 年 1 月第 1 版　2013 年 1 月第 1 次印刷
787×1092 毫米　开本：1/16　印张：29.5
字数：747 千字

定价：75.00 元
（如出现印装质量问题，我社发行部负责调换）

前　言

Linux 操作系统是可运行在多种硬件平台的个人计算机和工作站上的类 Unix 操作系统，它继承了 Unix 的全部优点，是真正的多用户、多任务、多平台的操作系统。在个人计算机和工作站上使用 Linux 操作系统，能充分发挥硬件的功能，使个人计算机能够作为工作站和服务器使用，提高工作站效率；Linux 符合 POSIX 标准，能同时支持 i386、MIPS、SPARC 和 PowerPC 等处理器；对于 Unix 上运行的软件，不用改动或稍加改动就可以运行在 Linux 操作系统上。

同 Unix 相比，Linux 要灵活得多。Linux 的用户界面非常友好，用户使用很方便。组成 Linux 操作系统的 Linux 内核以及各类开源组建受自由软件委员会指定的 GPL（General Public License）公用许可协议保护，Linux 系统内核及其大多应用软件的源代码都是向用户公开的。各地区系统发布商，根据各自地域文化和技术理念，在遵循 GPL 等开源软件协议的 Linux 内核与组件的基础上，加以工程化打造而成为可产品化的 Linux 操作系统，服务于社会应用需求。用户可使用 Linux 操作系统中自带的 C/C ++ 等编程语言编译器修改并编译自己喜欢的程序，同时也可以在 GNU 的许可下使用这些源程序的部分或全部。这种利用 Linux 源程序的可能性大大激发了世界范围内热衷于计算机事业的人们的兴趣和创造力，吸引了众多程序员为 Linux 工作。目前，所有 Unix 的主要功能都已经有相应的 Linux 工具和程序，大量的自由软件被移植到 Linux 操作系统上，世界各地流行着越来越多的 Linux 发行版本，如 Red Hat Linux、Red Flag Linux 等。在这些发行版本中，不仅包含 Linux 操作系统的内核，还包含各种编程语言的编译程序、数据库管理系统、图形用户界面、通信联网工具以及大量的其他应用软件。可见，Linux 已经是一个相当优秀的应用软件开发平台。

同 Windows 相比，Linux 也一样具有优势。Linux 强大的网络功能使 Internet 以及许许多多局域网中的服务器都被 Linux 占据，这不仅是由于 Linux 的稳定性相当出色，也是由于 Linux 的系统效率极高；与同类型的 Windows 服务器相比，

Linux 对硬件的要求可以降低一到两个档次，一台普通的 PC 机就能胜任复杂的网络服务工作；另外，Linux 下的 TCP/IP 网络方面的应用软件更是应有尽有，能实现许多在普通系统中无法实现的网络功能。在超级计算机应用领域中，以 500 强网站公布的数据可以看出，近 92% 的超级计算机系统是基于 Linux 构建的，这其中也得益于开源领域的集群技术的优势，为当今云计算技术架构的发展提供了坚实的技术基础。

不可否认，Linux 具有较 Unix 和 Windows 更卓越的性能，但价格却低得多。因此，如果想提高自己的计算机使用和编程水平，学习 Linux 确实是一条捷径。

在国家“核心电子器件、高端通用芯片及基础软件产品”重大专项进程中，作为基础软件主要承载平台的 Linux 操作系统得到极大重视，为此，需要更多的开发人员熟悉和了解基于 Linux 系统技术体系的开发技术。为促进 Linux 在我国的推广应用，让开发人员更快地步入 Linux 的奇妙世界，少走一些弯路，北京中科红旗软件技术有限公司红旗教育学院组织经验丰富的 Linux 编程人员，参考国际经典 Linux、Unix 编程书籍，编写本书，旨在充分发挥我国软件工作人员的积极性和创造力，振兴我国的软件事业。本书内容涵盖了 Linux / Unix 编程的主要方面，具有一定的深度，具体内容如下：

（1）第 1 章详细介绍 Linux 的软件开发环境，包括编译、调试、维护工具以及广泛应用的集成开发环境。

（2）第 2、3、4、5 章介绍 Linux 的文件系统、I/O 系统调用、系统数据文件方面的内容，本部分不仅详细介绍了传统 Unix 的文件、I/O 系统调用，还专门介绍了 Linux 文件、I/O 系统的特点。

（3）第 6、7、8、9 章介绍 Linux 进程模型，包括进程关系、进程控制、进程间通信、信号处理等方面的内容。

（4）第 10 章介绍终端 I/O 编程，它的用途主要包括：终端、计算机之间的连接、调制解调器、打印机等方面，因此非常复杂。

（5）第 11 章介绍网络套接字编程，socket 是最广泛使用和最成熟的网络编程接口规范，用来实现网络的互联。

（6）第 12 章介绍守护进程、系统服务等方面的内容，可以编写自己的守护进程来增加、增强系统服务。

（7）第13章介绍多线程编程，在操作系统层面上讲述了线程的层次和模型，还包括线程同步、互斥等方面的内容。

本书结合大量实例，概念清晰、内容丰富翔实、特色鲜明、实用性强。本书由北京中科红旗软件技术有限公司红旗教育学院组织编写，对参与本书编写的人员，在此表示感谢。由于本书涉及的内容丰富，加之篇幅、时间有限，书中难免存在纰漏和不足，欢迎读者批评指正。

北京中科红旗软件技术有限公司

2012 年 10 月

目　录

1 Linux 软件开发工具

1.1 gcc 和 g++

gcc 和 g++分别是 GNU 的 C 及 C++编译器，是 Linux 系统中将 C 及 C++语言源文件生成可执行程序的工具。

一个可执行的 C 或 C++程序需要经过如下四步生成：

第一步：由预处理程序对 C 或 C++语言源文件（*.c 或 *.cpp、*.C、*.cxx 等）进行宏扩展和条件处理，并导入前导文件，生成以 .i 为后缀的文件；

第二步：由编译程序将预处理后的文件（*.i）中的 C 或 C++语言源代码转换成汇编语言代码，生成以 .s 为后缀的汇编文件；

第三步：由汇编程序将汇编文件（*.s）中的汇编语言代码转换成目标代码（机器代码，*.o），生成以 .o 为后缀的目标文件；

第四步：由连接程序将目标文件（*.o）同指定的库文件进行连接，生成可执行程序。

若非特别指定，gcc 或 g++将自动调用上述工具，生成可执行程序。

gcc 和 g++命令的调用格式为：

```
gcc     [options]     filenames
g++     [options]     filenames
```

其中：options 为 gcc 和 g++命令的选项，可以为空；多个选项必需分开写，如“-dr”不同于“-d -r”。filenames 是 gcc 和 g++命令的输入文件列表，多个文件之间以空格分隔。

gcc 和 g++命令的选项很多，它们规定着 gcc 和 g++命令的行为；掌握 gcc 和 g++命令的关键就是正确使用它们的选项。

gcc 和 g++命令的选项大致可分为如下几类：

（1）全局选项：影响编译的全过程（从预处理到连接）；

（2）语言选项；

（3）预处理程序选项；

（4）汇编程序选项；

（5）连接程序选项；

（6）目录选项；

（7）优化选项；

（8）调试选项；

（9）警告选项。

下面分别介绍各类的常用选项（除非特殊声明，选项将同时适用于 gcc 和 g++命令）。

1.1.1 全局选项

1.1.1.1 -x language filenames

-x language filenames 选项指明 filenames 文件的编程语言（C 语言、C++语言等）及 gcc/g++编译的起始阶段（预处理、编译、汇编、连接）。

language 的可选值为 C、C++、assembler-with-cpp 等。其中：C、C++指明输入文件为 C 语言、C++语言文件，编译过程从预处理阶段开始；assembler、assembler-with-cpp 指明编译过程从汇编阶段开始。

-x language 选项对其后的多个文件都有效，直至遇到下一个-x 为止。

在不使用此选项时，gcc/g++利用文件名后缀来确定语言及编译的起始阶段。如：输入文件名以.c 或.cpp 为后缀时，从预处理阶段开始编译；以.i 为后缀时，从编译阶段开始编译；以.s 为后缀时，从汇编阶段开始编译；以.o 为后缀时，从连接阶段开始编译。使用-x 选项后，文件名后缀将不再对编译的起始阶段起作用。

1.1.1.2 -x none filenames

-x none filenames 选项使-x language 选项失去作用，其后文件以名字后缀决定语言及编译的起始阶段。

1.1.1.3 -c、-S、-E 选项

-c、-S、-E 这三个选项决定编译过程到哪一个阶段（预处理、编译、汇编、连接）结束，它们不能同时使用。

-c 选项激活预处理、编译及汇编程序，不执行连接程序；编译过程到汇编阶段结束；输出文件为目标文件（*.o），可用-o 选项另行指定输出文件名。

-S 选项激活预处理及编译程序，不执行汇编和连接程序；编译过程到编译阶段结束；输出文件为汇编文件（*.s），可用-o 选项另行指定输出文件名。

-E 选项激活预处理程序，不执行编译、汇编和连接程序；编译过程到预处理阶段结束；输出被送到标准输出，可用-o 选项重新定向到文件输出。

1.1.1.4 -o filename

-o filename 选项以 filename 覆盖 gcc/g++缺省的输出文件名；它可以同-c、-S、-E中的任何一个选项一同使用，为各阶段的输出文件（可执行文件、目标文件、汇编文件、预处理后的 C/C++语言源文件）改名。

因为-o 选项只能指定一个输出文件名，所以当有多个输出文件产生时，-o 选项不起作用。

如不使用此选项，缺省情况下，可执行程序以 a.out 命名，目标文件以*.o 命名，汇编文件以*.s 命名，预处理后的 C/C++语言源文件定向到标准输出。

1.1.1.5 -pipe

-pipe 选项用管道代替临时文件来处理不同编译阶段（预处理、编译、汇编、连接）