微机实用新技术丛书

TCP/IP与网络互联技术

裘实　阳光　晓文　等编著

国防工业出版社

TCP/IP 与网络互联技术

裘实　阳光　晓文　等编著

国防工业出版社

·北京·

图书在版编目(CIP)数据

TCP/IP 与网络互联技术/裘实等编著．—北京：国防工业出版社，1998.5
（微机实用新技术丛书）
ISBN 7-118-01843-0

Ⅰ．T… Ⅱ．裘… Ⅲ．计算机网络-协议 Ⅳ．TP393

中国版本图书馆 CIP 数据核字(97)第 24376 号

国防工业出版社出版发行
（北京市海淀区紫竹院南路 23 号）
（邮政编码 100044）
北京怀柔新华印刷厂印刷
新华书店经售
*
开本 787×1092 1/16 印张 19 434 千字
1998 年 5 月第 1 版 1998 年 5 月北京第 1 次印刷
印数：1—4000 册 定价：27.00 元

编 者 的 话

在当今计算机技术突飞猛进发展的时代，如何快速、准确地吸收和采用新技术，推动国民经济现代化的发展，是我们每个科技工作者的责任。一方面是铺天盖地、眼花缭乱的新名词、新软件，一方面是无数刚刚接触计算机概念，缺乏专业知识，而又在工作中急需使用最新软件技术的各行各业工作人员及计算机爱好者。为了更快、更好地架起这两者之间的桥梁，几位有识之士在国防工业出版社阎瑞琪主任的倡导下，组织起来，克服困难，认真负责、快速编写了这套微机实用新技术丛书，并将进一步把握新技术的脉搏，推出更多、更好的书籍奉献给读者。

本套丛书内容新颖、实用，语言简练，选题着眼读者最广泛的软件与技术，力求使读者在短时间内掌握、吸收，并能灵活运用。

参加丛书编撰的全部是具有博士或硕士研究生学位，并在计算机领域从事实用技术研究和开发的专家及学者。

丛书全体编者衷心企望本丛书能对广大读者的学习和工作提供有力的帮助，并衷心感谢阎瑞琪主任的支持和帮助。

由于时间仓促，本丛书无论在选题策划还是在编写细节上，都可能有不足之处，恳切希望广大读者批评指正。

前　言

随着 Internet 国际互联网和多媒体技术的发展，网络正以极快的速度进入人们的生活，TCP/IP 作为开放网络通信协议，对于 Internet 网络入门者和有意深入了解的读者来说无疑是最重要的内容。

本书适用于使用 TCP/IP 协议的系统管理人员。本书不专门讨论某个厂商的产品，而尽可能地研究多种不同的 TCP/IP 版本及其特色以及主要的操作系统。现有的许多书都是从基本原理的角度讨论 TCP/IP。在这一点上本书与这些书籍一致，但是它同时又为读者指出如何实现该协议，并使之运行在多种不同的操作系统上。

本书是为 TCP/IP 网络协议的高级用户和系统管理人员设计的。本书各章节中包含常见的 TCP/IP 环境的信息和处理过程，这些信息和处理过程是经过若干小时的故障检修和环境管理后总结出来的。

本书第一章介绍数据通信和网络基础知识；第二章到第五章重点介绍 Internet 与 TCP/IP 的基本知识和技术，包括 TCP/IP 原理、寻址方式、远程访问和文件传送以及路由选择；第六章专门讨论帧中继和 ATM 技术；本书第七章介绍了简单网络管理协议；第八章对与 Internet 非常有关的域名系统进行了专门讨论；与函件发送有关的 SMTP 将在第九章介绍；网络的安全性是每个人都很关心的问题，读完第十章后，相信读者会对这个问题有一个全面的了解；第十一章到第十五章分别讨论了 TCP/IP 协议在 NetWare，DOS 和 Windows，Windows NT，IBM OS/2 和 UNIX 上的实现；第十六章介绍了当前最热门的 Internet 上网技术及使用方法；第十七章与读者讨论了网络故障诊断问题。

裘实、阳光、晓文、李纪鸿 4 位同志参加了本书部分编写工作；刘海燕、赵雪梅、魏忠才 3 位同志负责本书的资料收集工作；李梅和李岳对本书进行了仔细审核；李敏对本书的编排付出了辛勤的劳动，在此对以上同志一并深表感谢。

作　者

目　录

第一章　数据通信技术基础

本章集中讨论常用的联网技术以及它们与协议、标准等的关系。这是本书唯一讨论基础知识的章节，其他所有章节都假定读者已经明白了联网概念和确立了实现 TCP/IP 的目标。

网络技术的飞速发展是 20 世纪末计算机领域的骄傲，TCP/IP 通信协议由于其开放性和实用性而受到普遍使用，在越来越多的读者加入到网络大家庭中的时候，在 Internet，3W，Java，Network Computer（简称 NC，网络计算机）风靡全球的世纪之末，了解一些网络基础知识，将会受益无穷，作为网络基础的数据通信技术是每一个读者通向网络圣殿的起点。本章通过讨论数据通信以及分布式环境的主要元素来综述各种形式的数据通信以及分布式环境。由于协议和模型与联网及客户机/服务器环境有关，所以在这里也对其进行了讨论。

分布式环境由跨越多个位置的不同系统资源（数据、计算能力等）构成。这些资源利用通信系统实现相互间的作用。这种情况下，通信系统是提供互相交换控制信息和数据的分布机制的媒介。通信系统对信息分配至关重要，它能够成为对最终用户是透明的，或者使最终用户能够看到提供实际资源互联的网络。在任何一种情况下，节点间的通信都是不可缺少的。节点间的通信需要连接相互作用的物理网络。

1.1　通信原理与分布式处理

客户机/服务器结构是协处理的一个子集，而协处理又是分布式处理的一个子集。分布式处理环境不仅仅是为客户机/服务器计算模式而建立，它还用于其他目的。目前促进分布式系统发展的因素有如下几点。

（1）微电子技术的发展改变了性能-价格比，这有利于建立多个低费用、高性能的系统。

（2）互联和通信费用快速减少。

（3）用户渴望更加经济、快速、先进和可靠的设施。

分布式处理的目标和优点是资源共享。一些资源，比如计算机、外设、专用处理器、程序、数据等，通过通信系统互联以达到共享这些资源的目的。互联系统形成的网络能够在不同地点、不同系统及不同终端和程序之间交换信息。

下述定义有助于理解网络和通信。

（1）通信系统是一系列硬件和软件的组合，能支持系统内和进程内不同节点的软件间的通信。节点通过网络进行互联，这提供了节点间的物理路径。两个或多个系统间的直接连接被称为“链”（Link）。

(2) 实现主要应用功能并且控制通信系统的系统被称作“主机”(或服务器)。

(3) 在分布式系统中,对象名代表一个系统、一个进程或者一个节点。一个地址代表命名的对象存在的地方。路径告诉读者如何到达那里。

数据是分布式系统中最经常共享的资源。大多数应用都需要那些拥有不同计算程序的用户共享数据。通过利用重发数据能够提高数据的可靠性。本地复制和分配的数据拷贝能够减少访问时间。通信系统能够用于不同地址、不同系统和程序间的数据传输和数据请求,尽管互联系统不一定构成分布式处理系统,但用于数据和信息交换的数据通信系统可以被看作是分布式处理系统。

1.2 通信系统的主要功能

通信系统的主要功能有命名和建立地址、分段、流量控制、同步、确定优先级以及纠错,下面分别加以说明。

1.2.1 命名和寻址

通信系统管理对象的名字。这些对象可以是进程、端口、邮箱、系统及用户间的会话。用户一般用符号形式提供名字,然后通信系统把这种形式用网络地址的形式重新表达。通信系统必须拥有逻辑名到物理名的转换表。

1.2.2 数据分段

如果需要传输的用户信息或者文件比网络包要大,通信系统必须把它分成多个段,并且在把它发往最终用户之前重新装配它。

具体原因如下。

(1) 太长的信息会增加其他用户的访问延时,因为太长的信息可能在较长的一段时间里独占网络的共享资源。

(2) 较短的信息能提高效率并减少传输错误率。

(3) 通信系统使用的内部缓存能够优化传输信息的大小。

(4) 组成某一发送路径的网络可以具有大小不同的包。

(5) 一些网络支持并行数据链路和传输。允许把长信息分段使用,这样便可以减少总的延时。

1.2.3 流量控制

许多网络都是在假定所有用户不会同时请求共享有限资源的基础上设计的。当网络通信量超过网络吞吐能力时,网络流量控制可通过协调通信实体之间的信息流来优化网络性能。

1.2.4 数据同步

在通信实体能够通信之前,它们的相互作用必须首先同步。如果接收方快于发送方,那么它可能获得并且错误地解释多余的信息。相反地,如果接收方慢于发送方,它便可能

丢失信息。

因为客户机/服务器结构总是把主进程分解为子进程，并且每个子进程在不同的主机上执行，所以必须有一种机制保证不同的子进程保持同步。相似地，当数据资源被共享或分布时，有必要协调分布资源以确保它们同步。IBM 的 Advanced Program to Program Communications(先进的程序到程序通信，简称 APPC)协议便是这种多级同步协议的例子。

1.2.5 确定优先级

通信系统能够为信息分配优先级，因而当竞争某项资源时，它便允许有区别地处理。高优先级的信息(告警、中断等)的延时较短。通信系统能够静态或动态地分配优先级(比如根据信息的内容，或者根据信息来源或目的等)。

1.2.6 差错控制

可靠的、无差错的通信是通信系统的一个主要目标。差错控制功能包括差错检测、纠错和恢复。差错检测可以通过下述 3 种方法实现。

(1) 信息冗余，冗余数据被用来进行比较以便决定是否有差错发生。

(2) 使用能够确定信息错误的控制信息。控制信息能够使用不同的算法来计算一个校验位或者所有信息位的校验和。通过将计算的结果和收到的结果进行比较，便可以检测到差错。

(3) 为信息指定序列号并检测序列错误。序列号用于确定丢失的、重复的或者脱离序列的信息。

纠错和恢复通过自动重发或者纠错编码来实现。

1.3 通信层次、协议和接口

通信系统负责在分布式系统的节点间提供通信。它允许网络节点向连接到通信网的任何其他节点发送信息。计算机网络结构允许同类型或不同类型的系统互联。

因为通信系统是复杂的，所以经常把它们分成不同的层。有些分层结构已具有标准化的模型。

1.3.1 通信的分层模型

为了阐述通信分层模型的概念，可构造一种用于说明通信的 3 层模型。考虑下面的 3 层模型，这 3 层模型是用于人与人之间进行通信的标准模型。

(1) 认识层包括概念，例如理解方式、知识、共享的东西以及通用符号等。这是人类可以理解的那一层。计算机用户在这一层实现接口。

(2) 语言层用词汇来表达概念。计算机使用 ASCII 或者 EBCDIC 字符。

(3) 物理传输层为实际通信提供介质。这一层可以有多种形式，比如空气中的声音振动、纸上的字或可视符号。数据通信可以使用电、无线电或者光信号作为传输介质。

上述例子说明了分层模式的本质。3 个分层之间是相互独立的。“上层”需要“下层”的支持。

分层结构的目标体现了结构化编程的目标，即利用明确定义的接口来定义功能模块。模块的概念清晰而且容易维护。假定某个模块的接口保持不变，那么模块的内部可以任意地进行修改。

1.3.2 客户机/服务器运算的分层模型

所有主要的网络都具有如下相同的目标。

(1) 连接性允许不同的硬件和软件能够互联到统一的、单系统映像的、网络化的系统。

(2) 模块化允许使用一些相对较小的通用组合软件建立不同用途的网络系统。

(3) 通过检错和纠错，支持无差错通信。

(4) 易于实现、使用和修改。

为了实现这些目标，网络体系支持模块化设计。每个模块的功能都被组织成为功能化分层。

在分析客户机/服务器的运算时，分层处理特别有用。分布式处理器之间的通信发生在多个级别上，从电缆上的信号到交换控制信息或数据的应用程序。在每一级别上，通信的本质有所不同，网络硬件使用电压或电流脉冲工作；应用程序使用一些包括名字的机制来工作，例如 Named Pipes 或者高级 Peer-to-Peer 通信。

客户机/服务器在所有这些层次上都具有这些特性，这样便可使得运行在不同计算机上的进程进行紧密结合。对层次的理解有助于了解客户机/服务器完成运算工作的方式。

在数据通信模式中，层次由实体构成。实体既可以是硬件也可以是软件进程。存在于不同网络节点的相同层的实体称为“同级实体”。在不同节点的同一级的层次被称为“同级层”。

典型的分布式系统结构由下述功能层构成。

(1) 应用层：体系结构的最高层。通常它完成应用进程的管理、数据的分配、进程内通信以及应用程序函数到可分布进程的分解。应用层的功能由低级层次支持。

(2) 分布式操作系统层：这个层次提供了应用层需要的全系统的分布式服务。它支持全程命名、目录、寻址、资源的共享、保护与同步、内部通信和恢复。分布式操作系统把分布功能统一成为单一的逻辑实体，并且负责建立单一系统映像(SSI)。

(3) 本地管理和核心层：这一层次支持单独节点上的操作系统。它支持本地进程内通信、内存和 I/O 访问、保护以及多任务。这一层次通过提供这些服务以及通过与其他节点同级层间的通信来支持高级别层次。

(4) 通信系统层：这一层次支持应用程序、分布式操作系统和本地管理层所需要的通信。

分层结构主要带来以下这些好处。

(1) 层次独立：每一层只关心相邻的下一层提供的服务。

(2) 灵活性：某一层次的变化不会影响其他层次。

(3) 易于实现和维护。

(4) 标准化：把分层功能、服务和接口做成结构化的实体，以使标准化工作变得容易开展。

跨越不同链路的通信是一项复杂的任务。除了在计算机技术中的应用外，“协议”这一术语可能是人们最熟悉的外交辞令。它用来表达各方都同意的行为规则。

在语言的使用上，社会生活要求人们每天都要遵循某种协议。语言的语法规定了人与人之间通信的规则。如果两个人使用相同的语言或者方言，他们便会很顺利地交换信息而没有什么障碍。如果一个德国人和一个日本人希望能够没有差错地交流，他们必须雇佣一个翻译。这个翻译理解语言的协议并且进行必需的翻译。

层次之间的通信由协议来管理。协议中包括信息交换的格式和顺序以及信息发送和接收中完成的动作。层次之间信息交换的规则和格式构成了层次接口。

通信协议之间的转换是必需的和精确的。在多数情况下，协议必须仔细地转换以免发生错误。

为了使得各通信协议概念化和易于管理，网络系统标准的早期开发者把这一过程分为几个具体的部分。

1.3.3 OSI 7 层通信模型

开放式系统互联 OSI 模型把数据从某点传输到另一点，这一任务分为 7 个不同的任务。任务被分层处理。每一层都有助于信息包的装配或分解。

数据以若干位为一组的方式通过网络的每一层，这一组数据位就叫作包(Packet)。每个包又分为如下 4 个不同的部分。

(1) 起始字符。

(2) 包头，指出这个信息包从哪里来及到哪里去以及包的类型(是数据包还是网络控制包)。

(3) 包中的数据。

(4) 最后的纠错位以及结束字符。

在 OSI 模型中，首先我们关心的是包头中的信息内容，它指明了信息的去向。在包头中，层次在发送包中建立，在接收包中被解开。层次按照同类方式安排。每一层只向紧挨着它的上一层或者下一层发送信息。图 1.1 显示了包的结构。

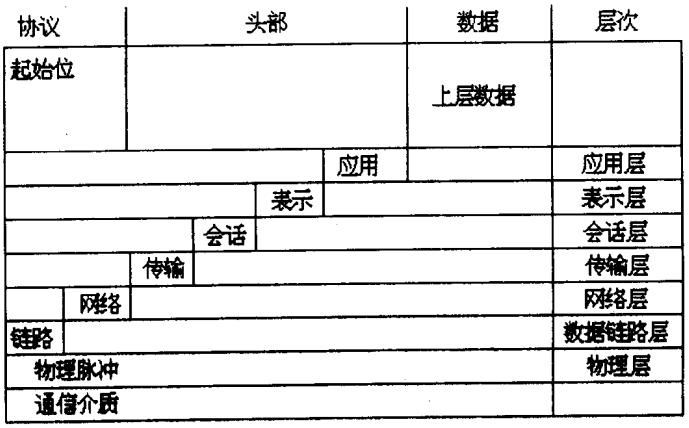

图 1.1 建立一个 OSI 包

每一层只向紧挨着它的上一层或下一层发送信息，但是它只能理解其他节点的同一层次发来的信息。比如网络层(第三层)把任何第三层信息取出后，再把 inbound 包发送到第四层。同样，这个第三层在把第三层信息加入包中之后，把 outbound 包发往第二层。对于接收包，它会对接收包进行检查以便确认是否需要采取某些动作。如果第三层信息需要把包传送到地址 04，然而接收方是地址 90，它便丢掉这个包，而不把它继续往下发。

对于发送包，第三层把源和目的地址增加到包中并且把扩大的包送往第二层等待处理。接收方的第三层只响应发送方第三层增加到包中的信息。节点上的每一层只和其他节点上的同一层进行通信，而不关心其他层做什么事情。

1. 物理层

物理层产生物理脉冲、电流以及涉及到把数据从网络接口卡(NIC)转移到通信系统所需要的光脉冲。RS-232 便是物理层标准的一个例子。物理层管理的是数据位。

物理层不包括通信系统，但包括与通信系统的连接。它处理上升时间和脉冲持续时间。物理层不关心连接器和电缆连接的细节。

2. 数据链路层

数据链路是收集数据位并且把数据作为包处理的第一层。这一层最后组装发送包，并且首次检查接收包。它为发送包增加纠错信息并且为接收包检错。不完整的或者不正确的包会在这里被丢弃。如果数据链路层能够确定错误包的来源所在，它会返回错误包。SDLC 和 HDLC 是在这一层运行的两个协议。

3. 网络层

当局域网(LAN)的规模超过一定范围或者地理区域后，必须把它们划分成为较小规模的 LAN。用路径选择器、网桥和网关划分 LAN 并生成子网。网络层通过多项设备对包进行路径选择以便确保数据包到达正确子网的正确设备处。

这一级保持一个路径选择表并确定哪条路径最快以及其他路径何时能用。这是设备开始对那些不能够从一个网络传递到另一个网络的包进行滤波的第一层，这样，网络通信量便可以降低。互联网协议(即 TCP/IP 中的 IP)在这一层工作，就像 NetWare 的 IPX 一样。这一层也是“无连接”及“数据流图”服务的层次。

4. 传输层

传输控制协议(TCP/IP 中的 TCP)工作在传输控制层。这是一个过渡层(管理路由包和错误恢复的最后一层)。它容纳了网络层中没有涉及到的缺陷。如果包在网络层被正确地接收，这一层次便很简单。如果通信系统无法提供可靠的包传输，这一层次便会很复杂。

5. 会话层

在很多网络设置中，都希望在通信实体间建立规范连接。这种连接保证了信息发送和接收的可靠性。这种要求在远程通信网中经常会有。所以，在大多数大型机通信中面向会话是正常的。LAN 一般被看作是高可靠的，在 LAN 通信中不经常采用会话控制。

会话层是保持“面向连接”的传输层。在这个层次上建立和打开连接的过程也就是“打包”和“拆包”的会晤过程。在这个级别上，包总是被假定为可靠的。检错不是这个层次的功能。TCP/IP 的 TCP 部分，IBM 的 NetBIOS 和 NetWare 的 SPX 部分在此层工作。

6. 表示层

此层未被完全定义或使用。此层中的处理用来实现应用层的可用数据在提交时的转

换。数据的压缩/解压缩和数据加密/解密过程能够在表示层实现。数据加密和压缩也能够通过在 OSI 应用层上运行的用户应用程序完成。

数据格式的转换也可以在这一层上实现。例如 ASCII 和 EBCDIC 之间编码方案的转换。然而,这个功能在大多数情况下也是通过用户应用程序实现的。

表示层经常被误认为把数据显示给用户。实际上,表示层是网络通信层,它并不直接和最终用户显示设备接口。屏幕显示的任务是用户工作站上的应用程序来完成的。

7. 应用层

应用层在 OSI 模型定义之中。该层处理安全事宜以及资源的有效性。应用层侧重于处理文件传递、任务传递以及虚拟终端协议。

8. 其他层

OSI 模型是在联网已经比较普遍而公共 LAN 协议未被广泛采用之前开发出来的。在它出现之后,它的分层定义已经进化成为不仅仅适合于 LAN,而且还适合于微机和大型机。DSI 模型增加了新的层次和子层次,例如增加了描述硬件细节(比如网络连接器和光纤)的第 0 层。

1.4 客户机/服务器连接部件

在客户机/服务器结构中,客户机和服务器系统是由一些互联着的部件组成的。每一种部件都有与其他部件连接的接口。客户机和服务器的通信可以看作相关部件之间的通信。进程部件和资源部件是两类不同的部件。进程部件是实现一定功能的软件,而资源部件提供进程部件要求的服务。

从客户机/服务器相互作用的角度来看,有源资源部件的行为类似于服务器,而资源部件和进程部件的用户的行为类似于客户机。

进程和资源部件为了通信的目的互相连接起来。这种连接是信息发出者与信息接收者之间的连接。客户机/服务器的连接既可能是动态的,也可能是静态的。静态连接建立在编译、装载或者在系统初始化时,而且可以改变。动态连接则可以在运行时随时改变。

1.4.1 通信与同步

在客户机/服务器环境中,各部分之间的协调与合作是通过通信系统的通信与同步动作实现的。通信功能涉及到信息的交换。它们由流量控制、差错控制、命名和寻址、阻塞和分段等功能组成。同步功能涉及到两个或多个部件之间的协调。

通信和同步紧密相关。当通信在共享内存中完成时,紧密耦合的系统(比如 Symmetric Multiprocessing 系统)和软件部分,用来实现同步。在更加传统的由通信网络互联的松散耦合系统中,类似信息传递的机制必须用于通信和同步。

在任何一种情况下,由通信系统提供的通信服务都可以是无连接的或是面向连接的。所谓无连接服务是指每一项事务与其他事务无关。

面向连接的服务提供了特定通信层传输的信息单元序列间的关系。这类似于在公共电话网中在两个电话用户间建立连接的过程。电话中心局的交换机在连接电话的系统中建立一条路径。每次路径都可能不相同。在路径建立之后,一条电路(Circuit)也就建立了,