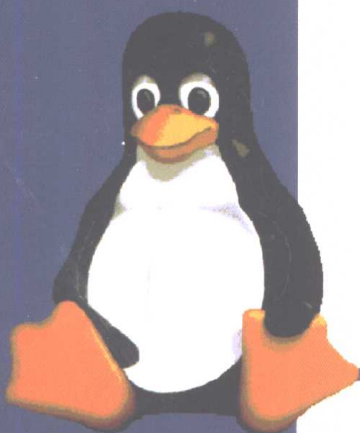




Linux 专家之路



Linux

下的Perl编程

郑武 老马 编著 雨人科技 策划





Linux 专家之路



Linux

下的Perl编程

郑武 老马 编著 雨人科技 策划

人民邮电出版社

图书在版编目 (CIP) 数据

Linux 下的 Perl 编程 / 郑武, 老马编著. —北京: 人民邮电出版社, 2001.11

(Linux 专家之路)

ISBN 7-115-09787-9

I. L... II. ①郑... ②老... III. PERL 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2001) 第 074383 号

内 容 提 要

本书是一本介绍 Linux 平台上 Perl 5 编程语言的专业参考书。书中详细地讲解了 Perl 语言的基本概念、基本数据结构和语法规则, 探讨了使用 Perl 语言进行 CGI 编程技术, 还讲述了如何使用 Perl 语言进行数据库的开发, 最后介绍了 Perl 语言在网络中的应用, 讲解如何使用 Perl 语言进行网络编程。

本书可用作 Perl 5 初级用户的入门教程, 也可作为中高级用户的 Perl 5 快速查询手册。

MS186/01

Linux 专家之路

Linux 下的 Perl 编程

◆ 编 著 郑 武 老 马

策 划 雨人科技

责任编辑 张瑞喜

执行编辑 郭立罡

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子函件 315@ pptph.com.cn

网址 <http://www.pptph.com.cn>

读者热线: 010-67129212 010-67129211(传真)

北京汉魂图文设计有限公司制作

北京顺义振华印刷厂印刷

新华书店总店北京发行所经销

◆ 开本: 787 × 1092 1/16

印张: 28.75

字数: 696 千字

2001 年 11 月第 1 版

印数: 1 - 4 000 册

2001 年 11 月北京第 1 次印刷

ISBN 7-115-09787-9/TP·2536

定价: 48.00 元(附光盘)

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

前 言

随着因特网的普及，掌握一门网络编程语言已经成为用户一种迫切的需要，而 Perl 5 正是当前网络编程中最常用的语言。与其他语言相比，Perl 5 具有明显的优势：语法简单灵活，可移植性好，具有强大的字符串处理能力，可以进行高效率的 CGI 编程。尤其值得一提的是，由于 Perl 的开放性，读者可以方便地从网上免费获得大量可重用的模块和源代码等资源。

本书深入浅出地讲解了 Perl 5 的基础知识及其高级应用。全书共分 17 章，第 1 章介绍了 Linux 操作环境和 Perl 5 编程语言的基本情况；第 2 章到 14 章讲解了 Perl 5 的基础知识，首先介绍了 Perl 5 的 3 种基本数据结构、操作符和控制语句，接着详细介绍了 Perl 5 最强大的功能——正则表达式和字符串处理，另外还介绍了函数、引用以及 I/O 操作和文件系统，在介绍了包和模块后，还对 Perl 5 中面向对象的编程进行了讨论，每个章节都有丰富的例子，读者通过它们可以很快地掌握 Perl 5 的使用。第 15 章到第 17 章分别介绍了 Perl 5 中的 CGI 编程、DBI 数据库编程和网络编程等 Perl 5 的高级应用。

本书每章后面均附有习题，以便读者练习掌握所学知识，并在书后面的附录中列有习题的参考答案。

本书配有光盘。光盘内容包括程序源代码，部分习题参考答案，雨人科技技术支持文档，PERL COMPILER、PERL EDITOR 和 WEB SERVER。

郑武负责全书的设计、统稿和修改，并编写了第 1 章、第 13 章、第 15 章、第 16 章和第 17 章，还负责了光盘内容的制作和例子的测试；老马负责编写了第 2 章至第 12 章和第 14 章；参加本书编排和指导工作的还有雨人科技的刘馨宇等同志。在此对所有关心、支持和帮助过本书编写工作的同志表示诚挚的谢意！

书中不足之处敬请读者指正，以便再版时修订。

联系邮件地址：yurennet@263.net

雨人科技网站：www.yurennet.com

编著者
2001 年 8 月

目 录

第 1 章 Linux 及 Perl 概述	1
1.1 Linux 系统简介	2
1.1.1 什么是 Linux	2
1.1.2 Linux 的发展历史	2
1.1.3 Linux 主要发行版本	3
1.1.4 Linux 的应用	6
1.1.5 谁在使用 Linux	7
1.2 Perl 介绍	8
1.2.1 Perl 概述	8
1.2.2 Perl 发展历史	10
1.2.3 Perl 的安装	11
1.2.4 Perl 资源	12
1.3 小结与练习	13
1.3.1 小结	13
1.3.2 习题与思考	13
第 2 章 标量数据	15
2.1 标量变量简介	16
2.1.1 何谓标量变量	16
2.1.2 标量变量的命名	17
2.2 数字标量	17
2.2.1 整型表示法	18
2.2.2 浮点型表示法	20
2.2.3 数字格式	21
2.2.4 定点数方案	21
2.3 字符串标量	22
2.3.1 单引号字符串	22
2.3.2 双引号字符串	23
2.3.3 反引号	26
2.3.4 引用操作符	27
2.4 常用的与标量有关的系统函数	27
2.4.1 数学函数	27
2.4.2 标量转换函数	28

2.5 小结与练习	29
2.5.1 小结	29
2.5.2 习题与思考	30
第 3 章 数组和列表数据	31
3.1 数组和列表的基本概念	32
3.1.1 什么是数组和列表	32
3.1.2 数组的表示	32
3.1.3 列表变量	34
3.2 数组元素的存取	35
3.2.1 列表的赋值	35
3.2.2 存取数组元素	36
3.3 对数组元素的其他操作	39
3.3.1 字符串中的方括号和变量替换	39
3.3.2 数组的输出	39
3.3.3 改变数组元素的值	40
3.4 与数组有关的函数	42
3.4.1 按字符顺序排序——sort 函数	43
3.4.2 反转数组——reverse 函数	44
3.4.3 数组去尾——chop 和 chomp 函数	44
3.4.4 数组连接/拆分——join/split 函数	44
3.4.5 push 和 pop 函数	45
3.4.6 shift 和 unshift 函数	46
3.4.7 拼接函数——splice 函数	46
3.5 小结与练习	47
3.5.1 小结	47
3.5.2 习题与思考	47
第 4 章 散列	49
4.1 散列简介	50
4.1.1 什么是散列	50
4.1.2 为什么要使用散列	50
4.1.3 散列变量	52
4.1.4 全局散列变量	54
4.2 存取散列的元素	54
4.2.1 访问散列中的元素	54
4.2.2 增加元素	55
4.2.3 数组与散列的相互转换	56

4.3 散列函数	57
4.3.1 keys 函数	58
4.3.2 values 函数	59
4.3.3 each 函数	60
4.3.4 delete 函数	60
4.4 散列应用举例	62
4.4.1 用散列实现多维数组	62
4.4.2 用散列实现结构	64
4.5 小结与练习	65
4.5.1 小结	65
4.5.2 习题与思考	65
第 5 章 操作符	67
5.1 数字操作符	68
5.1.1 基本数字操作符	68
5.1.2 数字比较操作符	69
5.1.3 赋值操作符	70
5.1.4 自增自减操作符	72
5.2 字符串操作符	74
5.2.1 字符串比较操作符	74
5.2.2 字符串联接和重复操作符	75
5.3 逻辑操作符和位操作符	77
5.3.1 逻辑操作符	77
5.3.2 位操作符	78
5.4 其他操作符	79
5.4.1 逗号操作符	79
5.4.2 条件操作符	80
5.5 操作符的优先级和结合性	82
5.6 小结与练习	84
5.6.1 小结	84
5.6.2 习题与思考	84
第 6 章 控制结构	87
6.1 控制结构中的有关问题	88
6.1.1 语句块	88
6.1.2 养成良好的编程习惯	88
6.2 条件语句	90
6.2.1 if 系列语句	90

6.2.2	unless 语句	94
6.2.3	if 语句的嵌套	95
6.3	循环语句	96
6.3.1	while/until 循环	97
6.3.2	do while/until 循环	99
6.3.3	for 循环	100
6.3.4	foreach 循环	101
6.4	其他控制语句	103
6.4.1	标记修饰符	103
6.4.2	修改循环控制流的几个语句	104
6.4.3	表达式修饰符	107
6.4.4	&&和 	108
6.5	小结与练习	109
6.5.1	小结	109
6.5.2	习题与思考	110
第 7 章	正则表达式	113
7.1	正则表达式简介	114
7.1.1	什么是正则表达式	114
7.1.2	一个简单的例子	114
7.2	正则表达式的构成	115
7.2.1	元字符	115
7.2.2	原子	116
7.3	正则表达式的正则	116
7.3.1	单字符类型	117
7.3.2	组合类型	118
7.3.3	定位类型	121
7.3.4	优先级	122
7.3.5	正则表达式正则的应用举例	122
7.4	模式修饰符	127
7.4.1	全局匹配	127
7.4.2	忽略大小写的匹配	129
7.4.3	多行匹配与单行匹配	130
7.4.4	单次匹配	131
7.4.5	多余空格	132
7.5	与正则表达式相关的其他问题	133
7.5.1	模式匹配变量	133
7.5.2	替代函数	134

7.5.3 内插变量.....	135
7.5.4 模式次序变量.....	136
7.5.5 扩展模式匹配.....	137
7.6 小结与练习	138
7.6.1 小结.....	138
7.6.2 习题与思考.....	139
第 8 章 字符串操作	141
8.1 字符串和数值的相互转换	142
8.1.1 数字与字符的相互转换.....	142
8.1.2 数字与字符串的相互转换.....	143
8.2 把标量插入字符串	144
8.3 字符串处理函数	145
8.3.1 改变大小写函数.....	146
8.3.2 字符串截尾函数.....	147
8.3.3 搜寻子字符串函数.....	149
8.3.4 子字符串的展开与替代.....	150
8.3.5 字符串转换函数.....	153
8.3.6 取字符串长度函数.....	154
8.3.7 分离和连接字符串函数.....	155
8.4 小结与练习	157
8.4.1 小结.....	157
8.4.2 习题与思考.....	158
第 9 章 自定义函数	161
9.1 一些常用的系统函数及其使用	162
9.2 自定义函数的定义	165
9.3 函数的调用	166
9.3.1 先定义后调用.....	167
9.3.2 用&调用	167
9.3.3 前向引用.....	168
9.3.4 用 do 调用	168
9.3.5 函数调用的返回值.....	168
9.4 更方便地使用函数	170
9.4.1 在函数中使用参数.....	170
9.4.2 在函数定义中使用私有变量.....	171
9.4.3 使用半私有变量.....	173
9.4.4 文件层级的 my 变量	175

9.5 Perl 5 中预定义的函数	176
9.6 函数的应用举例	176
9.6.1 二进制和十进制之间的转换函数	177
9.6.2 将角度转换为弧度	177
9.6.3 求一个数的对数	178
9.6.4 数字和英文的对应	178
9.6.5 一个较复杂的例子	180
9.7 小结与练习	182
9.7.1 小结	182
9.7.2 习题与思考	182
第 10 章 引用	185
10.1 引用简介	186
10.1.1 什么是引用	186
10.1.2 引用操作符	186
10.2 引用变量类型简介	187
10.2.1 直接引用变量	188
10.2.2 常量引用变量	188
10.2.3 函数引用变量	188
10.2.4 符号引用变量	189
10.2.5 匿名引用变量	190
10.2.6 对象引用变量	192
10.3 直接引用变量的使用	192
10.3.1 使用标量引用	192
10.3.2 使用数组引用	194
10.3.3 使用散列引用	195
10.3.4 文件句柄的引用	197
10.4 访问引用的其他方法	198
10.4.1 使用大括号	198
10.4.2 使用向右箭头操作符	199
10.4.3 使用方括号	199
10.4.4 多维数组	199
10.4.5 利用引用传给函数多个数组以及返回多个数组或散列	200
10.5 小结与练习	202
10.5.1 小结	202
10.5.2 习题与思考	202
第 11 章 文件系统	205

11.1 文件操作	206
11.1.1 打开/关闭文件	206
11.1.2 读/写文件	208
11.1.3 删除文件	209
11.1.4 更改文件名	209
11.1.5 产生/删除文件链接	210
11.1.6 判断文件状态	212
11.1.7 读取文件中的某一行	215
11.2 目录操作	216
11.2.1 产生/删除目录	216
11.2.2 打开/关闭目录	217
11.2.3 其他目录处理函数	217
11.3 更改文件和目录的权限、所有权和时间戳	219
11.3.1 更改权限	219
11.3.2 更改所有权	220
11.3.3 更改时间戳	220
11.4 使用 DBM 文件	222
11.4.1 打开/关闭 DBM 文件	222
11.4.2 DBM 散列的使用	223
11.5 小结与练习	223
11.5.1 小结	223
11.5.2 习题与思考	224
第 12 章 输入/输出处理	225
12.1 基本输入	226
12.1.1 来自 STDIN 的输入	226
12.1.2 输入操作符	227
12.1.3 @ARGV 数组	227
12.2 基本输出	229
12.2.1 print 函数	229
12.2.2 格式化输出函数 sprintf 和 printf	229
12.3 格式化输出	232
12.3.1 什么是格式	232
12.3.2 怎么定义格式	233
12.3.3 使用格式	234
12.3.4 字段容器	237
12.3.5 其他问题	240
12.4 小结与练习	241

12.4.1 小结	241
12.4.2 习题与思考	242
第 13 章 包和模块	243
13.1 使用 Perl 包	244
13.1.1 包的定义	244
13.1.2 包的引用和导入	245
13.1.3 包之间的切换	246
13.1.4 包和子程序	247
13.1.5 用包定义私有数据	247
13.1.6 访问符号表	248
13.2 使用 Perl 模块	250
13.2.1 创建模块	250
13.2.2 导入模块	251
13.2.3 模块使用举例	252
13.3 Perl 标准模块	257
13.4 小结与练习	267
13.4.1 小结	267
13.4.2 习题与思考	267
第 14 章 面向对象编程	269
14.1 对象与类简介	270
14.1.1 对象的特征	270
14.1.2 对象类	270
14.1.3 对象方法	271
14.1.4 对象变量	271
14.1.5 对象引用变量	271
14.2 用 Perl 5 创建类	272
14.2.1 构造函数	273
14.2.2 方法	277
14.2.3 Perl 类中的成员变量	281
14.2.4 析构函数	282
14.3 继承	283
14.4 方法的重载	287
14.4.1 子类方法的重载	287
14.4.2 操作符的重载	289
14.5 关于 Perl 面向对象编程的一些其他问题	292
14.6 小结与练习	295

14.6.1 小结	295
14.6.2 习题与思考	296
第 15 章 CGI 编程	297
15.1 Web 体系结构与 CGI 程序结构	298
15.1.1 Web 体系结构	298
15.1.2 CGI 程序结构	299
15.2 HTTP 协议	303
15.2.1 HTTP 通信模型	303
15.2.2 HTTP 请求	304
15.2.3 HTTP 响应	307
15.3 CGI 规范	308
15.3.1 CGI 程序工作原理	308
15.3.2 CGI 方法	309
15.3.3 环境变量	311
15.4 CGI 模块	314
15.4.1 使用 CGI 模块	314
15.4.2 创建动态文档	315
15.4.3 创建 HTML 元素	317
15.4.4 创建表单	321
15.5 计数器实现	331
15.5.1 设计思路	331
15.5.2 代码实现	334
15.6 留言簿实现	335
15.6.1 设计思路	335
15.7 小结与练习	343
15.7.1 小结	343
15.7.2 习题与思考	343
第 16 章 数据库编程	345
16.1 数据库基础	346
16.2 SQL 语句	347
16.2.1 关系型数据库简介	347
16.2.2 数据类型	348
16.2.3 创建和删除表	349
16.2.4 插入数据	350
16.2.5 数据查询	351
16.2.6 修改数据	355

16.3 使用 DBI	356
16.3.1 DBI 与 DBD	356
16.3.2 DBI 编程环境的搭建与测试	357
16.3.3 句柄	360
16.3.4 连接和断开连接	361
16.3.5 数据库操作	364
16.4 DBM 数据库	376
16.4.1 创建和打开 DBM 文件	377
16.4.2 插入和删除记录	378
16.4.3 遍历 DBM 数据库	379
16.5 小结与练习	381
16.5.1 小结	381
16.5.2 习题与思考	381
第 17 章 网络编程	383
17.1 Socket 基础	384
17.1.1 Socket 原理	384
17.1.2 客户端 / 服务器模型	385
17.2 Perl 对 Socket 的支持	388
17.2.1 和协议相关的函数	388
17.2.2 Socket 底层函数	394
17.3 面向连接模型应用	399
17.3.1 服务器端的程序	400
17.3.2 客户端的程序	403
17.4 使用预编译模块	404
17.4.1 使用 Net::FTP 模块	405
17.4.2 使用 Net::POP3 模块	405
17.5 RPC 的实现	406
17.5.1 Msg 模块实现	406
17.5.2 Msg 模块使用举例	417
17.5.3 RPC 模块的实现	418
17.5.4 RPC 模块使用举例	421
17.6 小结与练习	422
17.6.1 小结	422
17.6.2 习题与思考	422
附录 部分习题参考答案	423

第1章

Linux 及 Perl 概述



Linux 系统简介

Perl 介绍

小结与练习



从 Linux 的诞生到现在，经历了这么多年时间，我们对 Linux 这个名词听起来已经不再陌生了，有许许多多的人在学习和使用 Linux 操作系统，或在 Linux 系统上进行应用程序的开发。

Linux 是一个相当优秀的操作系统资源，这主要是因为 Linux 操作系统本身是免费的，而且用户可以获得它的源代码。此外，Linux 下的许多开发工具也是免费的，这些工具的源代码都是公开的，这意味着用户可以通过研究现有的代码来提高自己的编程水平。众所周知，学习程序设计最好的方法就是看专家写的程序，然后在上面修修改改（当然绝大多数的修改是为了自己对程序理解的更加深刻，而不是修改程序中的“bug”）然后调试，这样可以很快地提高自身的程序设计水平。

Perl 是一门很好的编程语言，它的功能非常强大，而且在许多操作系统上都有它的编译程序，Linux 操作系统也不例外。本书所要研究的就是在 Linux 操作系统上 Perl 语言的编程。

本章要点：

- Linux 系统简介。
- Perl 介绍。

1.1 Linux 系统简介

1.1.1 什么是 Linux

Linux 是一个稳定的、功能强大的而且免费的操作系统。

Linux 是 Internet 的产物，最初它属于一个学生（Linus Torvalds）的个人爱好，而后来却演变成为一个最流行的免费操作系统。对许多人而言，Linux 似乎是个谜，一个免费的东西怎么会有价值？在一个被一群软件巨头统治的（系统）软件王国里，一个由一些电脑黑客编写的操作系统如何能够参与竞争？一个由不同的国家不同的人编写的软件如何能够保持其稳定性和高效性？其答案是肯定的。Linux 具有非常好的可靠性、高效性和竞争能力，许多大学和研究机构都在用 Linux 来作计算；许多人已在其个人计算机上安装了 Linux；绝大多数公司都在或多或少地使用 Linux。Linux 被广泛地用来浏览 Web 站点，文件处理，发送 E-mail，玩计算机游戏等等。Linux 绝不是一个计算机界的玩具，而是一个由全世界的爱好者共同开发的非常完善的、专业化的操作系统。

1.1.2 Linux 的发展历史

Linux 的源头可以追溯到 UNIX 家族。1969 年，贝尔实验室的研究人员 Ken Thompson 开始在一台空闲的 PDP-7 机器上实验其多用户、多任务的操作系统（multi-user, multi-tasking operating system），不久 Dennis Richie 和其他两位同事加入了他的行列，他们与实验室中的其他同事一道开发出了最早期的 UNIX 版本。早期的 UNIX 是用汇编语言编写的，第 3 版时采

用了 C 语言。C 语言是 Richie 设计并编写的，以用来作为编写操作系统的语言。用 C 语言改写过的 UNIX 便可以移植到 PDP-11/45 和 DIGITAL 11/70 计算机上，这是 UNIX 操作系统的一个历史性的转折，使得 UNIX 正式从实验室走向大型机计算环境。很快，绝大多数的计算机制造商都发布了其相应的 UNIX 版本。

Linux 诞生的原因极其简单。Linus Torvalds 是 Linux 的作者和主要管理者，当时仅仅是一名学生，他当时使用的操作系统是 Minix。Minix 是一个非常简单、UNIX 风格、被广泛应用在教学上的操作系统。Linus 对 Minix 的功能不是很满意，决定自己动手编写一个软件。因为在学校里每天使用的都是 UNIX，所以他选择将 UNIX 作为软件的模型。最开始的工作是在一台 Intel 386 的 PC 机上，他的进展很迅速。Linus 对他所做的事情充满了兴趣，并通过刚刚出现的、还局限在学术领域的计算机网络，将已有的代码共享给其他的学生。许多人看见了 Linus 的软件很有趣并开始加入到开发的行列。不同的人由于在使用 Linux 时碰到不同的问题，所以这个软件也就不不断地被更新和完善。不久之后，Linux 就成为一个完整的操作系统。值得注意的是，Linux 中没有任何 UNIX 代码，而是根据 POSIX 标准重新编写的，Linux 中使用了许多在 Cambridge、Massachusetts 的 Free Software Foundation 提供的 GNU 软件。

多数人仅把 Linux 当做一个简单的工具来使用，也有许多用户在 Linux 上进行应用程序的开发，只有极少数人敢于为 Linux 编写设备驱动程序和系统核心的补丁 (patches)。Linus Torvalds 接受来自任何人和任何地方的关于系统核心的补充和修改，但是 Linus 严把质量关并由他自己将新的代码加入核心。当然，在任何时候，从事 Linux 核心开发的人员毕竟是只有一少部分人。

大多数 Linux 用户似乎不关心这个操作系统是如何构造和运行的。这不是个明智的决定，因为学习 Linux 是更好地理解操作系统功能的有利途径。Linux 不仅设计得很好，更重要的是其源代码是公开的。这是因为虽然作者拥有这个软件的版权，但在 Free Software Foundation's GNU Public License 的基础上，源代码是可以免费获取的。

对刚接触源代码的人，当他们看见一些叫做 kernel、mm 和 net 的子目录时，可能会觉得迷惑，它们含有什么？这些代码是如何工作的？为了解决上述问题，需要做的是对整个 Linux 的结构有个总体的了解。



源代码公开有什么好处？举个例子，假如用户对 foxmail 之类的邮件收发程序感兴趣，可是又不知道它是如何实现，这个时候，一个很好的办法就是去找 Linux 下的自由软件，查看它的源代码，仔细研究它，最后肯定会知道这个软件是如何实现的。

1.1.3 Linux 主要发行版本

与 Microsoft Windows 98 或 NT 不同，Linux 操作系统并不是一个单一的产品，在市场上有多种发行版本，它们并不都是一样的。

所有的发行版本具有同样的 Linux 内核，内核包含了所有核心的操作系统功能以及网络堆栈。另外，他们都提供有标准的工具、一系列的应用程序、一些打印的文档以及有限的技术支持。每种发行版本都带有 X Window 系统以及一个图形用户界面、Web Server、e-mail