

资深程序员十年经验总结，深入探讨Linux应用层和内核层的网络编程  
详细讲解HTTP服务器、协议栈和防火墙三个典型案例的实际开发过程



# Linux

宋敬彬  
孙海滨 等编著

# 网络编程

- ◎ 内容全面：全书涵盖Linux网络编程从基础到高级开发的方方面面
- ◎ 内容深入：重点讲解了技术性较强的Linux用户空间网络编程及内核网络编程
- ◎ 注重原理：对每个知识点都从原始概念和基本原理进行详细、透彻地分析
- ◎ 插图丰富：对比较复杂和难度较高的内容绘制了220余幅原理图进行讲解
- ◎ 代码经典：书中的示例代码大多是从实际项目总结而来，有很强的实用性
- ◎ 实践性强：贯穿450余个示例、70余个实例及3个案例进行讲解
- ◎ 案例典型：详细介绍了HTTP网络服务器、协议栈和防火墙的实现

Linux 典藏大系

ChinaUnix<sup>net</sup>



# Linux

宋敬彬  
孙海滨 等编著

# 网络编程

清华大学出版社

北京

## 内 容 简 介

Linux 是目前最流行的开源操作系统,网络功能在 Linux 下占有核心的地位。本书循序渐进地从应用层到 Linux 内核、从基本知识点到综合案例,向读者介绍如何在 Linux 下进行网络程序设计。本书内容分为 4 个部分:Linux 程序设计基础部分、Linux 用户空间网络编程部分、Linux 内核网络编程部分以及综合案例部分。内容包含 Linux 系统概述、Linux 编程环境、Linux 文件系统简介、Linux 下的进程和线程、TCP/IP 协议族、应用层网络服务程序、TCP 编程、主机信息获取、数据 IO 复用、UDP 编程、高级套接字、套接字选项、原始套接字、服务器模型、IPv6、Linux 内核网络部分结构及分布、netfilter 框架内报文处理。为了方便读者学习,本书最后一个部分介绍了 3 个综合案例,包括应用层的 Web 服务器例子、简单的应用层网络协议站例子和内核层网防火墙的例子。

本书适合广大的 Linux 平台下的网络程序设计人员和大专院校学生阅读,尤其是有一定 Linux 基础知识的编程技术人员。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

Linux 网络编程 / 宋敬彬, 孙海滨等编著. —北京: 清华大学出版社, 2010.1  
(Linux 典藏大系)

ISBN 978-7-302-20717-7

I. L… II. ①宋… ②孙… III. Linux 操作系统—程序设计 IV. TP316.89

中国版本图书馆 CIP 数据核字(2009)第 139544 号

责任编辑:夏兆彦

责任校对:徐俊伟

责任印制:王秀菊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市世界知识印刷厂

装 订 者:三河市金元印装有限公司

经 销:全国新华书店

开 本:185×260 印 张:44.75 字 数:1113 千字

版 次:2010 年 1 月第 1 版 印 次:2010 年 1 月第 1 次印刷

印 数:1~5000

定 价:79.00 元

---

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:034379-01

# 前言

Linux 操作系统已经成为目前最流行的开源操作系统，在服务器、嵌入式系统有着广泛的应用，并且逐步走入个人电脑的桌面操作系统。Linux 的网络程序设计的服务器领域、嵌入式领域有着广泛的应用。例如 Web 服务器、P2P 应用、嵌入式网络机顶盒、IPTV 机顶盒、手持设备等，上述产品大部分采用了开源的 Linux 系统。因此，熟悉并且能够编写网络程序代码，构建自己的网络架构程序是十分重要的。

本书全面地介绍了 Linux 的网络编程技术，并以实例介绍了 Linux 的应用层网络设计、网络协议栈的实现原理和 Linux 内核防火墙的技术。学完本书之后，读者可以有编写比较复杂项目的本领。

## 本书的特点

### 1. 循序渐进，由浅入深

为了方便读者学习，本书首先介绍 Linux 的开发环境，然后介绍基本的网络程序设计方法，再进行 Linux 内核的网络设计方法。最后，通过 3 个综合案例，综合运用上述知识，让读者更深刻地了解网络程序设计的知识。在每一部分的介绍中都是按照由浅入深的方式进行介绍，先介绍基础知识，再结合高级知识进行介绍。

### 2. 技术全面，内容充实

本书基本涵盖了 Linux 网络程序设计的所有知识面，特别对于高级网络编程、原始套接字等高级应用层网络程序设计给出了全面的介绍和丰富的例子程序。除了用户界面的网络程序设计外，本书还对内核空间的网络程序设计进行了详细的介绍，针对 netfilter 框架，做了很细致的讲解，并给出了一个全面使用 netfilter 框架的案例方便读者深入了解。

### 3. 对比讲解，理解深刻

由于 Linux 程序设计的知识用于空间和内核空间的代码和模块是相互作用的，在多个主要函数介绍过程中，本书对用户空间和内核空间进行交互式的对比介绍，使读者在了解如何使用的情况下，更深入地了解为什么这样用，“知其然并知其所以然”。

### 4. 案例精讲，深入剖析

根据本人多年的项目经验，只有实际接触案例和代码才能够对知识点更深入地了解。本书在介绍了 Linux 网络程序设计知识点的基础上，通过具有典型意义的 3 个案例，对各

个知识点包括应用层的 HTTP 协议的 Web 服务器、协议栈原理的协议栈案例和内核网络的防火墙案例进行了深入剖析。

## 本书的内容

第 1 章：为了方便读者对 Linux 历史文化的了解，这一章从 Linux 的发展历史、发展要素、发行版本和内核的选择、架构、GPL 许可证等几个方面向读者介绍 Linux 操作系统。

第 2 章：比较详细地介绍了在 Linux 环境下进行软件开发所必须的几个方面，包括 Linux 环境下的编辑器、GCC 编译器、Makefile 的编写、如何使用 GDB 进行程序调试等。

第 3 章：介绍了 Linux 下的文件系统的概念以及如何使用文件相关函数进行文件操作，通过本章读者可以掌握文件编程。

第 4 章：对 Linux 下进程、线程以及之间的通信方法进行了详细的介绍。内容包含进程和线程的概念、进程的产生方式、进程间的 IPC 通信、Linux 下的线程及通信方法。

第 5 章：重点介绍了 TCP/IP 的模型和多个主要的协议类型，并对 IP 地址的概念进行了介绍，最后简单介绍了字节序的概念。

第 6 章：介绍了用户空间的网络协议，包含 HTTP 协议、FTP 协议、NFS 协议和 TELNET 协议，最后介绍了如何利用 xinetd 定义自己的网络服务。

第 7 章：重点介绍了 TCP 网络程序设计的基础知识，包括套接字地址结构、TCP 网络程序流程。对套接字编程的主要函数进行了介绍，并提供了一个简单的例子。

第 8 章：重点介绍了主机信息的获取，包含网络字节需转换、IP 地址转换函数、套接字类型判定、协议名称处理、DNS 获取函数等。

第 9 章：介绍了 Linux 下的 IO 复用，除了常用的 Linux 下的 IO 函数，重点介绍了阻塞 IO 模型、非阻塞 IO 模型、IO 复用、信号驱动 IO 模型和异步 IO 模型等 IO 模型。

第 10 章：介绍了 UDP 协议的编程框架并给出了多个例子。对 UDP 编程中经常遇到的问题进行了介绍，并给出了问题解决的方案。

第 11 章：介绍了网络编程中的高级套接字，包含 UNIX 域协议编程、广播、多播和数据链路层访问。本章给出了详细的例子代码介绍上述的概念。

第 12 章：介绍套接字选项，对套接字选项的含义进行了详细的介绍，还介绍了 ioctl() 控制选项和 fcntl() 控制选项。

第 13 章：介绍了原始套接字，包含原始套接字的创建、接收数据、发送数据等，对经常使用原始套接字进行程序设计的洪水攻击进行了比较详细的介绍。

第 14 章：介绍服务器的模型，对循环服务器、简单并发服务器、TCP 高级并发服务器、IO 复用循环服务器进行了详细介绍。

第 15 章：简单介绍了 IPv6 协议，对 IPv4 和 IPv6 两种协议的不同点进行讲解，最后介绍了如何使用 IPv6 的接口进行程序设计。

第 16 章：介绍了 Linux 内核中的网络部分代码，以及网络在内核中的分布、软中断、网络数据在内核中的流程。

第 17 章：介绍了 Linux 内置防火墙 netfilter 的报文处理，并介绍了其中的 5 个钩子和内核模块程序设计框架。

第 18 章：介绍了一个 Web 服务器 SHTTPD 的综合案例。从需求、设计、实现等方面

介绍了 Web 服务器实现的主要框架和实现的技术难点。

第 19 章：介绍了一个应用层网络协议栈 SIP 的综合案例，以及网络协议栈的分层结构实现，协议栈包含了物理层、MAC 层、IP 层、ARP 层、UDP 层等方面。

第 20 章：介绍了一个内核防火墙 SIPFW 的综合案例。在 netfilter 框架的基础上，实现了一个可以进行用户交互的、可进行信息统计的、网络数据报文过滤型防火墙。

## 适合的读者

- 包括 Linux 网络开发工作人员；
- Linux 网络开发爱好者；
- 大中专院校的学生；
- 社会培训学生。

## 本书作者及编委会成员

本书由宋敬彬、孙海滨主笔编写。其他参与编写和资料整理的人员有武冬、郅晓娜、孙美芹、卫丽行、尹翠翠、蔡继文、陈晓宇、迟剑、邓薇、郭利魁、金贞姬、李敬才、李萍、刘敬、陈慧、刘艳飞、吕博、全哲、余勇、宋学江、王浩、王康、王楠、杨宗芳、张严虎、周玉、张平、张靖波、周芳、杨罡、于海滨、张晶杰、张利峰、杨景凤、陈锴、郑剑锋、叶佩思、张涛、赵东彪、王双。在此一并表示感谢！

本书编委会成员有欧振旭、陈杰、陈冠军、项宇峰、张帆、陈刚、程彩红、毛红娟、聂庆亮、王志娟、武文娟、颜盟盟、姚志娟、尹继平、张昆、张薛。

虽然我们对书中所述的内容都尽量予以核实，并多次进行文字校对，但因时间所限，可能还存在疏漏和不足之处，恳请读者批评指正。

编著者

# 目 录

## 第 1 篇 Linux 网络开发基础

第 1 章 Linux 操作系统概述	2
1.1 Linux 发展历史	2
1.1.1 Linux 的诞生和发展	2
1.1.2 Linux 名称的由来	3
1.2 Linux 的发展要素	3
1.2.1 UNIX 操作系统	4
1.2.2 Minix 操作系统	4
1.2.3 POSIX 标准	4
1.3 Linux 与 UNIX 的异同	5
1.4 操作系统类型选择和内核版本的选择	5
1.4.1 常见的不同公司发行的 Linux 异同	6
1.4.2 内核版本的选择	6
1.5 Linux 的系统架构	7
1.5.1 Linux 内核的主要模块	7
1.5.2 Linux 的文件结构	9
1.6 GNU 通用公共许可证	10
1.6.1 GPL 许可证的历史	10
1.6.2 GPL 的自由理念	10
1.6.3 GPL 的基本条款	11
1.6.4 关于 GPL 许可证的争议	12
1.7 Linux 软件开发的可借鉴之处	12
1.8 小结	13
第 2 章 Linux 编程环境	14
2.1 Linux 环境下的编辑器	14
2.1.1 vim 使用简介	14
2.1.2 使用 vim 建立文件	15
2.1.3 使用 vim 编辑文本	16
2.1.4 vim 的格式设置	18

2.1.5	vim 配置文件.vimrc	19
2.1.6	使用其他编辑器	19
2.2	Linux 下的 GCC 编译器工具集	19
2.2.1	GCC 简介	19
2.2.2	编译程序的基本知识	21
2.2.3	单个文件编译成执行文件	22
2.2.4	编译生成目标文件	22
2.2.5	多文件编译	23
2.2.6	预处理	24
2.2.7	编译成汇编语言	24
2.2.8	生成和使用静态链接库	25
2.2.9	生成动态链接库	26
2.2.10	动态加载库	29
2.2.11	GCC 常用选项	31
2.2.12	编译环境的搭建	33
2.3	Makefile 文件简介	34
2.3.1	一个多文件的工程例子	34
2.3.2	多文件工程的编译	36
2.3.3	Makefile 的规则	37
2.3.4	Makefile 中使用变量	39
2.3.5	搜索路径	43
2.3.6	自动推导规则	44
2.3.7	递归 make	44
2.3.8	Makefile 中的函数	46
2.4	用 GDB 调试程序	47
2.4.1	编译可调试程序	48
2.4.2	使用 GDB 调试程序	49
2.4.3	GDB 常用命令	52
2.4.4	其他的 GDB	59
2.5	小结	60
第 3 章	文件系统简介	61
3.1	Linux 下的文件系统	61
3.1.1	Linux 下文件的内涵	61
3.1.2	文件系统的创建	62
3.1.3	挂接文件系统	64
3.1.4	索引节点 inode	65
3.1.5	普通文件	66
3.1.6	设备文件	66
3.1.7	虚拟文件系统 VFS	68



3.2	文件的通用操作方法	72
3.2.1	文件描述符	72
3.2.2	打开创建文件 <code>open()</code> 、 <code>create()</code> 函数	72
3.2.3	关闭文件 <code>close()</code> 函数	76
3.2.4	读取文件 <code>read()</code> 函数	77
3.2.5	写文件 <code>write()</code> 函数	79
3.2.6	文件偏移 <code>lseek()</code> 函数	80
3.2.7	获得文件状态 <code>fstat()</code> 函数	83
3.2.8	文件空间映射 <code>mmap()</code> 函数	85
3.2.9	文件属性 <code>fcntl()</code> 函数	88
3.2.10	文件输入输出控制 <code>ioctl()</code> 函数	92
3.3	socket 文件类型	93
3.4	小结	93
<b>第 4 章</b>	<b>程序、进程和线程</b>	<b>94</b>
4.1	程序、进程和线程的概念	94
4.1.1	程序和进程的差别	94
4.1.2	Linux 环境下的进程	95
4.1.3	进程和线程	96
4.2	进程产生的方式	96
4.2.1	进程号	96
4.2.2	进程复制 <code>fork()</code>	97
4.2.3	<code>system()</code> 方式	98
4.2.4	进程执行 <code>exec()</code> 函数系列	99
4.2.5	所有用户态进程的生成进程 <code>init</code>	100
4.3	进程间通信和同步	101
4.3.1	半双工管道	101
4.3.2	命名管道	107
4.3.3	消息队列	108
4.3.4	消息队列的一个例子	114
4.3.5	信号量	116
4.3.6	共享内存	121
4.3.7	信号	124
4.4	Linux 下的线程	127
4.4.1	多线程编程实例	127
4.4.2	Linux 下线程创建函数 <code>pthread_create()</code>	129
4.4.3	线程的结束函数 <code>pthread_join()</code> 和 <code>pthread_exit()</code>	129
4.4.4	线程的属性	130
4.4.5	线程间的互斥	132
4.4.6	线程中使用信号量	133

4.5 小结	136
--------	-----

## 第 2 篇 Linux 用户层网络编程

第 5 章 TCP/IP 协议族简介	138
5.1 OSI 网络分层介绍	138
5.1.1 OSI 网络分层结构	138
5.1.2 OSI 的 7 层网络结构	139
5.1.3 OSI 参考模型中的数据传输	140
5.2 TCP/IP 协议栈	141
5.2.1 TCP/IP 协议栈参考模型	141
5.2.2 主机到网络层协议	143
5.2.3 IP 协议	144
5.2.4 网际控制报文协议 (ICMP)	146
5.2.5 传输控制协议 (TCP)	150
5.2.6 用户数据报文协议 (UDP)	154
5.2.7 地址解析协议 (ARP)	156
5.3 IP 地址分类与 TCP/UDP 端口	158
5.3.1 因特网中 IP 地址的分类	159
5.3.2 子网掩码 (subnet mask address)	161
5.3.3 IP 地址的配置	162
5.3.4 端口	163
5.4 主机字节序和网络字节序	163
5.4.1 字节序的含义	164
5.4.2 网络字节序的转换	164
5.5 小结	166
第 6 章 应用层网络服务程序简介	167
6.1 HTTP 协议和服务	167
6.1.1 HTTP 协议概述	167
6.1.2 HTTP 协议的基本过程	168
6.2 FTP 协议和服务	170
6.2.1 FTP 协议概述	170
6.2.2 FTP 协议的工作模式	172
6.2.3 FTP 协议的传输方式	172
6.2.4 一个简单的 FTP 过程	173
6.2.5 常用的 FTP 工具	173
6.3 TELNET 协议和服务	174
6.3.1 远程登录的基本概念	174

6.3.2	使用 TELNET 协议进行远程登录的工作过程	174
6.3.3	TELNET 协议	174
6.4	NFS 协议和服务	176
6.4.1	安装 NFS 服务器和客户端	176
6.4.2	服务器端的设定	176
6.4.3	客户端的操作	177
6.4.4	showmount 命令	177
6.5	自定义网络服务	177
6.5.1	xinetd/inetd	178
6.5.2	xinetd 服务配置	178
6.5.3	自定义网络服务	179
6.6	小结	180
<b>第 7 章 TCP 网络编程基础</b>		<b>181</b>
7.1	套接字编程基础知识	181
7.1.1	套接字地址结构	181
7.1.2	用户层和内核层交互过程	183
7.2	TCP 网络编程流程	184
7.2.1	TCP 网络编程架构	184
7.2.2	创建网络插口函数 socket()	186
7.2.3	绑定一个地址端口对 bind()	189
7.2.4	监听本地端口 listen	192
7.2.5	接受一个网络请求 accept()	194
7.2.6	连接目标网络服务器 connect()	199
7.2.7	写入数据函数 write()	200
7.2.8	读取数据函数 read()	201
7.2.9	关闭套接字函数 close()	201
7.3	服务器/客户端的简单例子	202
7.3.1	例子功能描述	202
7.3.2	服务器网络程序	203
7.3.3	服务器读取和显示字符串	205
7.3.4	客户端的网络程序	205
7.3.5	客户端读取和显示字符串	206
7.3.6	编译运行程序	206
7.4	截取信号的例子	207
7.4.1	信号处理	207
7.4.2	信号 SIGPIPE	208
7.4.3	信号 SIGINT	208
7.5	小结	208

第 8 章 服务器和客户端信息的获取 .....	210
8.1 字节序 .....	210
8.1.1 大端字节序和小端字节序 .....	210
8.1.2 字节序转换函数 .....	212
8.1.3 一个字节序转换的例子 .....	214
8.2 字符串 IP 地址和二进制 IP 地址的转换 .....	217
8.2.1 inet_xxx() 函数 .....	217
8.2.2 inet_pton() 和 inet_ntop() 函数 .....	219
8.2.3 使用 8.2.1 节地址转换函数的例子 .....	220
8.2.4 使用函数 inet_pton() 和函数 inet_ntop() 的例子 .....	223
8.3 套接字描述符判定函数 isssockettype() .....	223
8.3.1 进行文件描述符判定的函数 isssockettype() .....	224
8.3.2 main() 函数 .....	224
8.4 IP 地址与域名之间的相互转换 .....	225
8.4.1 DNS 原理 .....	225
8.4.2 获取主机信息的函数 .....	226
8.4.3 使用主机名获取主机信息的例子 .....	228
8.4.4 函数 gethostbyname() 不可重入的例子 .....	230
8.5 协议名称处理函数 .....	232
8.5.1 xxxprotoxxx() 函数 .....	232
8.5.2 使用协议族函数的例子 .....	233
8.6 小结 .....	236
第 9 章 数据的 IO 和复用 .....	237
9.1 IO 函数 .....	237
9.1.1 使用 recv() 函数接收数据 .....	237
9.1.2 使用 send() 函数发送数据 .....	239
9.1.3 使用 readv() 函数接收数据 .....	240
9.1.4 使用 writev() 函数发送数据 .....	240
9.1.5 使用 recvmsg() 函数接收数据 .....	242
9.1.6 使用 sendmsg() 函数发送数据 .....	244
9.1.7 IO 函数的比较 .....	246
9.2 使用 IO 函数的例子 .....	246
9.2.1 客户端处理框架的例子 .....	246
9.2.2 服务器端程序框架 .....	248
9.2.3 使用 recv() 和 send() 函数 .....	249
9.2.4 使用 readv() 和 writev() 函数 .....	251
9.2.5 使用 recvmsg() 和 sendmsg() 函数 .....	253
9.3 IO 模型 .....	256
9.3.1 阻塞 IO 模型 .....	256

9.3.2	非阻塞 IO 模型	257
9.3.3	IO 复用	257
9.3.4	信号驱动 IO 模型	258
9.3.5	异步 IO 模型	258
9.4	select()函数和 pselect()函数	259
9.4.1	select()函数	259
9.4.2	pselect()函数	261
9.5	poll()函数和 ppoll()函数	262
9.5.1	poll()函数	263
9.5.2	ppoll()函数	264
9.6	非阻塞编程	264
9.6.1	非阻塞方式程序设计介绍	264
9.6.2	非阻塞程序设计的例子	264
9.7	小结	266
<b>第 10 章</b>	<b>基于 UDP 协议的接收和发送</b>	<b>267</b>
10.1	UDP 编程框架	267
10.1.1	UDP 编程框图	267
10.1.2	UDP 服务器编程框架	269
10.1.3	UDP 客户端编程框架	269
10.2	UDP 协议程序设计的常用函数	270
10.2.1	建立套接字 socket()和绑定套接字 bind()	270
10.2.2	接收数据 recvfrom()/recv()	270
10.2.3	发送数据 sendto()/send()	275
10.3	UDP 接收和发送数据的例子	279
10.3.1	UDP 服务器端	279
10.3.2	UDP 服务器端数据处理	280
10.3.3	UDP 客户端	281
10.3.4	UDP 客户端数据处理	281
10.3.5	测试 UDP 程序	282
10.4	UDP 协议程序设计中的几个问题	282
10.4.1	UDP 报文丢失数据	282
10.4.2	UDP 数据发送中的乱序	284
10.4.3	UDP 协议中的 connect()函数	287
10.4.4	UDP 缺乏流量控制	287
10.4.5	UDP 协议中的外出网络接口	289
10.4.6	UDP 协议中的数据报文截断	290
10.5	小结	291
<b>第 11 章</b>	<b>高级套接字</b>	<b>292</b>
11.1	UNIX 域函数	292

11.1.1	UNIX 域函数的地址结构	292
11.1.2	套接字函数	293
11.1.3	使用 UNIX 域函数进行套接字编程	293
11.1.4	传递文件描述符	296
11.1.5	socketpair()函数	296
11.1.6	传递文件描述符的例子	297
11.2	广播	302
11.2.1	广播的 IP 地址	302
11.2.2	广播与单播的比较	303
11.2.3	广播的示例	304
11.3	多播	310
11.3.1	多播的概念	310
11.3.2	广域网的多播	311
11.3.3	多播的编程	311
11.3.4	内核中的多播	313
11.3.5	一个多播例子的服务器端	317
11.3.6	一个多播例子的客户端	318
11.4	数据链路层访问	319
11.4.1	SOCK_PACKET 类型	319
11.4.2	设置套接口以捕获链路帧的编程方法	320
11.4.3	从套接口读取链路帧的编程方法	321
11.4.4	定位 IP 包头的编程方法	322
11.4.5	定位 TCP 报头的编程方法	323
11.4.6	定位 UDP 报头的编程方法	325
11.4.7	定位应用层报文数据的编程方法	326
11.4.8	使用 SOCK_PACKET 编写 ARP 请求程序的例子	326
11.5	小结	329
<b>第 12 章</b>	<b>套接字选项</b>	<b>330</b>
12.1	获取和设置套接字选项 getsockopt()/setsockopt()	330
12.1.1	getsockopt()函数和 setsockopt()函数的介绍	330
12.1.2	套接字选项	331
12.1.3	套接字选项简单示例	332
12.2	SOL_SOCKET 协议族选项	336
12.2.1	SO_BROADCAST 广播选项	336
12.2.2	SO_DEBUG 调试选项	337
12.2.3	SO_DONTROUTE 不经过路由选项	337
12.2.4	SO_ERROR 错误选项	338
12.2.5	SO_KEEPALIVE 保持连接选项	338
12.2.6	SO_LINGER 缓冲区处理方式选项	339

12.2.7	SO_OOBINLINE 带外数据处理方式选项	342
12.2.8	SO_RCVBUF 和 SO_SNDBUF 缓冲区大小选项	342
12.2.9	SO_RCVLOWAT 和 SO_SNDLOWAT 缓冲区下限选项	343
12.2.10	SO_RCVTIMEO 和 SO_SNDTIMEO 收发超时选项	343
12.2.11	SO_REUSEADDR 地址重用选项	344
12.2.12	SO_EXCLUSIVEADDRUSE 端口独占选项	344
12.2.13	SO_TYPE 套接字类型选项	345
12.2.14	SO_BSDCOMPAT 与 BSD 套接字兼容选项	345
12.2.15	SO_BINDTODEVICE 套接字网络接口绑定选项	345
12.2.16	SO_PRIORITY 套接字优先级选项	346
12.3	IPPROTO_IP 选项	347
12.3.1	IP_HDRINCL 选项	347
12.3.2	IP_OPTNIO 选项	347
12.3.3	IP_TOS 选项	347
12.3.4	IP_TTL 选项	347
12.4	IPPROTO_TCP 选项	348
12.4.1	TCP_KEEPLIVE 选项	348
12.4.2	TCP_MAXRT 选项	348
12.4.3	TCP_MAXSEG 选项	349
12.4.4	TCP_NODELAY 和 TCP_CORK 选项	349
12.5	使用套接字选项	351
12.5.1	设置和获取缓冲区大小	351
12.5.2	获取套接字类型的例子	355
12.5.3	使用套接字选项的综合例子	356
12.6	ioctl()函数	361
12.6.1	ioctl()函数的命令选项	361
12.6.2	ioctl()函数的 IO 请求	363
12.6.3	ioctl()函数的文件请求	365
12.6.4	ioctl()函数的网络接口请求	365
12.6.5	使用 ioctl()函数对 ARP 高速缓存操作	372
12.6.6	使用 ioctl()函数发送路由表请求	374
12.7	fcntl()函数	374
12.7.1	fcntl()函数的选项	375
12.7.2	使用 fcntl()函数修改套接字非阻塞属性	375
12.7.3	使用 fcntl()函数设置信号属主	376
12.8	小结	376
<b>第 13 章</b>	<b>原始套接字</b>	<b>377</b>
13.1	概述	377
13.2	原始套接字的创建	379

13.2.1	SOCK_RAW 选项	379
13.2.2	IP_HDRINCL 套接字选项	379
13.2.3	不需要 bind() 函数	380
13.3	原始套接字发送报文	380
13.4	原始套接字接收报文	380
13.5	原始套接字报文处理时的结构	381
13.5.1	IP 头部的结构	381
13.5.2	ICMP 头部结构	382
13.5.3	UDP 头部结构	384
13.5.4	TCP 头部结构	386
13.6	ping 的例子	387
13.6.1	协议格式	388
13.6.2	校验和函数	389
13.6.3	设置 ICMP 发送报文的头部	390
13.6.4	剥离 ICMP 接受报文的头部	391
13.6.5	计算时间差	392
13.6.6	发送报文	393
13.6.7	接收报文	394
13.6.8	主函数过程	395
13.6.9	主函数 main()	397
13.6.10	编译测试	400
13.7	洪水攻击	400
13.8	ICMP 洪水攻击	401
13.8.1	ICMP 洪水攻击的原理	401
13.8.2	ICMP 洪水攻击的例子	401
13.9	UDP 洪水攻击	405
13.10	SYN 洪水攻击	409
13.10.1	SYN 洪水攻击的原理	409
13.10.2	SYN 洪水攻击的例子	409
13.11	小结	413
<b>第 14 章</b>	<b>服务器模型选择</b>	<b>414</b>
14.1	循环服务器	414
14.1.1	UDP 循环服务器	414
14.1.2	TCP 循环服务器	417
14.2	简单并发服务器	420
14.2.1	并发服务器的模型	420
14.2.2	UDP 并发服务器	420
14.2.3	TCP 并发服务器	423
14.3	TCP 的高级并发服务器模型	426



14.3.1	单客户端单进程, 统一 <code>accept()</code> .....	426
14.3.2	单客户端单线程, 统一 <code>accept()</code> .....	429
14.3.3	单客户端单线程, 各线程独自 <code>accept()</code> , 使用互斥锁.....	431
14.4	IO 复用循环服务器.....	435
14.4.1	IO 复用循环服务器模型介绍.....	435
14.4.2	IO 复用循环服务器模型的例子.....	436
14.5	小结.....	440
第 15 章	IPv6 简介.....	441
15.1	IPv4 的缺陷.....	441
15.2	IPv6 的特点.....	442
15.3	IPv6 的地址.....	443
15.3.1	IPv6 的单播地址.....	443
15.3.2	可聚集全球单播地址.....	443
15.3.3	本地使用单播地址.....	444
15.3.4	兼容性地址.....	445
15.3.5	IPv6 多播地址.....	446
15.3.6	IPv6 任播地址.....	446
15.3.7	主机的多个 IPv6 地址.....	447
15.4	IPv6 的头部.....	447
15.4.1	IPv6 头部格式.....	447
15.4.2	与 IPv4 头部的对比.....	448
15.4.3	IPv6 的 TCP 头部.....	449
15.4.4	IPv6 的 UDP 头部.....	449
15.4.5	IPv6 的 ICMP 头部.....	449
15.5	IPv6 运行环境.....	451
15.5.1	加载 IPv6 模块.....	451
15.5.2	查看是否支持 IPv6.....	452
15.6	IPv6 的结构定义.....	453
15.6.1	IPv6 的地址族和协议族.....	453
15.6.2	套接字地址结构.....	453
15.6.3	地址兼容考虑.....	455
15.6.4	IPv6 通用地址.....	455
15.7	IPv6 的套接字函数.....	456
15.7.1	<code>socket()</code> 函数.....	456
15.7.2	没有发生改变的函数.....	456
15.7.3	发生改变的函数.....	457
15.8	IPv6 的套接字选项.....	457
15.8.1	IPv6 的套接字选项.....	457
15.8.2	单播跳限 <code>IPV6_UNICAST_HOPS</code> .....	459