HZ BOOKS
华章科技

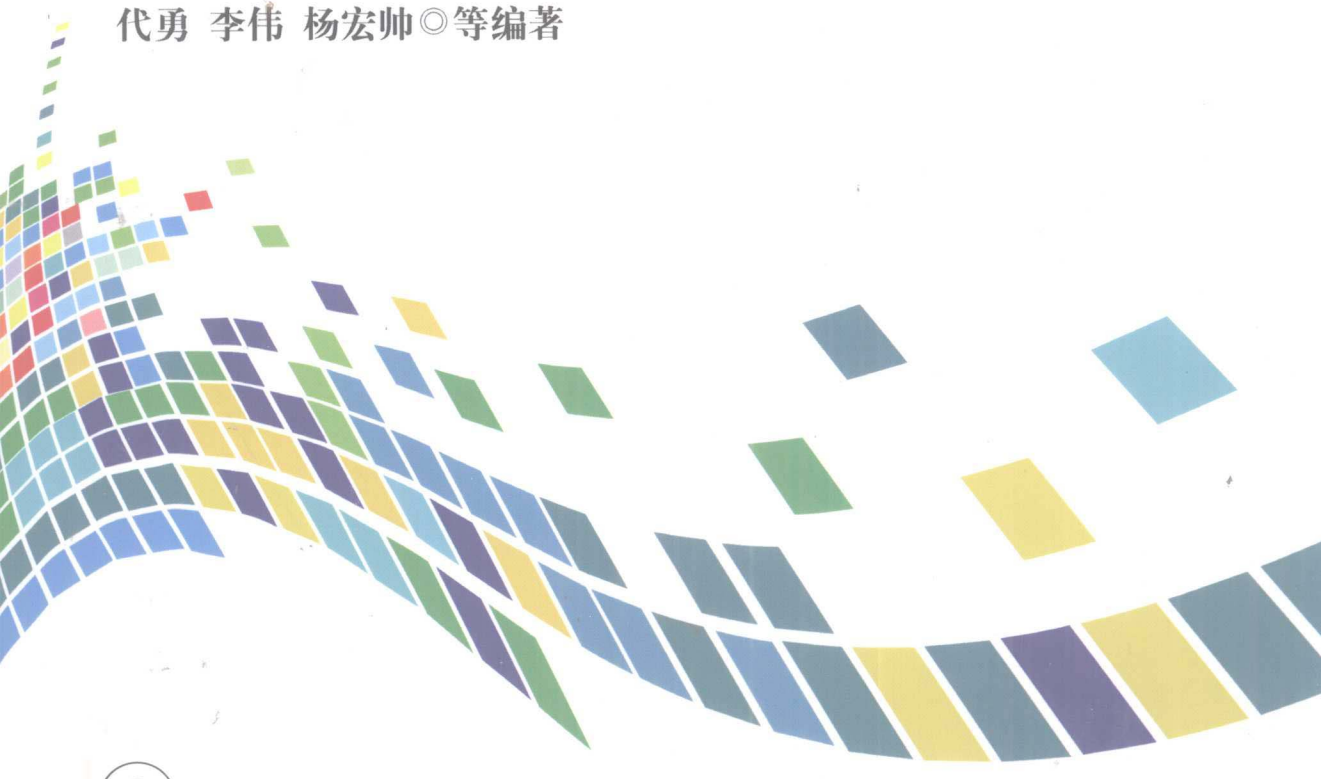
一线开发人员联手打造、汇集网络通信编程解决方案的经典之作

Visual C++

网络通信编程技术详解

Network Communication Programming Using Visual C++

代勇 李伟 杨宏帅◎等编著



CD-ROM

机械工业出版社
China Machine Press

Visual C++
网络通信编程技术详解

Network **C**ommunication **P**rogramming **U**sing Visual C++

代勇 李伟 杨宏帅◎等编著

机械工业出版社
China Machine Press

本书共18章，主要内容包括：TCP/IP协议模型与基础知识、Windows网络编程基础与网络的基本应用、IP配置信息管理程序设计、ARP表管理程序设计、基于Winsock的客户端/服务器端开发技术、路由管理程序设计、本地网络活动监视和端口扫描、TCP穿越NAT的P2P通信技术、即时通信与TCP/IP超级终端、FTP协议、HTTP协议、Telnet协议、SMTP和POP3协议、网络安全与防火墙设计、串口通信程序设计等。

本书内容全面，深入浅出，层次分明，注重知识的系统性、针对性和先进性，注重理论结合实践，培养工程应用能力。另外，本书还配有完整的综合实例源程序代码，便于读者在学习和实际开发中参考使用。

本书适合Visual C++编程技术人员、网络技术人员、网络安全管理人员和网络系统分析等相关领域的研究人员、工程技术人员、教师和学生作为技术参考手册使用，也适合网络程序设计初学者作为教材使用。

图书在版编目（CIP）数据

Visual C++网络通信编程技术详解/代勇等编著. —北京：机械工业出版社，2011.3

ISBN 978-7-111-33457-6

Ⅰ. V… Ⅱ. 代… Ⅲ. C语言—程序设计—应用—计算机通信网 Ⅳ. ①TP393②TP312

中国版本图书馆CIP数据核字（2011）第024735号

机械工业出版社（北京市西城区百万庄大街22号　邮政编码　100037）
责任编辑：张少波
北京诚信伟业印刷有限公司印刷
2011年5月第1版第1次印刷
185mm×260mm　·24.75印张
标准书号：ISBN 978-7-111-33457
　　ISBN 978-7-89451-869-9（光盘）
定价：55.00元（附光盘）

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88378991；88361066
购书热线：（010）68326294；88379649；68995259
投稿热线：（010）88379604
读者信箱：hzjsj@hzbook.com

PREFACE 前　言

网络编程技术是指利用 Windows 系统所提供的各类网络操作函数以及流行的网络协议，如 TCP/IP、FTP、HTTP、Telnet 等实现网络程序功能的基本原理和方法。目前，网络编程技术已在互联网技术应用、网络信息传输、网络信息安全、网络电子商务、网络管理信息系统等诸多领域得到非常广泛的应用。

网络编程技术的研究和应用离不开程序设计，Visual C++ 则是最有力、最常用的网络程序开发工具之一。本书全面地介绍了在 Visual C++ 环境下进行网络程序设计的方法，内容涵盖了 Windows 网络编程技术的理论基础和常用网络协议的 Visual C++ 应用技术，各种协议与方法在本书中均给出了基本原理、典型实例及其完整的 Visual C++ 源码，读者在理解了代码功能之后可以参考使用本书中提供的代码，进行修改、增加功能来组合成各种功能强大的网络应用程序。

在学习完本书之后，相信读者能够掌握 Visual C++ 环境下的网络编程技术，并可以进行网络程序实际项目设计开发工作。

本书特点

本书主要有以下特点。

1. 循序渐进，由浅入深

为了更好地吸引读者的注意力，激发读者的学习兴趣，本书首先介绍日常生活中最为常用的 TCP/IP 协议模型，然后介绍 Windows 网络编程基础和网络基本应用，在后续章节中结合具体的程序实例，逐步介绍流行的网络协议，如 TCP/IP、FTP、HTTP、Telnet 等实现网络程序功能的基本原理和方法。

2. 技术全面，内容充实

在保证实用的前提下，本书详细介绍了网络编程技术各个方面的知识。同时，结合程序实例介绍了应用网络编程技术进行网络程序开发的相关知识，各类网络基本功能应用、网络信息安全技术、电子邮件技术、黑客攻击网络的方法等都可以从本书中找到相关的知识介绍。

3. 分析原理，步骤清晰

掌握一门技术首先需要理解原理，本书注意把握各个知识点的原理，总结实现的思路和步骤。读者可以根据具体步骤实现书中的例子，将理论知识与实践相结合，这样更利于学习。

4. 代码完整，讲解详尽

本书中的每个知识点都有相应的实例代码，并对关键的代码部分进行了注释说明。每段代码的后面都有详细的分析，并给出了代码运行后的结果。读者可以参照运行结果阅读源程序，以便于加深理解。

5. 结合应用，注重实践

本书提供了大量综合应用实例，结合程序实例详细介绍了网络应用程序的开发过程。每章最后还配有“实践拓展”部分，介绍最新的网络编程技术应用知识，拓展读者视野，提高读者的实际应用能力。

主要内容

本书共18章，各章的主要内容如下。

第1章：首先介绍了OSI参考模型的层次结构、层次结构划分的原则和各层次之间的数据封装关系，然后介绍了目前应用的TCP/IP协议的基本构成和主要内容。

第2章：详细介绍了网络通信中涉及的基本概念，如IP地址和子网掩码等。结合在TCP/IP协议栈中自上而下的数据封装过程，对数据包的层次关系、封装过程和拆包过程进行了详细阐述，对传输层中的TCP和UDP协议的数据报结构字段与端口进行了讲解，对网络层的IP数据报各字段的意义和路由实现进行了详细介绍。

第3章：主要对Winsock编程原理、Winsock I/O模型、扩展性、套接字概念、I/O控制命令等内容进行了详细介绍，对WinInet网络编程基础和MFC Windows Sockets网络编程基础进行了简单描述。

第4章：主要介绍了一些网络的基本应用方法，包括如何获取主机的计算机名、域名和IP地址、子网掩码、网关、DNS序列以及网卡类型、物理地址信息等。

第5章：介绍了如何应用IP帮助函数来获取和设置本地计算机上的IP配置信息。

第6章：介绍了局域网中ARP协议的工作原理以及如何获取ARP表的各个函数，并结合实例讲述了如何应用ARP协议与函数。

第7章：介绍了使用Winsock开发典型客户端/服务器端程序的方法，并结合一个简单的TCP服务器实例来讲述Winsock的编程方法。

第8章：介绍了与路由管理相关的IP帮助函数，以及如何应用这些函数进行路由跟踪的程序实现方法。

第9章：结合程序实例讲述如何使用编程的方式来获取正在使用TCP/UDP进行网络访问的进程信息。同时，介绍了端口扫描的意义和实现方法，并以简单程序实例来说明如何对TCP/IP端口进行扫描。

第10章：介绍了如何利用NAT穿越技术来实现可靠的P2P的TCP流通信。

第11章：介绍了实现一个功能简单的即时通信程序的方法，用这种方法可以同时运行多个客户端连接到服务器端来进行文字信息交换。同时，也介绍了超级终端的作用，并以实例的形式说明了超级终端的基本功能。

第12章：介绍了FTP工作原理以及如何开发FTP服务器端和客户端程序。

第13章：介绍了HTTP协议的基本内容，并通过HTTP下载程序实例来进一步说明和讲解。

第14章：介绍了Telnet协议的基本内容，并结合实例进行说明。

第15章：介绍了SMTP模型的基本内容以及电子邮件接收/发送技术的实现方法，并结合实例来拓展说明。

第16章：介绍了网络防火墙的类型、特点、工作原理，以及如何利用Filter-Hook Driver来实现网络防火墙。

第17章：在第6章的基础上进一步详细介绍ARP工作原理、工作方式，并结合实例介

绍典型的局域网计算机诊断、ARP 欺骗与 ARP 表中毒、密码侦测、DNS 欺骗等内容。

第 18 章：介绍利用 PC 串口进行串行通信的基本原理，并根据实现串口通信的两种编程方式分别介绍了 MSComm ActiveX 控件和 Windows API 函数的使用方法。

读者对象

- Visual C++ 编程人员
- 大中专院校的学生
- 社会培训班的学生
- 高等教育学校的学生
- 网络系统管理员
- Web 开发人员
- 网络编程人员

本书光盘

包含原书各章实例和综合应用实例的完整源代码及测试用的图像文件，读者可以按照书中的说明对程序源代码进行编译和运行。

本书主要由代勇、李伟和杨宏帅编写，其中第 1 ~ 9 章由代勇编写，第 10 ~ 13 章、第 16 ~ 18章由李伟编写，第 14 和 15 章由杨宏帅编写。参与本书编写和资料整理工作的还有朱瑛、封海波、马云辉、管殿柱、赵景波、付本国、张轩、赵景伟、赵秋玲、张忠林、王献红、王臣业、张洪信等。代勇完成了全书的统稿工作，并和李伟审校了全书。本书在编写过程中得到了海军潜艇学院宋一兵高工和哈尔滨工业大学林琳副教授的大力支持，在此特别表示感谢！

感谢您选择了本书，希望我们的努力对您的工作和学习有所帮助，也希望您把对本书的意见和建议告诉我们。

作者：gdz_zero@126.com

编辑：zsb@hzbook.com

编　者

2010 年 12 月

CONTENTS 目 录

第 1 章　TCP/IP 协议模型

TCP/IP 是英文名称 Transmission Control Protocol/Internet Protocol 的缩写，通常译为传输控制协议/因特网协议。TCP/IP 协议是 Internet 的基础协议。该协议由网络层的 IP 协议和传输层的 TCP 协议组成。本章将介绍 OSI 参考模型基础和 TCP/IP 协议的基本构成。

1.1　OSI 参考模型

开放式系统互联参考模型（Open System Interconnection Reference Model，OSI/RM），是国际标准化组织（ISO）提出的一个试图使各种计算机在世界范围内互联为网络的标准框架。它提供给开发者一个必需的、通用的概念以便可以用来解释连接不同系统的框架。OSI/RM 本身并非一个国际标准，至今尚未出台严格按照 OSI/RM 定义的网络协议或国际标准，但是在制定有关网络协议和标准时都会把 OSI/RM 作为参考模型，并说明与该参考模型之间的对应关系，这正是 OSI/RM 的意义所在。OSI 参考模型的结构如图 1-1 所示。

图 1-1　OSI 参考模型的结构

1. OSI 参考模型的层次结构

OSI 参考模型一共划分为七层，其层次划分的主要原则为：

- 网中各节点都具有相同的层次；
- 不同节点的同等层具有相同的功能；
- 同一节点内相邻层之间通过接口通信；
- 每一层可以使用下层所提供的服务，并向其上层提供服务；
- 不同节点的同等层通过协议来实现对等层之间的通信。

第 1 ~3 层主要负责通信，统称为通信子网层。第 5 ~7 层属于资源子网，统称为资源子网层。第 4 层（传输层）起着衔接上三层和下三层的作用。各层主要负责的工作如下：

1）物理层：提供为建立、维护和拆除物理链路所需的机械、电子、功能和规程的特性；提供有关在传输介质上传输非结构的位流及物理链路故障检测的指示。

2）数据链路层：为网络层实体提供点到点无差错帧传输功能，并进行流控制。

3）网络层：为传输层实体提供端到端的交互网络数据传送功能，使得传输层摆脱路由选择、交换方式、拥挤控制等网络传输细节问题；可以为传输层实体建立、维护和拆除一条或多条通信路径；对网络传输中发生的不可恢复差错予以报告。

4）传输层：为会话层实体提供透明、可靠的数据传输服务，保证端到端的数据完整性；选择网络层能提供的最恰当服务；提供建立、维护和拆除传输连接功能。

5）会话层：为彼此合作的表示层实体提供建立、维护和结束会话连接的功能；完成通

信进程的逻辑名字与物理名字间的对应；提供会话管理服务。

6）表示层：为应用层进程提供能解释所交换信息含义的一组服务，如代码转换、格式转换、文本压缩、文本加密与解密等。

7）应用层：提供 OSI 用户服务，例如事务处理程序、电子邮件和网络管理程序等。

2. OSI 参考模型中的数据传输过程

在 OSI 参考模型中，将信息从上一层传送到下一层是通过命令方式实现的。这里的命令称作原语（Primitive），被传送的信息称为协议数据单元（Protocol Data Unit，PDU）。在 PDU 进入下一层之前，会在 PDU 中加入新的控制信息，这种控制信息称为协议控制信息（Protocol Control Information，PCI）。然后，会在 PDU 中加入发送给下一层的指令，这些指令称为接口控制信息（Interface Control Information，ICI）。

PDU、PCI 与 ICI 共同组成了接口数据单元（Interface Data Unit，IDU）。下一层在接收到 IDU 后，就会从 IDU 中去掉 ICI，此时的数据包称为服务数据单元（Service Data Unit，SDU）。随着 SDU 一层一层地向下传送，在每一层中都会加入自身的信息。图 1-2 表示相邻层数据包的处理过程。

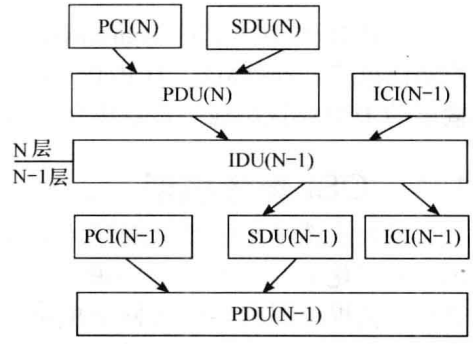


图 1-2　相邻层数据包的处理过程

在 OSI 参考模型中，数据流的传输过程如图 1-3 所示。

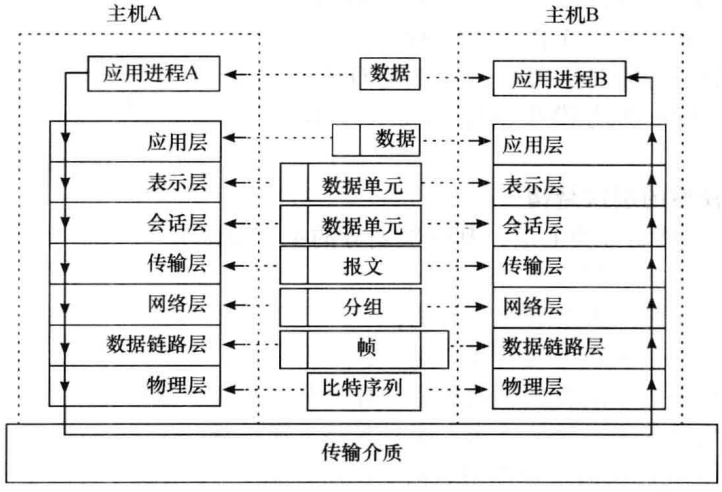


图 1-3　OSI 参考模型中的数据流传输过程

依据图 1-3，我们可以得出数据传输过程主要包括以下几个步骤：

1）当应用进程 A 的数据传送到应用层时，应用层为数据加上本层控制报头后，组织成应用层的数据服务单元，然后再传输到表示层。

2）表示层接收到该数据单元后，加上本层的控制报头，组成表示层的数据服务单元，再传送到会话层。依此类推，数据被传送到传输层。

3）传输层接收到该数据单元后，加上本层的控制报头，就构成了传输层的数据服务单元，称为报文（Message）。

4）传输层的报文传送到网络层时，由于网络层数据单元的长度有限制，传输层报文将

被分割成多个较短的数据字段，再加上网络层的控制报头，就构成网络层的数据服务单元，称为分组（Packet）。

5）网络层的分组传送到数据链路层时，再加上数据链路层的控制信息，就构成了数据链路层的数据服务单元，称为帧（Frame）。

6）主机A的数据链路层的帧被传送到物理层后，物理层将以比特流的方式通过传输介质传输出去。当比特流到达目的节点主机B时，再从物理层依次向上层传送，每层对应各层的控制报头进行处理，将用户数据传送到高一层，最终完成将主机A的数据传送给主机B的过程。

尽管主机A的数据在OSI参考模型中要经过复杂的处理过程，才能送到主机B的应用进程，但其实对于每台计算机的应用进程来说，OSI参考模型中数据流的复杂处理过程是透明的。应用进程A的数据似乎是“直接”传送给应用进程B，这就是开放系统在网络通信过程中所起到的最本质的作用。

事实上，OSI参考模型只是一个框架，它的每一层并不执行某种功能，功能的具体实现还需要协议，需要通过软件来实现。当数据在层间向下传输时，每一个层都会为传输中的数据增加一个包头（Header），用于标识包的来源与目的地。到了目的主机时，每一层都从数据中读取相应包头，执行所请求的任务，并负责向上传输数据包。

1.2 TCP/IP结构

TCP/IP协议是Internet的基础。虽然从名字上看TCP/IP包括两个协议，即传输控制协议（TCP）和因特网协议（IP），但TCP/IP实际上是一组协议，有上百种，如远程登录协议、文件传输协议和电子邮件协议等，而TCP协议和IP协议是保证数据完整传输的两个最基本的重要协议。所以说TCP/IP是Internet协议簇，而不只包括TCP和IP协议。

在20世纪70年代中期，美国国防部为其ARPANET广域网开发了网络体系结构和协议标准，以其为基础所组建的Internet是目前国际上规模最大的计算机网络。正因为Internet的广泛使用，TCP/IP协议成了事实上的标准。

表1-1是TCP/IP协议簇中一些常用协议的英文名称和用途说明。

表1-1 TCP/IP协议簇中的常用协议

协议名称	用　途
TCP（Transmission Control Protocol）	传输控制协议
IP（Internet Protocol）	因特网协议
UDP（User Datagram Protocol）	用户数据报协议
ICMP（Internet Control Message Protocol）	因特网控制报文协议
SMTP（Simple Mail Transfer Protocol）	简单邮件传输协议
SNMP（Simple Network Management Protocol）	简单网络管理协议
FTP（File Transfer Protocol）	文件传输协议
ARP（Address Resolution Protocol）	地址解析协议

1.2.1 TCP/IP模型

如图1-4所示，TCP/IP协议的分层模型由四个层次组成，分别为网络接口层、网络层、传输层和应用层，其各层的功能如下。

（1）网络接口层

网络接口层是TCP/IP协议的最底层，负责接收IP数据报并通过网络接口层进行比特流

传送，或者从网络上接收物理帧，并提取出 IP 数据报交给 IP 层。

（2）网络层

网络层负责相邻计算机之间的通信，其功能主要包括三方面：第一，处理来自传输层的分组发送请求。在收到请求后，将分组装入 IP 数据报，并填充报头，选择去往信宿机的路径，然后将数据报发往适当的网络接口。第二，处理输入数据报。首先检查其合法性，然后进行路径寻找：假如该数据报已到达信宿机，则去掉报头，将剩下部分交给适当的传输协议；如该数据报尚未到达信宿机，则转发该数据报。第三，处理路径、流控、拥塞等问题。

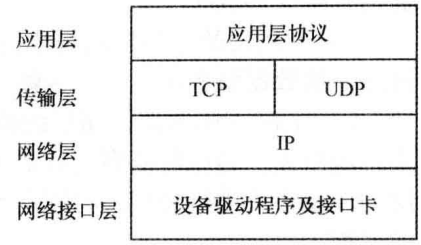


图 1-4　TCP/IP 模型的层次

（3）传输层

传输层提供应用程序间的通信功能，其功能包括两方面：第一，格式化信息流；第二，提供可靠传输。为实现可靠传输，传输层协议规定接收端必须发回确认，否则视为分组丢失，必须重新发送。

（4）应用层

应用层向用户提供一组常用的应用程序，如电子邮件、文件传输访问和远程登录等。

1.2.2　OSI 参考模型与 TCP/IP 模型的关系

在 1.1 节中已经介绍了关于 OSI 参考模型的相关概念，下面将 TCP/IP 模型与 OSI 参考模型进行比较，其层次关系如图 1-5 所示。

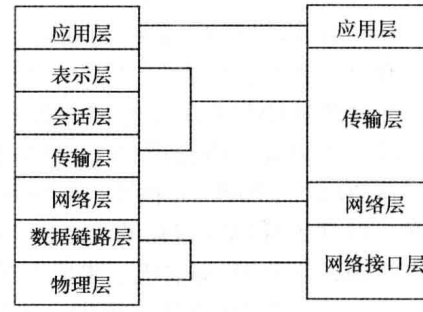


图 1-5　OSI 参考模型与 TCP/IP 模型的层次关系

位于 TCP/IP 协议簇第二层的 IP 层对应 OSI 参考模型的第三层网络层，位于 TCP/IP 协议簇第三层的 TCP 和 UDP 层对应 OSI 参考模型的第四层传输层。TCP 和 IP 是 TCP/IP 协议簇的中间两层，是整个协议簇的核心，起到了承上启下的作用。OSI 参考模型的各层功能与 TCP/IP 协议簇的对应关系如表 1-2 所示。

表 1-2　OSI 参考模型的各层功能与 TCP/IP 协议簇的对应关系

OSI 中的层	功　能	TCP/IP 协议簇
应用层	文件传输，电子邮件，文件服务，虚拟终端	TFTP、HTTP、SNMP、FTP、SMTP、DNS、Telnet
表示层	数据格式化，代码转换，数据加密	没有协议
会话层	解除或建立与别的节点的联系	没有协议
传输层	提供端对端的接口	TCP、UDP
网络层	为数据包选择路由	IP、ICMP、ARP、RARP、RIP、OSPF、BGP、IGMP
数据链路层	传输有地址的帧，错误检测	SLIP、CSLIP、PPP、MTU
物理层	以二进制数据形式在物理介质上传输数据	ISO 2110、IEEE 802、IEEE 802.2

1.2.3　TCP/IP 各层协议介绍

下面将对 TCP/IP 协议簇中的常用协议进行简要介绍，本书在后续章节中将对 TCP、UDP、IP 层的协议进行更为详细的描述与说明。

1. 应用层

应用层一般是指面向用户的服务，如HTTP、FTP、Telnet、DNS、SMTP和POP3等。

（1）HTTP

HTTP（Hyper Text Transport Protocol，超文本传输协议）是一个客户端和服务器端请求与应答的标准，其中客户端为终端用户，而服务器端则是网站。通过使用Web浏览器、网络爬虫或者其他工具，客户端能够发起一个到服务器上指定端口（默认端口为80）的HTTP请求，我们通常称这个客户端为用户代理（User Agent）。

应答的服务器上存储着一些资源，如HTML文件和图像，这时我们称该应答服务器为源服务器（Origin Server）。事实上，在用户代理和源服务器间可能存在多个中间层，比如代理、网关或者隧道（Tunnel）。尽管TCP/IP协议是互联网上最流行的应用，但HTTP协议并没有规定必须使用它和基于它支持的层。事实上，HTTP可以在任何其他互联网协议或其他网络上实现。HTTP只是假定其下层协议提供可靠的传输，任何能够提供这种保证的协议都可以被HTTP使用。

通常，由HTTP客户端发起一个请求，建立一个到服务器指定端口的TCP连接，而HTTP服务器则在指定端口监听客户端发送过来的请求，一旦HTTP服务器收到请求，服务器便向客户端发回一个状态行（如“HTTP/1.1 200 OK”）和响应消息。该响应消息的消息体可能是请求的文件、错误消息或其他一些信息。

HTTP使用TCP而不是UDP的原因在于，打开网页必须传送很多数据，而TCP协议可以提供传输控制并且按顺序组织数据和进行错误纠正。

（2）FTP

FTP用于在Internet上控制文件的双向传输。同时，它也是一个应用程序（Application），用户可以通过它将自己的PC机与世界各地所有运行FTP协议的服务器相连，来访问服务器上的文件信息。

在TCP/IP协议中，FTP标准命令中的TCP端口号为21，Port方式的数据端口号为20。在FTP协议的使用过程中，对处于网络连接的两台计算机所处的位置、连接方式、是否使用相同的操作系统均无特殊的要求。假设两台计算机通过FTP协议进行对话，并同时都能访问Internet，则可以使用FTP命令来进行文件传输。事实上，每种操作系统在使用上都有某些细微差别，但是每种协议基本的命令结构则是相同的。具体地说，FTP的文件传输有两种方式：ASCII传输模式和二进制数据传输模式。

1）ASCII传输模式。在ASCII传输模式中，如果用户正在复制的文件包含简单的ASCII码文本，则在进行文件传输时，FTP通常会自动地调整文件的内容以便把文件解释成另外一台计算机能够存储的文本文件格式。

2）二进制传输模式。如果用户正在传输的文件包含的不是文本文件，它们可能是程序、数据库、字处理文件或者压缩文件（尽管字处理文件包含的大部分是文本，其中也包含有指示页尺寸和字库等信息的非打印字符），则在复制任何非文本文件之前，需要用binary（二进制）命令告诉FTP逐字复制，即采用二进制传输模式。

在二进制传输模式中，首先得保存文件的位序，以便原始数据和复制数据是逐位对应的。如果在ASCII传输模式下传输二进制文件，在传送的时候也需要转译，这会使传输速度稍微变慢，同时也会损坏数据，使文件变得不可用。在大多数计算机上，ASCII传输模式一般假设每一字符的第一有效位无意义，因为ASCII字符组合不使用它。但如果传输二进制文件，则所有的位都是重要的。

（3）Telnet 协议

Telnet 协议作为 TCP/IP 协议簇中的一员，是通过 Internet 进行远程登录服务的标准协议和主要方式，它为用户提供了在本地计算机上完成在远程主机上工作的能力。在终端用户的计算机上使用 Telnet 程序，通过它连接到服务器，终端用户便可以在 Telnet 程序中输入命令，这些命令将会在服务器上运行，就像直接在服务器的控制台上输入一样，从而实现在本地远程控制服务器。如要开始一个 Telnet 会话，则必须输入用户名和密码来登录远程服务器。Telnet 是常用的远程控制 Web 服务器的方法。

Telnet 协议最初是由 ARPANET 开发的，现在主要用于 Internet 会话，其基本功能是允许用户登录远程主机系统。起初它只是让用户的本地计算机与远程计算机连接，从而成为远程主机的一个终端。而较新的版本支持在本地执行更多的处理，既可以提供更好的响应，同时也减少了通过链路发送到远程主机的信息数量。

（4）DNS 协议

DNS（Domain Name Service，域名解析服务）提供域名到 IP 地址之间的转换。在 Internet 上，域名与 IP 地址之间是一对一或多对一的关系。域名的作用是方便使用者记忆，但是计算机之间却只能识别 IP 地址；在域名与 IP 地址之间的转换工作被称为域名解析。

例如，在浏览网页时我们输入如 www.sina.com.cn 的网址，这是一个域名，而计算机网络上的计算机间只能用 IP 地址才能相互识别。当然，我们也可以在 IE 的地址栏中输入“202.108.33.32”的 IP 地址，但是这样的 IP 地址很难被使用者记住，那么应用域名解析服务，使用者只需要记住域名便可以登录相应的服务器。

域名解析需要由专用的域名解析服务器来完成，DNS 就是进行域名解析的服务器。当用户在应用程序中输入名称时，DNS 服务可以将此名称解析为与之相关的其他信息（如 IP 地址）。例如，我们在上网时所输入的网址，就是通过域名解析系统找到了相对应的 IP 地址，才能连接到对应的服务器，也就是说域名最终指向的是 IP 地址。

在 IPv4 中，IP 是由 32 位二进制数组成的，并将这 32 位二进制数分成 4 组，每组 8 位，将这 8 位二进制数转换成十进制数，即常见的 IP 地址，其范围为 0～255。在已开始试运行阶段、将会代替 IPv4 的 IPv6 中，将以 128 位二进制数来表示 IP 地址。

（5）SMTP

SMTP 是由一组用于由源地址到目的地址传送邮件的规则组成的，由这些规则来控制信件的中转方式，其在传输文件过程中使用 25 号端口。SMTP 属于 TCP/IP 协议簇，能帮助每台计算机在发送或中转信件时找到下一个目的地。通过 SMTP 所指定的服务器就可以把电子邮件寄到收信人的服务器。

SMTP 的一个重要特性是能够跨越网络传输邮件，即实现“SMTP 邮件中继”功能，该特性使得 SMTP 不必依赖特定的传输网络，而只需要发送端和接收端能够建立可靠有序的数据流信道。通常，一个互联网络上的主机可以分为三类：互联网上利用 TCP 协议可直接相互访问的主机、带防火墙分隔的利用 TCP 传输层协议可相互访问的主机、其他利用非 TCP 传输层协议可相互访问的主机。我们使用 SMTP 可实现相同网络上处理机之间的邮件传输，也可通过中继器或网关实现某处理机与其他网络上处理机之间的邮件传输。

（6）POP3 协议

POP3（Post Office Protocol 3，邮局协议的第 3 个版本）是规定个人计算机如何连接到互联网上的邮件服务器来进行收发邮件的协议。POP3 协议是因特网电子邮件的第一个离线协议标准，允许用户从服务器上把邮件下载到本地主机上，同时根据客户端的操作指令删除或保存邮件服务器上的邮件。而 POP3 服务器则是遵循 POP3 协议的接收邮件服务器，只下载

邮件，服务器并不删除邮件。

在工作过程中，POP3 客户向 POP3 服务器发送命令并等待响应，POP3 命令采用命令行形式，用 ASCII 码表示。服务器响应是由一个单独的命令行或多个命令行组成的，响应第一行以 ASCII 文本 + OK 或 + EORR（OK 示意成功，EORR 示意失败）指出相应的操作状态是成功还是失败。

在 POP3 协议中有三种状态，即认证状态、处理状态和更新状态。当客户端与服务器建立连接时，客户端向服务器发送自己的身份（这里指的是账户和密码），并由服务器确认，客户端由认证状态转入处理状态。在完成列出未读邮件等相应的操作后，客户端发出 QUIT 命令，即退出处理状态进入更新状态。开始下载未读邮件到本地计算机，完成之后，重新返回到认证状态，并在确认身份后断开与服务器的连接。

2. 传输层

传输层协议主要是指 TCP 和 UDP。

（1）TCP

TCP 是面向连接的通信协议，并提供一种可靠的数据流服务。它通过三次握手建立连接，在通信完成时拆除连接。由于 TCP 是面向连接的，它只能用于点对点的通信。它采用“带重传的肯定确认”技术来实现传输的可靠性。TCP 还采用一种称为“滑动窗口”的方式进行流量控制，窗口大小表示接收能力的大小，用以限制发送方的发送速度。

在客户端与服务器端进行通信前，要先交换传输层控制信息，为双方的通信做好准备。在这个握手阶段之后，就可以认为在这两个进程间存在一个 TCP 连接，且是一个全双工的连接。在消息发送完后，应用程序会通知 TCP 拆除这个连接。可靠的传输服务是为了保障彼此通信时能无差错地顺序传送所有数据。

当其中任何一个应用程序把字节流传送给套接字时，它可以指定 TCP 把同样的字节流传送到对方的套接字。TCP 能够调节数据传输过程中的拥塞问题，拥塞调节机制将在网络处于拥塞时阻止发送进程。确切地说，TCP 拥塞控制试图把每个 TCP 连接限定在公平共享网络带宽的基础上。

同时，TCP 也有其缺点，即 TCP 不保证最小传输速率。TCP 不允许发送进程以设想的速率发送数据，受到 TCP 拥塞机制的调节，发送进程有可能被迫以一个较低的平均速率发送。另外，TCP 不提供任何延时保障，发送进程把数据传入自己的 TCP 套接字后，这个数据将最终到达其接收套接字，但是中间所经历的时间无法保证，发送过程所消耗的时间长短也不能确定。

（2）UDP

UDP 是一个非面向连接的传输协议，在两个进程彼此通信之前没有握手过程，提供不可靠的数据传输服务。也就是说，当一个进程往自己套接字发送一个消息时，UDP 不能保证这个消息会最终到达接收套接字。另外，对于到达接收套接字的消息而言，它们的到达顺序也可能不是有序的。

UDP 不含拥塞控制机制，即发送进程能够以任意速率往 UDP 套接字上发送数据，尽管不能保证所有数据都能够到达接收套接字，但是会有相当比例的数据到达。实时应用程序的开发人员往往选择在 UDP 上运行他们的应用。与 TCP 类似，UDP 也不提供任何延时保障。另外，由于 UDP 数据中包括了目的端口号和源端口号信息，通信不需要连接，所以可以实现广播发送。

表 1-3 给出了在应用层中常见的协议所具体采用的传输协议类别。