

《网络编程与分层协议设计》

图书基本信息

书名：《网络编程与分层协议设计》

13位ISBN编号：9787111350521

10位ISBN编号：7111350529

出版时间：2011-7

出版社：机械工业

作者：刘飏

页数：252

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《网络编程与分层协议设计》

内容概要

《网络编程与分层协议设计:基于Linux平台实现》以Linux网络套接字编程和网络分层协议的设计与程序实现为主题，详细介绍如何在Linux平台下进行套接字程序设计，并给出了一个基于分层协议的应用实例，用于模拟Linux网络协议栈中IP层的核心功能—IP报文的分段和重组过程。《网络编程与分层协议设计:基于Linux平台实现》旨在通过具有一定复杂度的应用，切实帮助读者掌握网络协议分层的设计思想和程序实现方法。此外，《网络编程与分层协议设计:基于Linux平台实现》的许多程序示例直接使用了Linux的内核链表和散列链表，以及内核的其他数据结构和多线程等程序设计技巧，通过实际应用的形式有效地衔接了C语言、数据结构、操作系统、计算机网络和网络协议分析等课程的相关内容。

《网络编程与分层协议设计:基于Linux平台实现》可作为高等院校计算机、网络工程、通信工程等专业本科生与研究生“网络程序设计”课程的教材，也可作为相关领域工程技术人员的参考用书。

《网络编程与分层协议设计》

书籍目录

前言教学和阅读建议第1章 TCP/IP网络编程基础 1.1 网络和网络互联 1.1.1 ISO/OSI网络模型和TCP/IP协议栈 1.1.2 IP地址和端口 1.2 客户端/服务器编程模型 1.2.1 循环式服务器 1.2.2 并发式服务器 1.3 本章小结 习题第2章 Linux下的C编程环境 2.1 进程 2.1.1 进程标识 2.1.2 创建进程 2.1.3 终止进程 2.1.4 僵死进程 2.1.5 子进程退出的异步善后处理 2.2 线程 2.2.1 线程标识 2.2.2 创建线程 2.2.3 终止线程 2.2.4 线程同步 2.3 信号 2.3.1 信号的概念 2.3.2 常用信号及其含义 2.3.3 信号集和信号掩码 2.3.4 信号处理 2.3.5 可重入函数 2.4 时间 2.4.1 时间表示 2.4.2 获得时间 2.4.3 休眠时间 2.4.4 间隔定时器 2.5 文件 2.6 本章小结 习题第3章 网络编程中常用的典型知识 3.1 结构体大小的计算 3.2 数据存储与字节序 3.3 常用数据结构 3.3.1 Linux链表 3.3.2 Linux散列链表 3.4 函数指针 3.5 校验和 3.6 用户态下多定时器的实现 3.7 本章小结 习题第4章 基础套接字 4.1 套接字基本概念 4.1.1 创建套接字 4.1.2 域和地址族 4.1.3 套接字地址 4.1.4 流式套接字和数据报套接字 4.1.5 使用套接字 4.1.6 套接字编程常用基本函数和数据类型 4.1.7 客户端/服务器通信方式 4.2 流式套接字客户端/服务器编程 4.3 并发流式套接字服务器编程 4.3.1 服务器主进程 4.3.2 服务器子进程 4.3.3 服务器子进程终止 4.4 数据报套接字客户端/服务器编程 4.5 本章小结 习题第5章 高级套接字 5.1 套接字选项 5.1.1 getsockopt和setsockopt 5.1.2 套接字通用选项示例 5.2 I/O多路复用 5.2.1 I/O多路复用模型 5.2.2 select函数 5.2.3 文件描述符集合的使用 5.2.4 文件描述符的可读写状态 5.2.5 最大文件描述符个数 5.2.6 select函数的应用 5.3 非阻塞I/O 5.3.1 非阻塞I/O模型 5.3.2 fcntl函数设置非阻塞模式 5.3.3 非阻塞模式下的read和write 5.3.4 非阻塞模式下的connect 5.3.5 非阻塞模式下的accept 5.4 信号驱动I/O 5.4.1 信号驱动I/O模型 5.4.2 设置套接字工作于信号驱动I/O模式 5.4.3 信号驱动I/O示例 5.5 事件I/O 5.5.1 创建epoll上下文环境epoll_create 5.5.2 epoll设置epoll_ctl 5.5.3 等待事件发生epoll_wait 5.5.4 epoll程序示例 5.5.5 边沿触发ET和水平触发LT 5.6 原始套接字 5.6.1 创建原始套接字 5.6.2 原始套接字数据发送 5.6.3 原始套接字数据接收 5.7 本章小结 习题第6章 网络协议 6.1 IP协议 6.1.1 IP协议格式 6.1.2 IP协议头的C语言定义 6.1.3 IP报文分段 6.1.4 IP报文分段重组 6.2 ICMP协议 6.2.1 ICMP消息类型 6.2.2 ICMP消息通用格式 6.2.3 ICMP查询请求和应答消息格式 6.2.4 ICMP消息格式的C语言定义 6.3 FTP协议 6.3.1 FTP连接管理 6.3.2 FTP通信过程 6.3.3 FTP命令处理 6.3.4 数据(文件)传输过程 6.3.5 数据传输端口 6.4 一个简单应用层协议示例 6.5 分层协议示例—IP报文分段和重组模拟 6.5.1 程序功能描述 6.5.2 分层协议栈 6.5.3 分层协议定义 6.6 本章小结 习题第7章 ICMP协议程序设计 7.1 Ping程序功能模块 7.2 回应请求和应答实例分析 7.3 Ping程序实现 7.4 Ping程序的编译和测试 7.5 本章小结 习题第8章 FTP协议程序设计 8.1 FTP交互命令实例分析 8.2 FTP客户端程序功能模块 8.3 FTP客户端程序实现 8.4 FTP客户端程序的编译和测试 8.5 本章小结 习题第9章 IP报文分段和重组模拟程序设计 9.1 主机工作流程 9.2 转发服务器工作流程 9.3 消息分段 9.4 主机程序功能模块 9.5 转发服务器程序功能模块 9.6 消息分段的错序和丢失模拟 9.7 关键数据结构 9.8 主机程序实现 9.9 转发服务器程序实现 9.10 IP报文分段和重组模拟程序的编译和运行 9.11 本章小结 习题附录A C/C++源文件编辑附录B C/C++源文件编译附录C C/C++源文件调试参考文献

《网络编程与分层协议设计》

章节摘录

版权页：插图：

《网络编程与分层协议设计》

编辑推荐

《网络编程与分层协议设计:基于Linux平台实现》以Linux为平台，循序渐进地介绍了Linux网络套接字编程的基本原理和使用方法，并着重突出了网络分层协议的设计和程序实现。全书通过多个示例展示了Linux内核通用数据结构的使用，充分体现了数据结构+算法的程序设计思想，帮助读者切实掌握常用数据结构及其应用场景，从而快速提高网络软件的设计水平。程序示例丰富，具有一定复杂度。《网络编程与分层协议设计:基于Linux平台实现》提供了大量的程序示例，对Linux网络套接字编程的基本概念和协议分层的设计实现进行了详细说明。这些示例的设计和选择都经过精心安排，其中较为基础的程序示例除了演示基本的功能外，还反复作为后续较复杂程序示例的基本功能组件。从而形成了程序示例的有机搭配，既可帮助读者了解网络编程和分层协议设计的基本知识，又能让读者通过这些实际示例反复运用所学知识，达到熟能生巧和融会贯通。广泛使用优秀开源代码。《网络编程与分层协议设计:基于Linux平台实现》的程序示例大量直接或间接应用了优秀的开源软件，例如Linux内核的通用链表和散列链表在多个程序示例中作为基础组件而广泛使用。此外，《网络编程与分层协议设计:基于Linux平台实现》还参考并模拟了Linux网络内核源码中IP报文分段和重组的核心处理过程。通过学习优秀开源软件的设计实现，并且在正确的场合使用各类优秀开源软件，是快速提高网络编程能力的一条捷径。内容紧凑，中心突出。《网络编程与分层协议设计:基于Linux平台实现》围绕网络分层协议设计和实现这个中心进行讲解，所介绍的基础知识都是为掌握后续较复杂的示例而进行的必要铺垫，基本上做到了使读者在不参考其他书籍的情况下就能顺利掌握《网络编程与分层协议设计:基于Linux平台实现》的核心内容。读者对象广泛。《网络编程与分层协议设计:基于Linux平台实现》既可以单独作为网络编程的入门教材。也可以作为网络协议分析等相关课程的辅助实践教材。

《网络编程与分层协议设计》

精彩短评

- 1、这本书对于网络的IP层、TCP层和应用层若干协议的实现描述详尽，而且整本书不罗嗦，是一本突出重点的好书。
- 2、比起一般的入门教程又详细了很多，值得刚入门的同学们买
- 3、这本书里的例子是拿linux内核源码来举例的学习起来能扩展视野
- 4、课本也没有好与不好的
- 5、我的选修课教材。。
- 6、很难看懂。书挺好的。是这个东西本来就难。
- 7、尼玛这书居然有
- 8、不错的一本书，例子很丰富

1、《网络编程与分层协议设计》的笔记-第2页

1.协议分层

在TCP/IP协议族中，网络地体系结构分为5层，分别是物理层、数据链路层、网络层(IP)、传输层(TCP/UDP)和应用层。协议是指各层通信实体就本层所交换数据地表示方法和含义达成地一致约定。

2.套接字对

源IP、源端口、目的IP和目的端口4个参数又称为套接字对，主机之间地通信就是通过建立在一对套接字对之间的逻辑通道进行地。

IP地址仅用于标识参与通信地源和目的主机(只负责将数据从源主机发送到目的主机)；端口号用来标识主机中地通信进程(通信实际发生在位于网络中不同主机中的应用或进程)。

端口号(16位无符号整数 0~65535)

0 ~ 1023 众所周知端口，由IANA(Internet Assigned Numbers Authority)负责分配和管理

1024~49151 可注册端口，若需使用应向IANA注册并由IANA公开，以免发生重复

49152~65535 动态端口 任意进程可使用

3.客户端/服务器编程模型

对于TCP协议，服务端通常可使用并发式服务器进行编程(众所周知端口工作于被动打开模式用于监听客户端的连接请求，为每个已经建立连接地客户端分配一个临时端口用于数据交换，Linux下通常采用父子进程实现此工作方式)；

对于UDP协议，通常在服务器端使用循环式服务器编程(服务器一次只处理一个请求，当处理结束向客户端发回应答之后，才能接受下一个客户端请求)。

《网络编程与分层协议设计》

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com