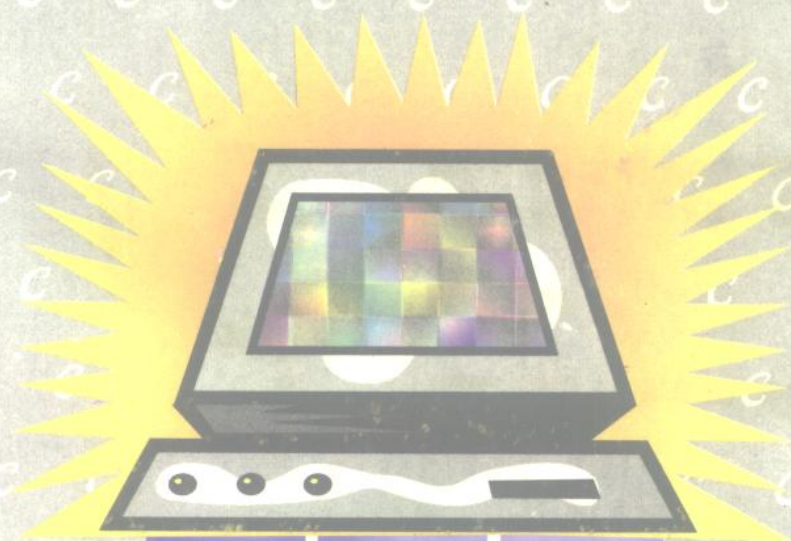


网络环境下的

编程技巧及实例

方敏 罗昌隆 雷震甲 编著



西安交通大学出版社

网络环境下的 C 编程技巧及实例

方敏 罗昌隆 雷震甲 编著

西安交通大学出版社

内容简介

本书从实用的角度介绍了计算机网络环境下的软件开发方法及如何在网络环境下利用已有的网络资源有效地开发网络应用程序。内容包括 UNIX、NetBIOS、Novell NetWare 网络环境下的编程设计方法。它包括网络开发环境、基本概念、基本应用接口和网络编程实例等。

本书根据作者的工作经验并参考了国内外同行较新的网络设计方法编著而成,结构简洁、内容新颖,可读性强。对大量的原理和抽象概念的叙述尽量简化,凡具有一定软件基础的读者阅读和理解此书的内容都不会感到困难。本书可供从事计算机网络研究、开发和教学的工程师以及大专院校的师生参考使用。

(陕)新登字 007 号

网络环境下的 C 编程技巧及实例

方敏 罗昌隆 曹震甲 编著

责任编辑 陈丽

组稿编辑 林全

*

西安交通大学出版社出版发行

(西安市咸宁路 28 号 邮政编码 710049)

西安华字印刷厂印装

各地新华书店经销

*

开本: 787×1092 1/16 印张: 15 字数: 326 千字

1996 年 7 月第 1 版 1996 年 7 月第 1 次印刷

印数: 1—5000

ISBN7-5605-0845-6/TP·131 定价: 20.00 元

前 言

有人说,90年代是计算机网络的年代。的确,在90年代,异种机网络和异网互联有了较大的突破。复杂的网络系统与相应的数据库系统等日益集成化。计算机网络大发展的结果,使得网络应用与开发日趋广泛。

为了适应计算机网络日益发展和应用的迅速普及,计算机网络的设计、应用开发以及网络编程技术的介绍显得迫切需要。与这些需要相比,实用的参考资料较少。除此以外,目前多数大专院校的计算机网络教材大都介绍计算机网络组成和组网原理,缺乏实践性的应用开发和具体编程技术的介绍。为了适应这样的形势需要,我们编著了此书。

书中融进了我们从事网络教学和开发的一些经验,还参阅了大量国内外有关的资料。内容分为以下几部分:

第1章及第2章介绍了计算机网络的基本概念和几种广泛使用的协议。

第3章介绍了 NetBIOS 支持的程序设计。

第4章给出了 NetWare 环境下 IPX/SPX 程序设计。

第5及第6章介绍了 UNIX 操作系统提供的网络环境及进程间通信的方法。

第7章至第9章介绍了通信程序的应用程序接口。主要介绍的是传输层接口 TL1、伯克利管套及它们的应用。

第10章介绍了网络选择及名称到地址的映射机制。

第11章介绍了 TCP/IP 这个被广泛使用的协议的作用,以及它的实现方法。

第12章介绍了文件的共享机制 RFS 和 NFS。

限于作者水平书中难免有不足之处,敬请广大读者批评指正。此外,本书在编写过程中参阅了大量的文献资料,其中主要部分列在本书后,在此向这些文献的作者表示感谢。

作 者

1996年6月

目 录

第1章 网络系统概述

1.1 计算机网络的形成与发展	(1)
1.2 网络模型	(3)
1.2.1 ISO/OSI 网络体系结构研究方法	(3)
1.2.2 ISO/OSI 参考模型	(4)
1.2.3 客户、服务器模型	(5)
1.3 计算机网络的组成	(6)
1.4 服务方式	(7)
1.4.1 面向连接的服务和非连接的服务	(8)
1.4.2 全双工与半双工连接	(8)
1.4.3 流控制	(8)
1.4.4 差错控制	(8)
1.4.5 字节流	(9)
1.5 同步方式	(9)
1.6 计算机网络基本术语	(9)
1.6.1 地址	(9)
1.6.2 连接与相关	(9)
1.6.3 打包与拆包	(9)
1.6.4 带外数据与缓存	(10)
1.6.5 分组交换	(10)
1.7 计算机网络分类	(10)
1.7.1 远程网	(11)
1.7.2 局域网	(11)
1.7.3 综合业务数字网	(12)

第2章 网络的通信协议

2.1 TCP/IP 协议集合	(13)
2.1.1 数据链路层	(14)
2.1.2 网络层	(14)
2.1.3 传送层 UDP 和 TCP	(16)
2.1.4 应用层	(17)
2.2 XNS Xerox 网络系统	(18)
2.2.1 网间传送协议	(18)
2.2.2 远程过程调用协议 Courier	(19)
2.2.3 数据链路层概述	(19)
2.2.4 网络层 IDP	(19)

2.2.5	传输层	(20)
2.2.6	应用层	(21)
2.3	SNA 系统网络结构	(21)
2.3.1	数据链路层	(22)
2.3.2	路由控制层	(22)
2.3.3	逻辑单元 LU6.2	(23)
2.3.4	应用层	(24)
2.4	NetBIOS 基本原理	(24)
2.4.1	名字服务	(26)
2.4.2	会话服务	(26)
2.4.3	数据报服务	(26)
2.4.4	通用命令	(26)
2.5	UNIX 到 UNIX 的拷贝 UUCP	(26)
第 3 章 NetBIOS 程序设计		
3.1	NetBIOS 概述	(28)
3.1.1	NetBIOS 提供的服务	(28)
3.1.2	网络控制块	(29)
3.1.3	应用程序结构	(30)
3.2	NetBIOS 命令	(33)
3.2.1	一般性管理命令	(34)
3.2.2	名字服务命令	(36)
3.2.3	数据报服务命令	(37)
3.2.4	会话服务命令	(38)
3.2.5	NetBIOS 命令的定义	(42)
3.3	NetBIOS 的基本程序	(43)
3.3.1	添加名字	(43)
3.3.2	发送数据报	(44)
3.3.3	接收数据报	(45)
3.3.4	删除名字	(46)
3.3.5	建立会话连接	(46)
3.3.6	通过会话连接发送和接收报文	(47)
3.3.7	关闭会话连接	(48)
3.3.8	测试会话状态	(49)
3.3.9	测试适配器状态	(50)
3.4	程序设计实例	(51)
第 4 章 IPX/SPX 程序设计		
4.1	IPX 协议	(73)
4.2	SPX 协议	(76)

4.3	IPX/SPX 的功能调用	(77)
4.3.1	IPX 功能调用	(77)
4.3.2	SPX 功能调用	(81)
4.4	IPX/SPX 程序设计指南	(85)
4.4.1	IPX 基本编程技术	(85)
4.4.2	SPX 基本编程技术	(92)
4.4.3	点对点通讯实例	(95)
第5章 UNIX 操作系统的网络环境		
5.1	UNIX 系统的一些基本概念	(104)
5.1.1	UNIX 系统的基本术语	(105)
5.1.2	标识符	(109)
5.1.3	文件	(110)
5.2	信号	(114)
5.2.1	信号的产生	(114)
5.2.2	信号的类型及定义	(115)
5.2.3	可靠信号	(117)
5.3	进程间的关系及相互制约	(118)
5.3.1	进程的创建	(118)
5.3.2	进程的终止和父子进程的同步	(119)
5.3.3	系统调用 EXEC	(120)
5.4	守护进程	(121)
5.4.1	守护进程的预处理	(122)
5.4.2	守护程序框架	(124)
第6章 进程间通信及实现方法		
6.1	管道	(126)
6.2	流	(130)
6.3	基于流的管道	(131)
6.4	流管道的应用	(134)
第7章 传送层接口 TLI		
7.1	引言	(136)
7.2	TLI 结构	(136)
7.3	TLI 函数	(136)
7.4	传输端点地址	(146)
第8章 TLI 的应用模式及传送层接口程序设计		
8.1	TLI 服务模式	(148)
8.2	实用网络库函数	(150)
8.3	面向连接的应用	(152)
8.4	面向无连接的应用	(161)

第9章 管套的应用

9.1 管套	(168)
9.2 管套地址	(170)
9.3 虚电路服务中管套的应用	(171)
9.4 数据报服务中的管套使用	(173)
9.5 管套函数	(174)
9.6 管套应用实例	(186)

第10章 网络选择及名称到地址的映射

10.1 概述	(193)
10.2 网络选择	(194)
10.3 名称到地址的映射	(196)

第11章 TCP/IP 协议的应用及实现

11.1 引言	(202)
11.2 TCP/IP 的环境	(203)
11.2.1 IP 地址(网际地址)	(203)
11.2.2 网际协议	(204)
11.2.3 UDP 和 TCP	(205)
11.2.4 ICMP	(205)
11.2.5 文件传送协议 FTP	(206)
11.2.6 简化的文件传输协议 TFTP	(207)
11.3 UDP、TCP 协议的 UNIX 实现	(210)
11.3.1 网络环境中资源的管理	(210)
11.3.2 IP 协议的实现	(213)
11.3.3 ICMP 协议的实现	(214)
11.3.4 UDP 协议的实现	(215)
11.3.5 TCP 协议的实现	(216)

第12章 文件共享

12.1 RFS 的功能	(219)
12.2 NFS 功能	(220)
12.3 一个 RFS 应用实例	(221)

附录 通讯和网络命令总汇	(227)
--------------------	-------

参考文献

第1部分 Ethernet/802.3 LANs

用户系统。

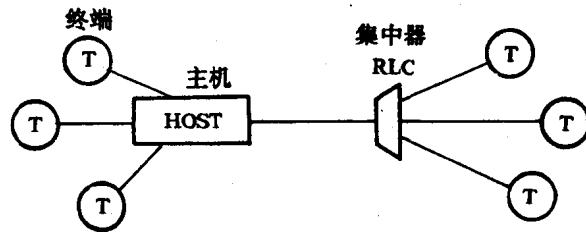


图 1-1 单处理机联机系统结构图

上述联机终端网络存在一些缺点：

(1)主机既要承担通信工作，又要承担数据处理，所以主机负荷重、效率低。

(2)通信线路利用率低，特别是在终端远离主机时，分散的终端都要单独占用一条通信线路，费用贵。

为了克服以上缺点，在主机和通信线路之间设置了前置机，专门处理与终端的通信工作，将通信功能从主机中分离出来，使主机能集中更多时间进行数据处理，在终端集中的区域设置集中器，大量终端先通过低速线路连到集中器上，集中器则通过高速线路与主机相连，见图 1-2。

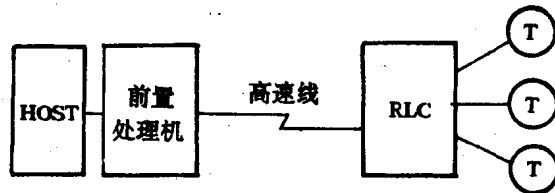


图 1-2 具有前置处理机的系统结构图

2. 计算机-计算机网络

计算机技术的发展和通信技术的进步使得多个单处理机联机终端网络互连起来，形成了多处理机为中心的网络。

利用通信线路将多个计算机连接起来一般有两种形式：第一种形式是通过通信线路将主计算机互连起来，主机既承担数据处理也承担通信工作；第二种形式是设置通信控制处理机 CCP (Communication Control Processor)，主机间的通信通过 CCP 的中继功能间接进行，把通信从主机中分离出来。由 CCP 组成的传输网络称通信子网，见图 1-3。

通信子网规模进一步扩大就成为社会公用的数据通信网，如图 1-4 所示。这种网络具有以下优点：允许异种机入网，兼容性好，通信线路利用率高。广域网及国家级的计算机网络大多采用这种形式。

3. 体系结构标准化网络

随着多种结构网络的出现，为了使网络的系统软件、网络硬件具有通用性，网络向标准化方向发展。按体系结构标准化形成的网络称为第三代网络。

(1)各计算机制造商网络体系结构标准化

1974 年，IBM 公司在世界上首先提出了完整的计算机网络体系标准化的概念，宣布了 SNA 标准 (System Network Architecture)，并以此为标准建立网络，称为 SNA 网。随后，DEC

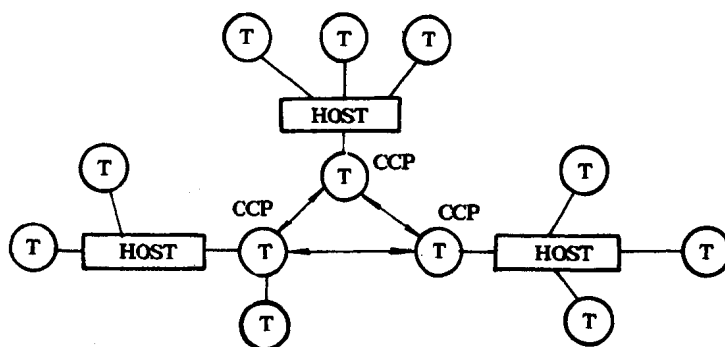


图 1-3 具有通信子网的计算机网络

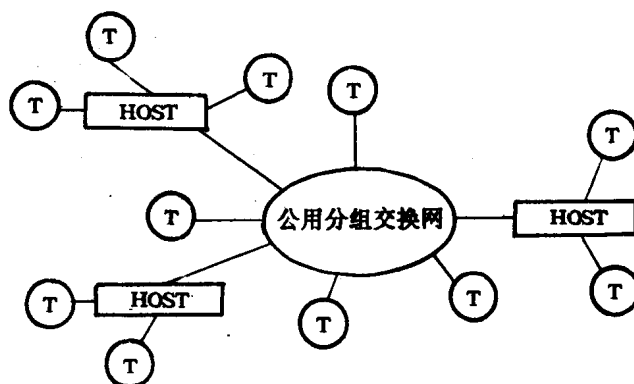


图 1-4 具有公用数据通信网的计算机网络

公司公布了 DNA(数字网络系统结构),Unive 公司公布了 DCA(数据通信体系结构),宝来公司公布了 BNA(宝来网络体系结构)等。

(2)国际体系结构标准化

1979 年,国际标准化组织(ISO)成立 TC97(计算机与信息处理标准化委员会)下属的 SC16(开放系统互连分技术委员会),制定了“开放系统互连参考模型”(OSI),以便于异种计算机互连构成网络。1980 年,美国电子电气工程师协会成立 IEEE802 局域标准委员会,制定了 IEEE802 标准,从而使局域网的发展走向标准化。

1.2 网络模型

1.2.1 ISO/OSI 网络体系结构研究方法

世界上不同年代、不同厂家、不同型号的计算机系统千差万别,将这些系统互连起来就要彼此开放。所谓开放系统就是遵守互连标准协议的实系统。实系统是由一台或多台计算机、有关软件、终端、操作员、物理过程和信息处理手段等的集合,是传送和处理信息的自治整体。采用抽取实际系统中涉及互连的公共特性构成的模型系统,然后研究这些模型系统,即开放系统

互连的标准。这样就可避免涉及具体机型和技术的实现细节,还可避免技术的进步对互连标准的影响。所谓模型化的方法就是用功能上等价的开放系统模型代替实开放系统。

模型化的方法按两步进行。第一步研究开放实系统的基本成分和涉及他们的组织及功能,构造开放系统互连的顺序式的七层模型,即应用层、表示层、会话层、传输层、网络层、数据链路层和物理层。第二步在七层模型的框架内详细和精确地描述抽象开放系统的功能,构成开放系统互连的服务和协议。ISO/OSI 七层模型如图 1-5 所示。

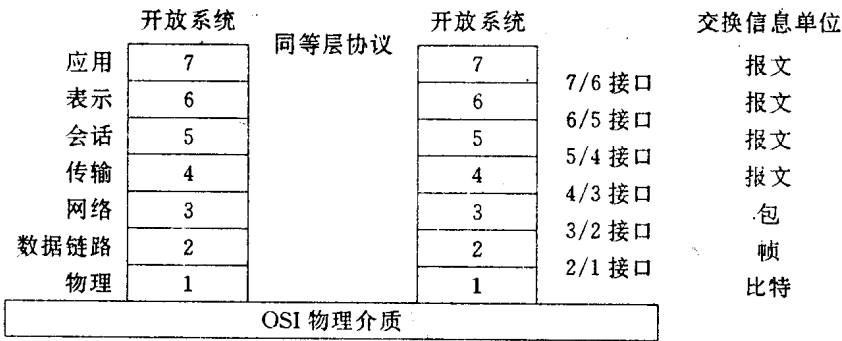


图 1-5 七层参考模型、协议和接口

OSI 参考模型是用来协调开发进程间通信的标准,它提供的仅是概念上和功能上的框架,为各层标准规范规定范围,并为各层标准的相容提供共同参考。

1.2.2 ISO/OSI 参考模型

开放系统互连基本参考模型把网络通信的体系结构分为七层,见图 1-5。这七层协议表示网络上的用户与用户要求通信时,其信息必须通过七层的内容所规定的处理之后,才能在物理介质上传输,以确保通信的可行性与通信的正确可靠性。

1. 分层的优点

OSI 参考模型是按分层的原则制定的,分层的概念在一些已有的通信协议中采用,OSI 继承了各种标准的这一特点,并把分层化作做得更加细致、合理。分层化的方法,为网络通信设计带来如下好处:

①简化了网络通信设计的复杂性。网络通信是非常复杂的一个过程,把整个过程划分为数个功能层次,这样一来使得整个结构功能完整,层间分工清晰,层内功能单一且易于实现,使整个系统的实现成为简单化的可行的过程,方便了设计。

②根据所完成的功能或技术上的明显不同来划分层次,每一层的功能是固定的和具体的,要求也比较明确。不同的网络可以根据自己具体的条件,采用不同的方法去实现某一层的功能。虽然采用的方法各异,但只要符合协议标准要求,各系统或终端就可以连接起来进行通信。

③由于采用了分层技术,在结构、硬件或软件方面出现了某些新技术或新方法的时候,很容易对某一层进行更新,以新的方法,新的技术取代老的方法与技术,易于实现技术的更新换代。

2. 分层的原则

在分层工作中必须解决把一个复杂的通信过程分成几层,如何分层以及把层次的边界定

在什么地方等问题。为解决上述问题,对分层提出了一些原则:

①对技术或处理上有明显不同的控制功能,应建立独立的层次来完成。

②对每一层建立它与相邻上、下层的边界及连接的接口。

③建立层次边界的地方应该是需要将相应接口标准化且通过边界的交互信息量最少的地方。

④对容易局部化的功能建立一层,以便将来出现新技术、新方法时,使得该层可以整体地重新设计,而不改变相邻层之间的关系。

⑤允许在一个层内改变功能和协议,而不影响其它层。

⑥把功能相似的集中到同一层。

⑦对过去已被经验证明是成功的层应保留。

⑧在同一层内,根据通信服务需要,可建立若干功能子层。

3. OSI 模型各层功能

①物理层,用来实现网内两实体间的物理连接,按比特串行传送数据信息。

②数据链路层,用以建立相邻结点之间的数据链路,传送按一定格式组织起来的比特信息,即数据帧。

③网络层,接收来自源机的报文,把它转换成报文分组(具有地址标识和网络协议信息的格式化信息组),送到指定的目标机。

④传输层,提供一种独立于通信子网的数据传输服务,在网内两实体间建立通信信道,用以传输信息。

⑤会话层,在两个合作的实体之间建立通信关系,组织和同步它们的会话,并为管理它们的数据交换提供必要的手段。

⑥表示层,处理实体在通信时或者要传送时,对引用信息的表示进行转换,以消除网内实体间的语义差异。

⑦应用层,是 OSI 参考模型的最高层,负责两个应用进程之间的通信,为网络用户间的通信提供专用程序包。

4. 层与层之间的关系

层与层之间的关系是上下连接的关系。下层对上层提供服务,每一层都利用下一层所提供的服务实现本层的功能,并为上层提供服务。信息传输在发送过程中是从高向低层逐层地传递,在接收过程中则是从低层向高层传送。

1.2.3 客户、服务器模型

决大多数网络应用的关系是属于客户——服务器模型。这个模型包括三个部分:客户、服务器和服务器提供的服务。

服务器是这样的一个计算机,它为客户提供服务。它随时等待着客户的请求以便为客户提供服务操作。客户机向服务器提出服务请求,并传回服务的结果。一般的服务有文件传送、电子邮件、远程命令的执行和把当天时间返回给客户进程等。

客户——服务器的运行模式一般为:服务器上的进程初始化自身,开始执行服务进程,并进入睡眠状态以等待客户进程的请求。在与网络服务器相连的系统上,若某一客户进程正在执行,当需要服务器提供服务时,客户进程通过网络把服务请求送给服务器进程,服务器就为客户

户服务,并回送结果。服务完之后,服务器又进入睡眠状态等待下一客户请求。

服务器可以根据服务的方式分为两类:

第一类:重复服务器(iterative server)。这种服务器上的进程能够预知在短时间内可以处理完请求,那么便亲自处理它。典型的服务是通报时间。

第二类:并发服务器(concurrent server)这种服务器上的进程事先不能预知完成服务请求需要花多长时间,便以并发的方式处理它。并发服务器通过激活别的进程来完成服务请求,而本身转入睡眠状态以等待另一个请求的到来。典型的服务是处理文件信息,例如对文件的访问和打印文件。

客户与服务器的地位不是永久不变的,当一个计算机提供文件传送服务时,这个计算机就是一个服务器;当一个计算机允许用户访问另一个机器上提供的文件传送服务时,这个计算机就是一个客户机。

客户、服务器通信必须要经过以下的几个步骤:

①当客户提出服务请求时,必须向服务器提供传送地址。这就像打电话时,必须有对方的电话号码。

②在客户机想与服务器通信之前,服务器要能够获得确切的传送地址。

这里涉及两个概念,一个是传送地址,一个是网络中数据的传送。为了获得传送地址,每个服务器均有一个名称,名称和传送地址有一个映射关系,有专用的名称到地址的映射机制,负责把服务器和服务的名称映射为服务器上服务处理程序的入口地址。

网络应用中数据的传递是由传送提供者完成的,传送提供者是一个能够接收网络信息并把它传送到远程机器上的软件。

在 UNIX 系统中,传送提供者是传送层协议的实现。UNIX 系统有三种常用的传送提供者:TCP/IP、XNS 以及 OSI 的传送协议。其中最常用的是 UNIX 的 TCP/IP。

1.3 计算机网络的组成

计算机网络从逻辑功能上看可分为资源子网和通讯子网。资源子网提供访问网络和处理数据的功能,由主计算机系统、终端控制器和终端组成。通讯子网提供网络的通讯功能,它由网络结点、通讯链路和信号变换设备等组成。

主机:它是资源子网的关键设备,可以是微型机、小型机、中型机或大型机。它有以下功能:首先为本地用户使用本机资源提供服务功能;其次,为本地用户访问网络其它结点设备以及共享其它主机资源提供必要的通讯软件和分布处理的能力;另外,还可为网上其它用户或主机共享本机的资源提供服务。

终端设备:它是计算机网络面向用户的窗口。终端可作为主机的配置连到主机,通过主机的网络访问软件来访问网络并和其它用户通讯。终端的类型较多,计算机网络在应用层用虚拟终端协议来适配各类终端,每个实际的终端和虚拟终端之间进行协议的转换。

分组组装/拆卸设备:它是为终端直接入网而提供的设备。它的功能是对终端输入输出字符进行组装和拆卸,以使其能按通讯子网的分组格式要求组成信息单位在网络内传送。

通讯控制处理机(CCP):也称通讯处理机。它是一种在数据通讯系统或计算机网络中处理通讯控制功能的专用计算机,为主机或用户终端提供中继转发功能,是通讯子网中最主要的组

成部件。

网络控制中心:管理整个网络的运行,系统故障检测和恢复,计费,用户登录及网络运行情况分析等。

远程线路集中器:它是多终端共享线路的一种形式,可把一群终端集接,然后再用高速线路连到通讯子网或主机上,终端也可通过集中器入网。

信关:它是网络互连设备,通过信关可将多个不同类型的网络互连成网际网,把一种协议转换成另一种协议。在 TCP/IP 协议中,信关表示一个网络级的路径选择器。有时把信关用于描述在网络层以上的层次上实现特定转换的软件。例如,把一种格式的电子邮件转换成另一种格式电子邮件的邮件信关。

通讯线路和通讯设备:如调制解调器、多路复用、交换设备以及主机配置的前端机等。其中为了节省通信费用,将多条信道复用在一条物理线路上,这种技术称为多路复用技术。在网络中许多层次都可以进行多路复用。一般多路复用技术分为两类:频分多路复用和时分多路复用。

频分多路复用:将具有一定带宽的线路划分为若干条较小带宽的信道,每条信道归一个用户使用。

时分多路复用:把时间划分为若干时间片,每个用户占用一个时间片,在用户的时间片内,该用户可使用信道的全部带宽。

与多路复用相反的是多路分解。上层软件从其下层接收数据时,它必须能够识别接收数据的用户进程,这就是多路分解。例如,当 UDP 软件从 IP 层接收数据时,它必须识别接收数据的用户进程,UDP 和 TCP 均使用一个 16 位的端口来识别用户进程。

网络系统软件:网络系统软件主要由以下几部分组成:

服务器平台,主要指服务器操作系统,是网络的核心。其中主要包括网络文件系统、存储管理以及任务调度。

网络服务软件,在服务器平台上运行,提供客户/服务器应用环境的软件。

传输协议软件等。

网卡:也称网络适配器。用于文件服务器、工作站等智能设备。网卡电路负责协议的产生与检测,支持对应的网络类型。

网桥:它在数据链路层上实现,负责把接收到的包内的部分信息从一个网络复制到另一个网络。

路由选择器:负责把信息从一个网络传送到另一个网络,并且确定信息分组的传送路径。它在网桥功能的基础上增加了路径选择功能。

1.4 服务方式

计算机网络是一个极为复杂的系统,为了减少设计的复杂程度,通常把计算机网络的功能分为若干层次,较高层次建立在较低层次的基础上,又为其更高层次提供服务功能。因此,每一层均向上一层提供服务。在这里主要介绍传送层给用户应用程序提供的服务。服务类型一般有:面向连接、非连接、全双工及半双工、顺序、流控制、差错控制、字节流服务等。

1.4.1 面向连接的服务和非连接的服务

在网络中,主机与主机上的应用程序之间的通信一般有如下过程,源主机先与其通讯处理机通信,这个通讯处理机又与某个中间节点通信,再经过若干个中间节点,最后到达目的节点,目的节点再与目的主机通信。也就是说,由于通信子网的支持以及网络层向传输层提供的服务支持,主机才能够与主机通信。那么,什么样的服务来支持这种通信过程呢?一般有两种形式的服务类型:面向连接的服务及面向非连接的服务。

面向连接的服务要求在两个应用程序通信前先建立两者之间的逻辑连接。一般用虚电路来描述这种服务,对两个应用程序来说,数据的传输实际上是通过分组交换网络进行的,但好像在它们之间存在一条专用线路为之服务。面向连接的服务一般包括三个过程:建立连接、传输数据和断开连接。

面向非连接的服务也称数据报服务。数据报含有全称编址,独立地从一个系统传送到另一个系统,最终到达目的节点。这类服务不用事先建立逻辑连接,因此,也不需要建立连接、传输数据和断开连接三个通信过程。信息包在网络中传输,是否丢失网络是不知道的,而且,接收的顺序和发送的顺序不一定一样。所以,它提供不可靠的传输服务,由主机对信息包的传输提供差错、重复和丢失管理。这一点与信件投递服务相似。信件发出后,是一封封独立投递的,收信人收到信件的顺序和发信人发信的顺序不一定一样。在投递的过程中是否丢失,邮局也无法知道。

1.4.2 全双工与半双工连接

半双工信道是指数据可以沿信道的两个方向传送,但在同一时刻,只能沿一个方向传送,如图 1-6 所示。全双工信道是指数据能同时沿信道的两个相反方向同时传输,如图 1-7 所示。半双工通信采用半双工信道,全双工通信采用全双工信道。例如:TCP/IP 协议是全双工通信。

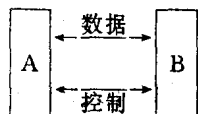


图 1-6 半双工通信

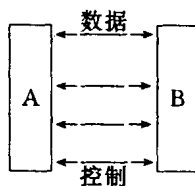


图 1-7 全双工通信

1.4.3 流控制

为了获得较高的信息传输速率,远距离的通信双方必须在收发数据的速度上保持一致。以维持信息的发送方发送数据的速率与接收方处理数据的速率相协调,保证不丢失传送的数据,称为流控制。

1.4.4 差错控制

由于通信线路的固有特性,不可避免地要产生差错。差错控制保证接收方应用程序所收到的数据是无差错的。为了处理在网络中可能的数据丢失情况,发送方发送数据之后,启动计时

器计数,如果在约定的时间段内还未收到接收方发回的应答信息,就认为该数据包已丢失,发送方重新发送该数据包。

1.4.5 字节流

字节流服务对数据流不提供任何记录边界划分。例如 TCP/IP 就是一个字节流协议。

1.5 同步方式

同步方式是指接收端要按照发送端的每个码元的重复频率以及起止时间来接收数据,也就是要在时间基准上保持一致。在通信时,接收端要校准自己的时间和重复频率,以便和发送端取得一致。

一般有以下几种同步方式。

位同步:使接收端接收的每一位数据信息都与发送端准确地保持同步。

字符同步:从正确接收的位流中区分出字符边界,准确地实现以字符为单位的同步。

1.6 计算机网络基本术语

1.6.1 地址

为了在网络中标识通信的两个系统上的进程,需要进行如下寻址:

①指定一个特定网络,因为一个主机可能与多个网络相连。

②主机地址,网络上的一个主机应有一个唯一的标识符。通常主机地址由网络号及在该网络上的主机号组成。

③进程标识,一个主机上的每个进程都有一个唯一的进程号。例如:TCP 和 UDP 均使用 16 位的端口号标识用户进程,TCP/IP 协议组用 32 位整数标识网络号和主机号。

1.6.2 连接与相关

连接是指两个进程之间的通信链路,相关是指组成一个连接的两个进程的元素组。如:

{协议,本地地址,本地进程,外部地址,外部进程}

其中,本地地址是指本地主机的网络号和主机号;外部地址是指外部主机的网络号和主机号;本地进程和外部进程是指连接的两个系统上的特定进程。例如:在 TCP/IP 协议集合中,一个连接的元素组定义如下,协议类型为 TCP;本地地址为 192. 43. 235. 7,本地进程是 1500,外部地址为 192. 43. 235. 6,外部进程为 31。

1.6.3 打包与拆包

当用户发送的数据进入网络模型中的各个不同的层次时,每个层次都要加上有关的信息,这个过程称为打包。相反,当用户接收数据时,其网络的各个层次自底向上都要去掉相关层的控制信息,最后只把用户需要的数据递交给接收用户,这个过程称为拆包。