

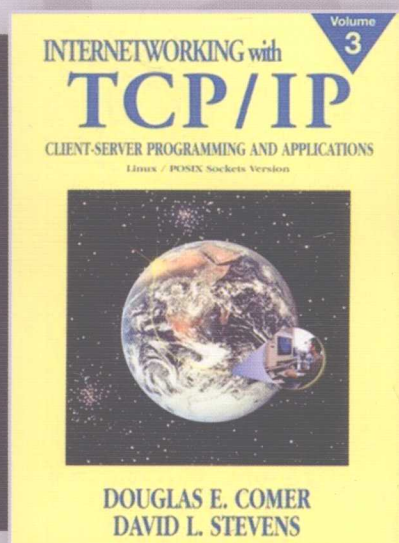
Douglas E. Comer

PEARSON
Prentice
Hall

用TCP/IP 进行网际互连

第三卷

—— 客户-服务器编程与应用
(Linux/POSIX套接字版)



Internetworking

With TCP/IP

Client-Server Programming
and Applications

Linux/POSIX Sockets Version

[美]

Douglas E. Comer
David L. Stevens

著

赵刚 林瑶 蒋慧 等译
谢希仁 审校



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

国外计算机科学教材系列

用 TCP/IP 进行网际互连

第三卷——客户 - 服务器编程与应用 (Linux/POSIX 套接字版)

Internetworking With TCP/IP

Volume III: Client-Server Programming and Applications

Linux/POSIX Sockets Version

[美] Douglas E. Comer 著
David L. Stevens

赵 刚 林 瑶 蒋 慧 等译
谢希仁 审校

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书是关于计算机网络的经典教材,是目前美国大多数大学所开设的计算机网络课程的主要参考书。目前国内外能见到的各种关于TCP/IP的书籍,其主要内容都参考了本书。本书的特点是强调原理,概念准确,深入浅出,内容丰富新颖。全书共分为三卷。第三卷主要讨论应用软件如何使用TCP/IP,重点研究了客户-服务器范例,并考察了分布式程序中的客户和服务,举例说明了各种设计,讨论了应用网关和隧道技术。

Authorized translation from the English language edition, entitled Internetworking With TCP/IP Volume III: Client-Server Programming and Applications, Linux/POSIX Sockets Version, ISBN: 0-13-032071-4 by Douglas E. Comer, David L. Stevens, published by Pearson Education, Inc, publishing as Prentice Hall, Copyright © 2001.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

Simplified Chinese language edition published by Publishing House of Electronics Industry, Copyright © 2008.

This edition is authorized for sale only in the People's Republic of China excluding Hong Kong, Macau and Taiwan.

本书中文简体专有翻译出版版权由Pearson教育集团所属的Prentice Hall授予电子工业出版社。其原文版权及中文翻译出版版权受法律保护。未经许可,不得以任何形式或手段复制或抄袭本书内容。

此版本仅限在中华人民共和国境内(不包括香港、澳门特别行政区以及台湾地区)发行与销售。

版权贸易合同登记号:图字:01-2000-3473

图书在版编目(CIP)数据

用TCP/IP进行网际互连.第3卷,客户-服务器编程与应用:Linux/POSIX套接字版/(美)科默(Comer, D. E.), (美)史蒂文(Stevens, D. L.)著.赵刚等译.-北京:电子工业出版社,2008.10

书名原文: Internetworking With TCP/IP Volume III: Client-Server Programming and Applications, Linux/POSIX Sockets Version

(国外计算机科学教材系列)

ISBN 978-7-121-07386-1

I. 用... II. ①科... ②史... ③赵... III. ①计算机网络-通信协议-教材 ②Linux操作系统-程序设计-教材 IV. TN915.04 TP316.89

中国版本图书馆CIP数据核字(2008)第140676号

责任编辑:谭海平

印刷:北京市天竺颖华印刷厂

装订:三河市金马印装有限公司

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:787×1092 1/16 印张:28.75 字数:718千字

印次:2008年10月第1次印刷

定价:55.00元

凡所购买电子工业出版社的图书有缺损问题,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zlt@phei.com.cn,盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线:(010)88258888。

出版说明

21 世纪初的 5 至 10 年是我国国民经济和社会发展的关键时期,也是信息产业快速发展的关键时期。在我国加入 WTO 后的今天,培养一支适应国际化竞争的一流 IT 人才队伍是我国高等教育的重要任务之一。信息科学和技术方面人才的优劣与多寡,是我国面对国际竞争时成败的关键因素。

当前,正值我国高等教育特别是信息科学领域的教育调整、变革的重大时期,为使我国教育体制与国际化接轨,有条件的高等院校正在为某些信息学科和技术课程使用国外优秀教材和优秀原版教材,以使我国在计算机教学上尽快赶上国际先进水平。

电子工业出版社秉承多年来引进国外优秀图书的经验,翻译出版了“国外计算机科学教材系列”丛书,这套教材覆盖学科范围广、领域宽、层次多,既有本科专业课程教材,也有研究生课程教材,以适应不同院系、不同专业、不同层次的师生对教材的需求,广大师生可自由选择和自由组合使用。这些教材涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。同时,我们也适当引进了一些优秀英文原版教材,本着翻译版本和英文原版并重的原则,对重点图书既提供英文原版又提供相应的翻译版本。

在图书选题上,我们大都选择国外著名出版公司出版的高校教材,如 Pearson Education 培生教育集团、麦格劳-希尔教育集团、麻省理工学院出版社、剑桥大学出版社等。撰写教材的许多作者都是蜚声世界的教授、学者,如道格拉斯·科默(Douglas E. Comer)、威廉·斯托林斯(William Stallings)、哈维·戴特尔(Harvey M. Deitel)、尤利斯·布莱克(Uyless Black)等。

为确保教材的选题质量和翻译质量,我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本系列教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师、博士,也有积累了几十年教学经验的老教授和博士生导师。

在该系列教材的选题、翻译和编辑加工过程中,为提高教材质量,我们做了大量细致的工作,包括对所选教材进行全面论证;选择编辑时力求达到专业对口;对排版、印制质量进行严格把关。对于英文教材中出现的错误,我们通过与作者联络和网上下载勘误表等方式,逐一进行了修订。

此外,我们还将与国外著名出版公司合作,提供一些教材的教学支持资料,希望能为授课老师提供帮助。今后,我们将继续加强与各高校教师的密切联系,为广大师生引进更多的国外优秀教材和参考书,为我国计算机科学教学体系与国际教学体系的接轨做出努力。

电子工业出版社

教材出版委员会

主 任 杨芙清 北京大学教授
中国科学院院士
北京大学信息与工程学部主任
北京大学软件工程研究所所长

委 员 王 珊 中国人民大学信息学院院长、教授
胡道元 清华大学计算机科学与技术系教授
国际信息处理联合会通信系统中国代表

钟玉琢 清华大学计算机科学与技术系教授、博士生导师
清华大学深圳研究生院信息学部主任

谢希仁 中国人民解放军理工大学教授
全军网络技术研究中心主任、博士生导师

尤晋元 上海交通大学计算机科学与工程系教授
上海分布计算技术中心主任

施伯乐 上海国际数据库研究中心主任、复旦大学教授
中国计算机学会常务理事、上海市计算机学会理事长

邹 鹏 国防科学技术大学计算机学院教授、博士生导师
教育部计算机基础课程教学指导委员会副主任委员

张昆藏 青岛大学信息工程学院教授

关于作者

Douglas Comer 博士是 TCP/IP 协议和因特网的国际公认专家。自 20 世纪 70 年代末、80 年代初形成因特网以来,他就一直致力于因特网的研究工作,他也是负责指导因特网开发的因特网体系小组 (IAB) 的成员,还是 CSNET 技术委员会的主席和 CSNET 执行委员会的成员。

Comer 为一些公司提供网络设计和实现的咨询,还给全世界的技术和非技术人员开 TCP/IP 和互连网络的专业讲座。他的操作系统 Xinu 以及 TCP/IP 协议的实现在他的书中都有介绍,并且应用到了商业产品中。

Comer 是 Purdue 大学计算机科学系的教授,他主要教授计算机网络、互连网络和操作系统的课程,并进行相关的研究。除了撰写一系列畅销的技术书籍外,他还是《Software - Practice and Experience》杂志的北美地区编辑。Comer 是 ACM 会员 (Fellow)。

其他的信息可查询以下网址:

www.cs.purdue.edu/people/comer

David Stevens 从 Purdue 大学的计算机科学系获得了硕士 (1985) 和博士 (1993) 学位。从 1983 年起,他就是 BSD UNIX 系统的编程人员,主要从事于 BSD UNIX 内核的开发。他实现了大部分 TCP/IP 协议族,并和 Comer 博士一起合作编写了几本计算机科学的教科书。他感兴趣的专业方向是操作系统、计算机网络和大规模软件系统设计。

近年来,Stevens 在可伸缩联网 (scalable networking) 领域进行高性能多处理器系统的研究,为 Sequent Computer Systems 公司和 IBM 公司工作。他是 ACM 和 IEEE 的成员。

前言

Douglas E. Comer 博士的系列著作——《用 TCP/IP 进行网际互联》是一套令人瞩目的丛书。能向开放源码（open source）读者介绍该书的第三卷，对我来说实在是荣幸之至。

开放源码和 TCP/IP 的历史是紧密相连的：没有网络把你和合作者连接起来，就不能进行协作！而且，最早一批开放源码软件就有 TCP/IP 协议的实现。我记得在 20 世纪 80 年代初，“开放源码”还不像现在那样受媒体青睐，理解网络体系结构和实现的研究者屈指可数，而 Douglas 就是其中的佼佼者——他是一项广泛研究计划的负责人，全线出击，对当时遇到的很多问题提出挑战。

记得在 20 世纪 90 年代初，我们已经看到将技术应用到大工程领域的巨大趋势，这些领域渴望着知识和解决方案。那时，为公司构造基于互联网的环境，对工程师来说还是一个巨大的挑战。于是，Douglas 便开始教导他们，让他们能够掌握下层网络的复杂性，给他们提供辛勤耕耘得来的经验教训。

21 世纪来临了，新一代的设计者正在为因特网编写分布式应用程序。当前，我们听到许多激动人心的因特网应用，如 napster、gnutella 以及 infrasearch。但奇怪的是，现在的开发人员很少有人牢固掌握网络工程原理——坦率地说，他们缺乏对基础的理解，这种缺乏不可避免地造成了应用程序的适应能力不强或者干脆就不能工作。

正因为如此，Doug 与 David L. Stevens 合著的第三卷：客户-服务器编程与应用才与今天的因特网息息相关。这本书教给我们如何设计和构建客户-服务器应用程序，而且更重要的是，它还教给我们如何理解每种设计决策中所蕴涵的利弊得失。

我希望读者能够像业界前辈一样，从 Comer 博士的智慧中获益。

Marshall T. Rose

Theorist, Implementor, and Agent Provocateur

Petaluma, California

序 言

Linux 操作系统声名正旺，作为服务器系统，它对联网界尤其重要。这本使用 Linux 的新版第三卷是为那些渴望了解如何创建联网应用的程序员撰写的。大致说来，本书考察这样的问题，“应用软件如何使用 TCP/IP 协议通过因特网进行通信？”。本书重点研究了客户-服务器范例，并考察了在分布式程序中客户和服务器的两部分所用的算法。本卷举例说明了每种设计，并讨论了包括应用层网关和隧道等技术。另外，本卷还重温了几个标准应用协议，并用它们来说明一些算法和实现技术。

尽管本卷可以单独阅读和使用，但它实际是和另外两卷共同构成了一套丛书。丛书第一卷考虑的问题是：“什么是 TCP/IP 互联网？”；第二卷考察的问题是：“TCP/IP 软件是如何工作的？”，它给出了更多的细节，考察了工作代码，比第一卷探讨得更深入。因此，虽然程序员可以只通过第三卷学习创建网络应用，但学习其他各卷可以更好地理解下层技术。

第三卷的这个新版本包含了最新的技术，如，有一章解释了 Linux 程序如何利用 POSIX 线程设施创建并发服务器；关于 NFS 的章节讨论了 NFS 的第三版，这一版将为 Linux 界采用。此外，还有部分章节解释了 slirp 等程序所蕴涵的概念，这种程序能通过拨号电话连接访问因特网，而不要求每台计算机有一个惟一的 IP 地址。

还有两章显得特别及时，它们集中讨论了流式概念以及相关的技术，这些技术用于通过因特网发送音频和视频数据。第 28 章描述了实时协议（RTP）、编码、抖动缓存等基本概念。第 29 章展示了用于接收和播放 MP3 音频的 RTP 实现。

本书代码可在线获得。要通过万维网得到一个副本，可在以下网址的联网书籍清单中查找第三卷：

<http://www.cs.purdue.edu/homes/comer/netbooks.html>

要通过 FTP 访问代码，使用以下网址：

<ftp://ftp.cs.purdue.edu/pub/comer/TCPIP-vol3.linux.dist.tar.Z>

本书前几章介绍了客户-服务器模型，以及应用程序用于访问 TCP/IP 协议软件的套接字（socket）接口。此外，还描述了并发进程和用于创建进程的操作系统函数。随后的几章介绍了客户和服务器的设计。

本书阐明了各式各样可能的设计并不是没有规则的。实际上，这些设计都遵循了一种模式。在考虑了并行性和传输的选择后，就可以理解这一观点。例如，有一章讨论了使用面向连接传输（如 TCP）的非并发服务器设计，而另一章讨论了相似的设计，但它使用无连接传输（如 UDP）。

我们描述了每个设计如何适应于各种可能的实现，但是，并没有试图开发一种客户-服务器交互的抽象“理论”。我们只是强调实用的设计原则，以及对程序员很重要的技术。每种技术在某些情况下都有其优点，并且每种技术都已用于正在工作的软件中。我们相信，理解这些设计之间的概念联系，将有助于读者理解每种方法的优缺点，并更容易在它们之间进行选择。

本书包含了多个例子程序，他们展现了各种设计实际上是如何进行的。大多数例子实现了标准

的TCP/IP应用协议。在每一种情况下，我们都试图选择一个应用协议，使它可表达一种设计思路而又不太难理解。因此，虽然很少有令人激动的例子程序，但这里的每一个例子都说明了一个重要的概念。在第三卷的这个版本中，所有的例子程序都使用Linux套接字机制（即套接字API）；本书还有两个其他版本，他们含有相同的例子，只不过使用了微软的Windows Sockets和AT&T的TLI接口。

后几章集中讨论中间件，讨论了远程过程调用的概念，并描述它是怎样被用于构造分布式程序的。这些章将远程过程调用技术与客户-服务器模型相联系，并说明如何使用软件从远程过程调用描述生成出客户和服务器程序。有关TELNET的章节展现了细枝末节如何在一个实际工作的程序中占据支配地位，以及即使是实现一个简单的、面向字符的协议，其代码如何会变得复杂。本部分最后两章是关于流式传输协议的。

本书很大部分的重点在并发处理。编写过并发程序的学生可能熟悉我们所描述的许多概念，因为这些概念适用于所有的并发程序，而不仅仅是网络应用。没编写过并发程序的学生可能会觉得这些概念很难。

本书适于作为向高年级学生教授“套接字编程”，或向低年级研究生介绍分布式计算的一个学期的课程。由于本书重点是如何使用互联网，而不是互联网是如何工作的，因此学生几乎不需要太多的网络背景知识就能理解这些内容。只要教师按合适的进度循序渐进，本科生课程中不会有特别的概念令人感到太难。介绍操作系统概念或并发编程实际经验的基础课程，可提供最佳背景材料。

学生只有亲手使用教材后，才会欣赏它。因此，任何课程都应安排编程实践，强迫学生将其想法运用到实际程序中。大学本科生可通过反复设计其他的应用协议来学习基本概念。研究生则应构建更为复杂的分布式程序，这些程序强调一些细微的技术（如第16章中的并发管理技术和第18章和第19章中的互连技术）。

在此要感谢许多人的帮助。Purdue大学因特网研究小组的成员们给本书原稿提供了许多技术信息和建议。Michael Evangelista校对了本书并编写了RTP代码。Gustavo Rodriguez-Rivera阅读了本书的许多章节，并做了很多实验测试细节，还编辑了附录1。Dennis Brylow对本书许多章节提出了建议。Christine Comer进行了修订并改进了行文和一致性。

Douglas E. Comer

David L. Stevens

目 录

第 1 章	引言和概述	1
1.1	使用 TCP/IP 的因特网应用	1
1.2	为分布式环境设计应用程序	1
1.3	标准和非标准的应用协议	1
1.4	使用标准应用协议的例子	1
1.5	telnet 连接的例子	2
1.6	使用 TELNET 访问其他服务	3
1.7	应用协议和软件的灵活性	4
1.8	从提供者的角度看服务	4
1.9	本教材的其余部分	5
1.10	小结	5
	深入研究	5
	习题	5
第 2 章	客户 - 服务器模型与软件设计	7
2.1	引言	7
2.2	动机	7
2.3	术语和概念	8
2.3.1	客户和服务器	8
2.3.2	特权和复杂性	8
2.3.3	标准和非标准客户软件	9
2.3.4	客户的参数化	9
2.3.5	无连接的和面向无连接的服务器	10
2.3.6	无状态和有状态服务器	10
2.3.7	无状态文件服务器的例子	11
2.3.8	有状态文件服务器的例子	11
2.3.9	标识客户	12
2.3.10	无状态是一个协议问题	13
2.3.11	充当客户的服务器	13
2.4	小结	14
	深入研究	14
	习题	15
第 3 章	客户 - 服务器软件中的并发处理	16
3.1	引言	16
3.2	网络中的并发	16

3.3	服务器中的并发	17
3.4	术语和概念	18
3.4.1	进程概念	18
3.4.2	局部和全局变量的共享	19
3.4.3	过程调用	20
3.5	一个创建并发进程的例子	20
3.5.1	一个顺序执行的 C 实例	20
3.5.2	程序的并发版本	21
3.5.3	时间分片	22
3.5.4	单线程的进程	23
3.5.5	使各进程分离	23
3.6	执行新的代码	24
3.7	上下文切换和协议软件设计	25
3.8	并发和异步 I/O	25
3.9	小结	25
	深入研究	26
	习题	26
第 4 章	协议的程序接口	27
4.1	引言	27
4.2	不精确指明的协议软件接口	27
4.2.1	优点与缺点	27
4.3	接口功能	28
4.4	概念性接口的规约	28
4.5	系统调用	28
4.6	网络通信的两种基本方法	29
4.7	LINUX 中提供的基本 I/O 功能	29
4.8	将 Linux I/O 用于 TCP/IP	30
4.9	小结	31
	深入研究	31
	习题	31
第 5 章	套接字 API	32
5.1	引言	32
5.2	Berkeley 套接字	32
5.3	指明一个协议接口	32
5.4	套接字的抽象	33
5.4.1	套接字描述符和文件描述符	33
5.4.2	针对套接字的系统数据结构	34
5.4.3	主动套接字或被动套接字	35
5.5	指明端点地址	35

5.6	类属地址结构	35
5.7	套接字 API 中的主要系统调用	36
5.7.1	socket 调用	37
5.7.2	connect 调用	37
5.7.3	send 调用	37
5.7.4	recv 调用	37
5.7.5	close 调用	38
5.7.6	bind 调用	38
5.7.7	listen 调用	38
5.7.8	accept 调用	38
5.7.9	在套接字中使用 read 和 write	38
5.7.10	套接字调用小结	39
5.8	用于整数转换的实用例程	39
5.9	在程序中使用套接字调用	40
5.10	套接字调用的参数所使用的符号常量	40
5.11	小结	41
	深入研究	41
	习题	41
第 6 章	客户软件设计中的算法和问题	43
6.1	引言	43
6.2	不是研究细节而是学习算法	43
6.3	客户体系结构	43
6.4	标识服务器的位置	44
6.5	分析地址参数	45
6.6	查找域名	45
6.7	由名字查找某个熟知端口	46
6.8	端口号和网络字节顺序	47
6.9	由名字查找协议	47
6.10	TCP 客户算法	48
6.11	分配套接字	48
6.12	选择本地协议端口号	48
6.13	选择本地 IP 地址中的一个基本问题	49
6.14	将 TCP 套接字连接到某个服务器	49
6.15	使用 TCP 与服务器通信	50
6.16	从 TCP 连接中读取响应	50
6.17	关闭 TCP 连接	51
6.17.1	对部分关闭的需要	51
6.17.2	部分关闭的操作	51
6.18	UDP 客户的编程	51
6.19	连接的和非连接的 UDP 套接字	52

6.20	对 UDP 使用 connect	52
6.21	使用 UDP 与服务器通信	52
6.22	关闭使用 UDP 的套接字	53
6.23	对 UDP 的部分关闭	53
6.24	关于 UDP 不可靠性的警告	53
6.25	小结	53
	深入研究	54
	习题	54
第 7 章	客户软件举例	55
7.1	引言	55
7.2	小例子的重要性	55
7.3	隐藏细节	55
7.4	针对客户程序的过程库例子	56
7.5	connectTCP 的实现	56
7.6	connectUDP 的实现	57
7.7	构成连接的过程	57
7.8	使用例子库	60
7.9	DAYTIME 服务	60
7.10	针对 DAYTIME 的 TCP 客户实现	61
7.11	从 TCP 连接中进行读	62
7.12	TIME 服务	63
7.13	访问 TIME 服务	63
7.14	精确时间和网络时延	63
7.15	针对 TIME 服务的 UDP 客户	64
7.16	ECHO 服务	65
7.17	针对 ECHO 服务的 TCP 客户	66
7.18	针对 ECHO 服务的 UDP 客户	67
7.19	小结	69
	深入研究	70
	习题	70
第 8 章	服务器软件设计的算法和问题	71
8.1	引言	71
8.2	概念性的服务器算法	71
8.3	并发服务器和循环服务器	71
8.4	面向连接的和无连接的访问	72
8.5	传输协议的语义	72
8.5.1	TCP 语义	72
8.5.2	UDP 语义	73
8.6	选择传输协议	73

8.7	面向连接的服务器	73
8.8	无连接的服务器	74
8.9	故障、可靠性和无状态	74
8.10	优化无状态服务器	75
8.11	四种基本类型的服务器	76
8.12	请求处理时间	77
8.13	循环服务器的算法	77
8.14	一种循环的、面向连接的服务器的算法	78
8.15	用 INADDR_ANY 绑定熟知端口	78
8.16	将套接字置于被动模式	78
8.17	接受连接并使用这些连接	79
8.18	循环的、无连接的服务器的算法	79
8.19	在无连接的服务器中构造应答	79
8.20	并发服务器的算法	80
8.21	主线程和从线程	80
8.22	并发的、无连接的服务器的算法	81
8.23	并发的、面向连接服务器的算法	81
8.24	服务器并发性的实现	82
8.25	把单独的程序作为从进程来使用	82
8.26	使用单线程获得表面上的并发性	83
8.27	各服务器类型所适用的场合	83
8.28	服务器类型小结	84
8.29	重要问题——服务器死锁	85
8.30	其他的实现方法	85
8.31	小结	85
	深入研究	86
	习题	86
第9章	循环的、无连接服务器 (UDP)	87
9.1	引言	87
9.2	创建被动套接字	87
9.3	进程结构	90
9.4	TIME 服务器举例	91
9.5	小结	92
	深入研究	92
	习题	93
第10章	循环的、面向连接的服务器 (TCP)	94
10.1	引言	94
10.2	分配被动的 TCP 套接字	94
10.3	用于 DAYTIME 服务的服务器	95

10.4	进程结构	95
10.5	DAYTIME 服务器举例	95
10.6	关闭连接	98
10.7	连接终止和服务器的脆弱性	98
10.8	小结	98
	深入研究	99
	习题	99
第 11 章	并发的、面向连接的服务器 (TCP)	100
11.1	引言	100
11.2	并发 ECHO	100
11.3	循环与并发实现的比较	100
11.4	进程结构	101
11.5	并发 ECHO 服务器举例	101
11.6	清除游离 (errant) 进程	104
11.7	小结	105
	深入研究	105
	习题	105
第 12 章	将线程用于并发 (TCP)	106
12.1	引言	106
12.2	Linux 线程概述	106
12.3	线程的优点	106
12.4	线程的缺点	107
12.5	描述符、延迟和退出	107
12.6	线程退出	108
12.7	线程协调和同步	108
12.7.1	互斥	108
12.7.2	信号量	108
12.7.3	条件变量	109
12.8	使用线程的服务器实例	109
12.9	监控	113
12.10	小结	113
	深入研究	113
	习题	114
第 13 章	单线程、并发服务器 (TCP)	115
13.1	引言	115
13.2	服务器中的数据驱动处理	115
13.3	用单线程进行数据驱动处理	116
13.4	单线程服务器的线程结构	116
13.5	单线程 ECHO 服务器举例	117

13.6 小结	119
深入研究	119
习题	120
第 14 章 多协议服务器 (TCP, UDP)	121
14.1 引言	121
14.2 减少服务器数量的动机	121
14.3 多协议服务器的设计	121
14.4 进程结构	122
14.5 多协议 DAYTIME 服务器的例子	122
14.6 共享代码的概念	125
14.7 并发多协议服务器	125
14.8 小结	126
深入研究	126
习题	126
第 15 章 多服务服务器 (TCP, UDP)	127
15.1 引言	127
15.2 合并服务器	127
15.3 无连接的、多服务服务器的设计	127
15.4 面向连接的、多服务服务器的设计	128
15.5 并发的、面向连接的、多服务服务器	129
15.6 单线程的、多服务服务器的实现	129
15.7 从多服务服务器调用单独的程序	130
15.8 多服务、多协议设计	131
15.9 多服务服务器的例子	131
15.10 静态的和动态的服务器配置	137
15.11 UNIX 超级服务器, inetd	138
15.12 inetd 服务器的例子	140
15.13 服务器的几种变形清单	141
15.14 小结	141
深入研究	142
习题	142
第 16 章 服务器并发性的统一、高效管理	143
16.1 引言	143
16.2 在循环设计和并发设计间选择	143
16.3 并发等级	143
16.4 需求驱动的并发	144
16.5 并发的代价	144
16.6 额外开销和时延	144
16.7 小时延可能出麻烦	145

16.8	从线程 / 进程预分配	146
16.8.1	Linux 中的预分配	146
16.8.2	面向连接服务器中的预分配	147
16.8.3	互斥、文件锁定和 accept 并发调用	147
16.8.4	无连接服务器中的预分配	148
16.8.5	预分配、突发通信量和 NFS	149
16.8.6	多处理器上的预分配	149
16.9	延迟的从线程 / 进程分配	149
16.10	两种技术统一的基础	150
16.11	技术的结合	150
16.12	小结	151
	深入研究	151
	习题	151
第 17 章	客户进程中的并发	153
17.1	引言	153
17.2	并发的优点	153
17.3	运用控制的动机	153
17.4	与多个服务器的并发联系	154
17.5	实现并发客户	154
17.6	单线程实现	155
17.7	使用 ECHO 的并发客户例子	156
17.8	并发客户的执行	160
17.9	例子代码中的并发性	161
17.10	小结	161
	习题	162
第 18 章	运输层和应用层的隧道技术	163
18.1	引言	163
18.2	多协议环境	163
18.3	混合网络技术	164
18.4	动态电路分配	165
18.5	封装和隧道技术	166
18.6	通过 IP 互联网的隧道技术	166
18.7	客户和服务器之间的应用级隧道技术	166
18.8	隧道技术、封装和电话拨号线	167
18.9	小结	168
	深入研究	168
	习题	168
第 19 章	应用级网关	169
19.1	引言	169