21世纪流行软件丛书

快易通

Linux网络程序设计

Wizard创作室 策划

唐礼勇
郭志峰 编著
张云霞

北京大学出版社

21 世纪流行软件丛书

快易通 Linux 网络程序设计

Wizard 创作室　策划
唐礼勇　郭志峰　张云霞　编著

北 京 大 学 出 版 社
北　　京

内 容 提 要

本书从不同层次，系统地介绍了在 Linux 环境下进行网络程序设计的方法，比较全面地覆盖了 TCP/IP 环境下网络编程的各个方面。本书面向具有 C 语言和网络基础的读者，可作为 Linux 网络程序设计的教材，也可供工程技术人员参考。对于其他 UNIX 系统，本书也同样具有参考价值。

图书在版编目(CIP)数据

快易通 Linux 网络程序设计/唐礼勇，郭志峰，张云霞编著. —北京：北京大学出版社，2001.6
(21 世纪流行软件丛书)
ISBN 7-301-02135-6

Ⅰ. 快… Ⅱ. 唐… Ⅲ. Linux 操作系统 Ⅳ. TP316.89

书　　名：**快易通 Linux 网络程序设计**
著作责任者：唐礼勇　郭志峰　张云霞
责 任 编 辑：沈承凤
标 准 书 号：ISBN 7-301-02135-6/TP·176
出　版　者：北京大学出版社
地　　址：北京市海淀区中关村北京大学校内　100871
网　　址：http://cbs.pku.edu.cn
电　　话：出版部 62752015　发行部 62754140　编辑部 62752038
电 子 信 箱：zpup@pup.pku.edu.cn
排　版　者：兴盛达打字服务社　62549189
印　刷　者：北京市银祥福利印刷厂
发　行　者：北京大学出版社
经　销　者：新华书店
787 毫米×1092 毫米　16 开本　14.125 印张　349 千字
2001 年 6 月第 1 版　2001 年 6 月第 1 次印刷
定　　价：23.00 元

21 世纪流行软件丛书

前　言

时代的列车已经驶入了 21 世纪——一个信息时代的世纪，每一个希望成为时代弄潮儿的热血青年，都应把握住时代的脉搏，学习和掌握体现这个时代特点的技术。软件技术作为信息时代的核心技术之一，不仅仅是专业的软件开发人员需要掌握的，而且对于广大的软件爱好者也是需要学习和了解的。

一般来说，学习软件知识有两种方式，一种是偏重理论知识的学习，另一种则偏重应用和操作技能的学习，它们各有优缺点。前一种强调"为什么这样做?"，适合学校中的系统学习；后一种强调"怎样做?"，主要适用于社会上的广大软件爱好者和软件应用人员。但这两种方式的优缺点是互补的，即前一种可容易地做到举一反三；后一种可率先做到功能的使用，易于收到立竿见影的效果。

"快易通"系列丛书融会贯通了这两种学习方式的优点，并且偏重于后者，即在介绍基本内容的基础上，在一个重点讲述的知识点后面加上一个或多个非常典型的实例，再加上精彩的理论反思例子。读者可以从本系列图书中，既获得理论方面的知识，也学到非常实用的技能。

本系列图书的读者对象为初中级用户。由于本书内容的精练性和实用性，本丛书也特别适用于各类培训班选作培训教材。本系列图书的特点是：

1. 选题范围广

本丛书以适应信息技术大众化的要求为主要目的，突出实用化、系列化、大众化。故采取开放式选题，即选题面向不断发展着的计算机技术应用的实际需要和国际上使用的新技术，选题不断增加又保持前后有序。以经典流行软件为主，但同时兼顾一些应用面较窄但技术先进、有前途的新软件。对于兼有中西文版本的软件，详述中文版，以全力满足中国用户的需要。

2. 充分体现"快易通"的特点

本系列图书能够满足广大读者都希望容易地掌握、贯通所有的知识点。

快：提炼知识点，使广大读者能快速入门，现学、现会、现用，掌握纲要和总体把握知识点。用简单精炼的语言介绍知识点，给初学者一个全貌，明白知识点的内容。这部分内容也可以作为老用户的速查手册。

易：容易掌握。这是继"快"后面的进一步提高，以贴切的例子来教读者掌握知识点，应用知识点。如果不能就本知识点给出合适的例子，那么就用提问的方式来深化，提问切中知识要点，能带动读者深入思考，轻松掌握本知识点。

通：知识点串联。综合应用知识的介绍与例子，串联多个知识点。

相信本套丛书一定成为广大软件爱好者的良师益友，成为您在新世纪中工作学习的好帮手。

策划者

2000 年 1 月

前　言

最近几年来，在计算机界有两件事情的发展是值得大书特书的。首先当属 Internet，另外一件应轮到自由软件运动，而 Linux 操作系统同时在这两件事情中都占据了显著的地位。大量的厂商和投资都已经被吸引到 Linux 以及 Internet 上来。另外，随着开发人员的不懈努力，Linux 系统在可用性方面也不断地得到加强，走到个人桌面上的日子指日可待。

这是一本同时涉及 Linux 和网络的教材，讲解如何在 Linux 操作系统下开发网络程序。

我们将课程的学习分为循序渐进的五个部分。

第一部分是“概论”，包括 Linux 的历史及其特征、TCP/IP 协议及 Linux 网络环境简介、常用 Linux 系统调用介绍。了解 UNIX 系统原理、有网络基本知识的读者可跳过本章。

第二部分是“基础篇”，首先介绍了 Linux 网络编程中最基本的概念——Socket（套接字）接口，接着介绍基本数据结构，以及 TCP 和 UDP 编程如何用 Socket 接口来实现。这一部分还讨论了不同模式的客户/服务器模型。通过对几个简单示例程序的详细分析，使读者了解系统核心是如何处理 TCP 和 UDP 协议的，同时也有助于读者分析并解决以后程序设计中所碰到的问题。

第三部分是“中级篇”，首先分析不同的 I/O 模型，给出实现 I/O 复用的简单实例；接下来介绍套接字选项处理函数及数据结构，通过套接字选项，您可以对套接字提供更为精细的控制。这一部分的最后讨论 Linux 下守护进程的实现方法，对于开发服务器程序来说，掌握守护进程的实现方法是必须的。学习完“基础篇”后的读者，如果您还需要提高自己所写的程序的性能，这一部分会对您有所帮助的。

第四部分是“提高篇”，包括 UNIX 域协议、线程的概念及其应用、非阻塞式 I/O 示例、信号驱动 I/O 等内容。通过 UNIX 域协议，您可以使用同样的 Socket 接口，开发既具有 TCP/IP 协议的优点，而又不带来本地应用速度损失的程序。由于省去了进程切换的开销，线程可以进一步提高您的系统的性能。

最后一部分是“高级篇”，包括原始套接字接口、广播与多播、数据链路访问。使用原始套接字接口，可以绕开系统所提供的 TCP/IP 协议栈，直接构造自己的 IP 数据包。广播及多播技术在视频点播、网络会议等方面有广泛的应用前景。数据链路访问接口提供直接访问数据链路层的手段，可用于实现网络协议分析。

本书给出了大量的示例程序，可作为您在开发系统中的参考。

学会熟练应用两个技巧，可以使您永远能跟上系统的发展。其中之一是 man，它可以向您提供您正在使用的系统的信息；另外一个是直接去检查/usr/include/netinet 下的头文件，您可以在手头缺少资料时直接查到系统数据结构的细节。

衷心感谢北大出版社的沈承凤老师，作为本书的责任编辑，她付出了很多的心血。

编著者

2001 年 5 月

目　录

第一部分　概　论

第二部分　基 础 篇

第三部分　中　级　篇

第四部分　提　高　篇

第五部分　高　级　篇

第一部分　概　　论

第 1 章　什么是 Linux?

1.1　Linux 的简介

Linux 在最近几年取得了空前的成功。它出色的性能成了 Windows 2000 服务器有力的竞争对手。到 2000 年初，它已占据了服务器操作系统 25%的市场份额。另外，由于 Internet 的发展，计算机的应用格局也发生着改变。过去，PC 上聚集了大量的应用软件，在 Internet 的时代，越来越多的却是与网络、服务器紧密相连。可以预见，将来用户将使用各种各样的小型无线或便携式的“Internet 浏览设备”乃至信息家电来获取众多 ASP 在 Internet 上提供的各种信息和服务。在这些小型设备上，Linux 稳定、经济、占有资源小等特点为其广阔的应用前景奠定了坚实的基础。

那么，究竟什么是 Linux 呢？现在对它的历史作一简单介绍。

Linux 与 UNIX 有着密不可分的关系。1969 年，Bell 实验室的 Ken Thompson 和 Dennis Richie 一起在一台 PDP-7 计算机上开发了一种多用户、多任务操作系统，就是著名的 UNIX。后来 UNIX 很快成了众多操作系统的佼佼者。Linux 其实是一种类 UNIX 的操作系统。芬兰赫尔辛基大学的 Linus Torvalds 是 Linux 的创始人、作者及主要维护者。他以自己所熟悉的 UNIX 作为原型，在一台 Intel 386 PC 上开始了编写工作。由于进展很快，受到工作成绩的鼓舞，他将这项成果通过互联网与其他人共享。有人看到了这个软件并开始分发。每当出现新问题时，就会有人去寻找解决办法并加入 Linux 中。很快地，Linux 成为了一个真正可用的操作系统。值得注意的是 Linux 并没有包括 UNIX 源码，它是按照公开的 POSIX 标准重新编写的。

POSIX 意为“基于 UNIX 的可移植操作系统接口”。它是由电气与电子工程师学会(IEEE)开发的一个标准，同时也是 ISO 和 IEC(国际电工委员会)采纳的国际标准。它主要说明基于 UNIX 内核的操作系统 C 语言接口，例如进程原语、进程环境、文件与目录、终端 I/O、系统数据库，以及 tar 和 cpio 归档格式等等。如果想了解 POSIX 标准化工作或当前版本，可以访问http://www.pasc.org/standing/sdll.html，以获取相应信息。

可以说，Linux 是一种操作系统的内核。它是一组源代码开放的程序集，由 1000 多个文件组成。它十分精巧，总体代码可以压缩得很小，从而适于各类嵌入式电子设备。在系统结构上，它采用了传统的单体(monolithic)内核结构，没有采用较复杂的“微内核”结构，可以高效运行，节省资源。

按照一定规律，每隔一段时间就会发布一个 Linux 版本，供全世界开发者测试及使用。版

本序号为奇数者是开发版，序号为偶数者为阶段性发布的稳定运行版。当前，最新的稳定内核版本是 2.4.4。

Linux 发展到现在，不仅有着稳定强大的内核，同时还有着完善的基础软件和丰富的应用程序。Linux 系统中常用的系统程序大部分是由 GNU 计划(www.gnu.org)开发出来的，有不少机构或个人利用自己闲暇时间，不计报酬地为 Linux 开发应用程序，这些程序大多是自由软件，任何人都可以免费地在网络上取得。

Linux 具有 UNIX 系统的程序接口和操作方式，也继承了 UNIX 稳定高效的特点。网络上安装 Linux 的主机连续运作一年以上而不死机、不必关机是很平常的事，不过 Linux 却不像一般 UNIX 需负担庞大的版权费用，而且要在专属的昂贵硬件上才可以使用。现在，它对硬件支持越来越广，性能也越来越好，使个人计算的功能可以和中级工作站，如 Sun 公司的 SPARC 等相匹敌。

Linux 的商业发行版实际上是由特定版本的 Linux 内核、属于 GNU 体系功能性支撑模块和一些运行于 Linux 上的商用软件所集成的。它的产生主要是因为每个人去下载程序再一一安装非常不便，于是便有某些有系统整合能力的公司去搜集、整合 Linux 上的程序，把“内核－系统程序－应用程序”整合起来构成一个完整的操作系统，让一般用户可以简便地安装完整的系统，这就是所谓的“安装套件”(distribution)。这些公司又称为 Linux 发行商，他们并不拥有其发行版中各软件模块的版权，而是拥有发行套件的品牌价值，以含于其中的集成系统的质量和相关特色服务进行市场竞争。

我们一般说的 Linux 系统便是针对这些安装套件而言。Linux 安装套件的种类繁多，著名的有早期非常流行的 Slackware 套件，还有近来越来越多的人使用的 Debian 以及在国际 Linux市场占有率超过一半的 RedHat Linux。这些不同的安装套件都算是 Linux 系统，同样都用 Linux 内核，收录的程序大同小异，相互间的程序都可以共享，不同的地方只在一些系统设置跟程序套件的管理方式而已。

同样是 Linux 系统，却分成不同公司、机构整合出来的不同安装套件，这就是大家常常在网络上看到 Linux 有那么多“种”的原因。

1.2　Linux 的发布版本

不同的 Linux 版本有不同的内核版本，其中收录不同的工具、实用程序和驱动程序模块，并配有自己的安装、升级程序。下面简单介绍一下用得最为广泛的发行套件。

Red Hat：由 Red Hat Software 公司发行，其标志为戴红帽子的小企鹅。Red Hat Linux 支持多种硬件平台，安装与系统管理界面也非常友好，软件包以 RPM 的方式升级，收集了 GNU、ShareWare 等丰富的软件包。并且 Red Hat Linux 还配有详细完整的联机文档。大家可以参考以下网址：

WWW：http://www.redhat.com

ftp：ftp://ftp.redhat.com

Slackware：由 Walnut Greek CDROM 公司正式发布。Slackware Linux 是较早的流行版本，现在仍然非常普及。Slackware 的目录结构相对简单，配置文件较清楚。可参考以下网址：

WWW：http://www.cdrom.com/titles/os/slack96.htm

FTP：ftp://ftp.cdrom.com.

Debian Linux：是由网络上的 Linux 热衷者负责维护的发行版本。Debian 有与 Red Hat 版本相似的功能，它的包管理系统提供一千多个软件包。它也采取 Red Hat 的 RPM 类似的升级方式，有着良好的安全特性。可参考以下网址：

WWW：http://www.debian.org/

FTP：ftp://ftp.debian.org/debian/

另外，还有 Caldera 的 OpenLinux 和 S.u.S.E. Linux 等等其他的英文 Linux 版本。

在我国，在准确把握 Linux 开放性的协作开发模式及 Linux 发行版经营活动的特性方面，也有所突破。现就国内 Linux 版本作一简单介绍。

中软 Linux(COSIX Linux)：由中软总公司发布。中软 Linux 有实现 B1 安全级别的面向安全需要的版本，并且配合相应的网络安全产品和其他安全软件；中软 Linux2.0 企业版适用于网络服务器，可为企事业单位提供廉价和稳定的系统；中软 Linux2.0 标准版是简单易用的面向广大计算机爱好者的版本。

红旗 Linux：由中国科学院软件所发布。其 Linux PC 服务器版集成了字处理、办公套件软件；在高端，推出了 16 个 PC 机组成的 Cluster 群集系统，可以作大规模并行处理使用。

1.3　Linux 的特性

Linux 操作系统在短短的几年之内得到了非常迅猛的发展，这与 Linux 具有的良好特性是分不开的。Linux 包含了 UNIX 的全部功能和特性。简单地说，Linux 具有以下主要特性：

1. 开放性

开放性是指系统遵循世界标准规范，特别是遵循开放系统互连(OSI)国际标准。凡遵循国际标准所开发的硬件和软件，都能彼此兼容，可方便地实现互连。

2. 多用户

多用户是指系统资源可以被不同用户各自拥有使用，即每个用户对自己的资源(例如文件、设备)有特定的权限，互不影响。Linux 和 UNIX 都具有多用户的特性。

3. 多任务

多任务是现代计算机的最主要的一个特点。它是指计算机同时执行多个程序，而且各个程序的运行互相独立。Linux 系统调度每一个进程平等地访问微处理器。由于 CPU 的处理速度非常快，其结果是，启动的应用程序看起来好像在并行运行。事实上，从处理器执行一个应用程序中的一组指令到 Linux 调度微处理器再次运行这个程序之间只有很短的时间延迟，用户是感觉不出来的。

4. 良好的用户界面

Linux 向用户提供了两种界面：用户界面和系统调用。Linux 的传统用户界面是基于字符的命令行界面，即 shell。shell 有很强的程序设计能力，用户可方便地用它编制程序，从而为用户扩充系统功能提供了更高级的手段。

系统调用给用户提供编程时使用的接口。用户可以在编程时直接使用系统提供的系统调用命令。系统通过这个界面为用户程序提供低级、高效率的服务。

Linux 还为用户提供了图形用户界面。它利用鼠标、菜单、窗口、滚动条等设施，给用户呈

现一个直观、易操作、交互性强的友好的图形化界面。

5. 设备独立性

设备独立性是指操作系统把所有外部设备统一当作文件来看待，只要安装它们的驱动程序，任何用户都可以像使用文件一样，操纵、使用这些设备，而不必知道它们的具体存在形式。

具有设备独立性的操作系统，通过把每一个外围设备看作一个独立文件来简化增加新设备的工作。当需要增加新设备时，系统管理员就在内核中增加必要的连接。这种与外设连接的方式(也称作设备驱动程序)保证每次调用设备提供服务时，内核以相同的方式来处理它们。当新的及更好的外设提供给用户时，操作系统允许在这些设备连接到内核后，就能不受限制地立即访问它们。设备独立性的关键在于内核的适应能力。其他操作系统只允许一定数量或一定种类的外部设备连接。而设备独立性的操作系统能够容纳任意种类及任意数量的设备，因为每一个设备都可通过其与内核的专用连接独立进行访问。

Linux 是具有设备独立性的操作系统，它的内核具有高度适应能力。随着更多的程序员加入 Linux 编程，会有更多硬件设备加入到各种 Linux 内核和发行版本中。另外，由于用户可以免费得到 Linux 的内核源代码，因此，用户可以修改内核源代码，以便适应新增加的外部设备。

6. 丰富的网络功能

完善的内置网络是 Linux 的一大特点。Linux 在通信和网络功能方面优于其他操作系统。其他操作系统不包含如此紧密地和内核结合在一起的连接网络的能力，也没有内置这些联网特性的灵活性。而 Linux 为用户提供了完善的、强大的网络功能。

支持 Internet 是其网络功能之一。Linux 免费提供了大量支持 Internet 的软件。Internet 是在 UNIX 领域中建立并繁荣起来的，在这方面使用 Linux 是相当方便的，用户能用 Linux 与世界上的其他人通过 Internet 网络进行通信。

文件传输是其网络功能之二。用户能通过一些 Linux 命令完成内部信息或文件的传输。

远程访问是其网络功能之三。Linux 不仅允许进行文件和程序的传输，它还为系统管理员和技术人员提供了访问其他系统的窗口。通过这种远程访问的功能，一位技术人员能够有效地为多个系统服务，即使那些系统位于相距很远的地方。

7. 可靠的系统安全措施

Linux 采取了许多安全技术措施，包括对读、写进行权限控制、带保护的子系统、审计跟踪、核心授权等，这为网络多用户环境中的用户提供了必要的安全保障。

8. 良好的可移植性

可移植性是指将操作系统从一个平台转移到另一个平台使它仍然能按其自身的方式运行的能力。

Linux 是一种可移植的操作系统，能够在从微型计算机到大型计算机的任何环境中和任何平台上运行。可移植性为运行 Linux 的不同计算机平台与其他任何机器进行准确而有效的通信提供了手段，不需要另外增加特殊的和昂贵的通信接口。

第 2 章　TCP/IP 协议及 Linux 网络常用命令

TCP/IP(Transmission Control Protocol/Internet Protocol)作为整个 Internet 运作的核心,在讨论网络编程的时候,当然地成为我们首要关注的概念。

网络和 Linux 有着密不可分的关系。从某种意义来说,Linux 是一个针对 Internet 和 WWW(World Wide Web)的产品。它的开发者和用户用 Web 来交换信息思想、程序代码,而 Linux 自身也常常被用来支持各种组织机构的网络需求。这一章讲的是 Linux 如何支持如 TCP/IP 等网络协议。

TCP/IP 协议最初是为支持 ARPANET(一个美国政府资助的研究性网络)上计算机通信而设计的。ARPANET 提出了一些网络概念如包交换和协议分层(一个协议使用另一个协议提供的服务)。ARPANET 于 1988 年隐退,但是它的继承者(NSFl NET 和 Internet)却变得更强大了。现在我们所熟知的万维网 WWW 就是从 ARPANET 演变过来的,它本身就是基于 TCP/IP 协议的。UNIX 被广泛应用于 ARPANET,它的第一个网络版本是 4.3 BSD。Linux 的网络实现是以 4.3 BSD 为模型的,它支持 BSDsockets(及一些扩展)和所有的 TCP/IP 网络。选这个编程接口是因为它很流行,并且有助于应用程序从 Linux 平台移植到其他UNIX平台。

2.1　TCP/IP 网络简介

TCP/IP 作为事实上的网络体系结构和协议标准,它也是将复杂的通信功能分成独立的层次,每一层与邻接的层次通信,上层功能来自底层支持,如图 2.1 所示。

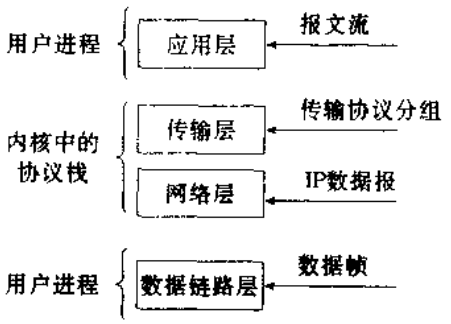

图 2.1　TCP/IP 协议分层

1. 应用层(application)

这一层包括所有利用传输层协议发送数据的进程和向用户提供的一组常用应用程序,如文件传输、电子邮件等。如:FTP(文件传送协议)、TELNET(网络终端协议)、SMTP(简单邮件传送协议)、DNS(域名解析)、RIP(寻径信息协议)、NFS(网络文件系统)等。

2. 传输层(TCP、UDP)

这一层提供应用程序间端到端的通信。包括:传输控制协议 TCP,提供有连接的数据传输服务;用户数据报协议 UDP,提供高效、无连接数据报传送服务。

3. 网络层(IP)

这一层又称网际网层，提供不同网络的主机间的数据传送能力。完成包括管理 Internet 中的地址、路由选择、数据报分片重组等工作。

4. 数据链路层

这一层包括各种物理网络，如 Ethernet、Token Ring 等等。链路层提供了 TCP/IP 与各种物理网络的接口，为数据报的传送和校验提供了可能。

TCP/IP 是由众多的协议组成的协议族，如图 2.2 所示。

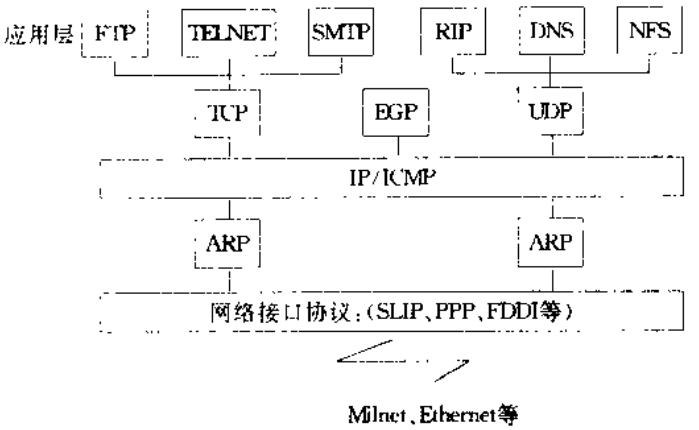

图 2.2　TCP/IP 协议族

下面就针对上图中的主要协议作一简单说明。

在 IP 网络中，每台机器都有一个 IP 地址——一个 32 位的数字，它唯一地标识这台机器。WWW 是一个非常巨大并且迅速增长的网络，连在它上面的每台机器都必须有一个独立的 IP 地址。IP 地址由四个用点分开的数字表示，如 10.42.0.9。这个 IP 地址实际上分成两个部分：网络地址和主机地址。每部分的长度是可以变化的(有好几类 IP 地址)。以 10.42.0.9 为例，网络地址是 10.42，主机地址是 0.9。主机地址又进一步分为子网地址和主机地址。还是以 10.42.0.9为例，子网地址是 10.42.0，主机地址是 10.42.0.9。这种划分可以允许某部门划分他们自己的子网络。例如，如果 10.42 是 virtual 计算机公司的网络地址，则 10.42.0 可能是研发部门子网，10.42.1 可能是行政部门子网。这些子网可以是分别建立的，可能租用电话线或用微波进行相互间通信。IP 地址由网络管理员分配，用 IP 子网可以很好地管理网络。IP 子网的管理员可以自由分配子网内的 IP 地址。

通常，IP 地址是比较难记的，而名称则容易多了，像 linux.virtual.com.cn 就比10.42.0.9 要好记一些。但是必须有一些机器来将网络名称转变为 IP 地址。这些名称可以静态地定义在 /etc/hosts 文件中，也可以由 Linux 请求域名服务器(DNS)来解析它。后一情况下，本地主机必须知道一个或一个以上的 DNS 服务器并且这些服务器要将其名称指定到/etc/resolv.conf 中。

当你想要与另一台计算机连接时，比如说想阅读一个 Web 页，你的 IP 地址就会被用来与那台机器交换数据。这些数据被包含在一些 IP 包中，每个 IP 包都有一个 IP 头用来包含源机器的 IP 地址和目的机器的 IP 地址、校验和以及其他有用的信息。IP 包的校验和用来让 IP 包

的接收端判断 IP 包是否在传输过程中发生错误，如由于通信线路故障而引起的错误等。应用程序想要传输的数据可能被分成很多个容易处理的小包。IP 数据包的大小是根据传输媒体的变化而不同的；以太网包通常比 PPP 包要大一些。目的主机在将数据送给接收端应用程序前需要将这些包重新拼装起来。如果你从一个比较慢的站点访问一个有大量图像的 Web 页，就会看到数据的分割与重组。

同一子网内的主机之间可以直接发送 IP 包，而其他的 IP 包将被送到一个特定的主机：网关。网关（或路由器）是用来连接多个 IP 子网的，它们会转发从子网内来的 IP 包。例如，如果子网 10.42.1.0 和 10.42.0.0 之间通过一个网关相连，那么任何从子网 0 发往子网 1 的包必须由网关转发，网关可以帮这些包找到正确的路径。本地主机建立路由表是为了让 IP 包找到正确的机器。每一个目的 IP 都有一个条目在路由表中，用以告诉 Linux 将 IP 包送到哪一台主机。这些路由表是随网络的拓扑结构变化而动态变化的。

IP 协议是一个在不同网络之间传输数据的协议，其他上层协议可以用它来传输数据，典型的有 TCP 协议和 UDP 协议。

传输控制协议（TCP）是一个可靠的端对端的协议，它用 IP 来传送和接收它自己的包。正如 IP 包有它自己的头一样，TCP 也有它自己的头。TCP 是一个面向连接的协议，两个网络应用程序通过一个虚连接相连，即使它们之间可能隔着很多子网、网关、路由器。TCP 可靠地传送和接收两应用程序间的数据，并保证数据不会丢失。当用 IP 来传输 TCP 包时，IP 包的数据段就是 TCP 包。每一个通信主机的 IP 层负责传送和接收 IP 包。

用户数据报协议（UDP）也用 IP 层来传输它的包，不像 TCP，UDP 不是一个可靠的协议，但它提供了一种数据报服务。有多个协议可以使用 IP 层，接收 IP 包的时候必需知道该 IP 包中的数据是哪个上层协议的，因此 IP 包头中有个一字段包含着协议标识符。例如，当 TCP 请求 IP 层传输一个 IP 包时，IP 包的包头中用标识符指明该包包含一个 TCP 包，IP 接收层用该标识符决定由哪一协议来接收数据，这个例子中是 TCP 层。当应用程序通过 TCP/IP 进行通信时，它们不仅要指定目标的 IP 地址，而且还要指定应用的端口地址。一个端口地址唯一地标识一个应用，标准的网络应用使用标准的端口地址，例如 Web 服务使用 80 端口。这些已登记的端口地址可在 /etc/services 中找到。

为了实现 TCP/IP 协议族，不能仅仅有 TCP、UDP 和 IP。IP 协议层本身用很多种物理介质将 IP 包从一个主机传到其他主机。这些介质可以加入它们自己的协议头。以太网层就是一个例子。以太网允许很多个主机同时连接到同一根物理电缆。传输中的每一个以太网数据帧可以被所有主机接收到，因此每一以太网设备需要有个唯一的地址来标识它。任何传送给该地址的以太网帧被有该地址的以太网设备接收，而其他主机则忽略该帧。这个唯一的地址内置于每一以太网设备中，通常是在网卡出厂时就写在 ROM 中了。以太网地址有 6 个字节长，如：08-00-2b-00-49-A4。以太网帧可以携带很多种协议（作为数据），如 IP 包，并且也包括它们头中的协议标识符。这使得以太网层能正确地接收 IP 包并将它们传给 IP 层。

为了能通过像以太网这样的多连接协议传送 IP 包，IP 层必须找到每一 IP 主机的以太网地址。IP 地址仅仅是一个地址概念，以太网设备有它们自身的物理地址。从另一方面说，IP 地址是可以被网络管理员根据需要来分配和再分配的，而网络硬件只对含有它们自己的物理地址或多点传送地址的以太网帧作出响应。Linux 用地址解析协议（ARP）来将 IP 地址转变成真正的硬件地址，如以太网地址。如果一个主机想知道某一 IP 地址对应的硬件地址，它就用一个

多点传送地址将一个包含了该 IP 地址的 ARP 请求包发给网上所有节点，拥有该 IP 地址的的目标主机则响应一个包含物理硬件地址的 ARP 应答。ARP 不仅仅局限于以太网设备，它能够用来在其他一些物理媒介上解析 IP 地址，如 FDDI。对那些不支持 ARP 的网络设备，Linux 将不使用 ARP。还有一个提供相反功能的反向地址解析协议(RARP)，用来将物理网络地址转变为 IP 地址。这一协议常常被网关用来响应包含远程网络 IP 地址的 ARP 请求。

当配置 Linux 网络与外部世界建立通信连接的时候，通常要指定以下信息。说明这些的目的，是让大家加深 TCP/IP 的概念在实际运用中的理解。

(1) Hostname：主机名。是一个字符串，用来命名在网络上的主机。必要时向网络管理员请教命名规则。

(2) Domain name：域名。本地网络的域名。

(3) IP address：IP 地址。如 192.168.0.12。

(4) Network mask：网络掩码。用来确定 IP 地址中网络占的位数和主机号占的位数。如 255.255.255.0。

(5) Network address：网络地址(本地网络)。网络掩码与 IP 地址的逻辑乘。如 IP 地址 192.168.0.12，掩码为 255.255.255.0，则网络地址为 192.168.0.0。

(6) Broadcast address：广播地址。网络掩码取反后与网络地址的逻辑加。

(7) Gateway address：网关地址。通过网关可以访问其他网络的主机。

(8) Name server address：域名服务器地址。提供域名主机名和 IP 地址间的转化。

2.2　客户/服务器模型

客户/服务器(Client/Server)是现今大多数网络应用系统采用的结构。如我们已非常熟悉的 Web 浏览器和 Web 服务器。

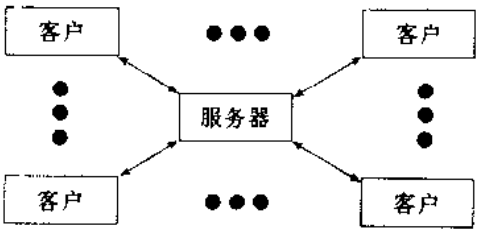

图 2.3　一个服务器同时处理多个客户的请求

多个客户可以向同一个服务器发起服务请求，服务器执行请求，并负责向这些客户返回请求响应。客户应用程序主动请求与服务器联系；而服务器应用程序一般是等待来自客户的请求。服务器由系统执行，在系统生存周期内始终处于运行状态。这就像许多人同时访问一个 Web 站点，由这个站点的 Web 服务器分别返回我们每个人所请求的页面。

服务器根据处理请求方式的不同，分成循环服务器和并发服务器两种类型。循环服务器依次处理每个服务请求，在没有完成一次任务时，随后的请求就等待；而并发服务器则建立多个进程来分别处理每一个服务请求。显然，前一种方式在长时间的服务情况下，会使响应速度大大降低；后一种方式由于并发许多进程消耗资源，对服务器的性能要求较高。

客户与服务器之间的这种通信就涉及到网络通信协议。当然，这个协议就是我们在上一节所讲述的 TCP/IP 协议族。客户与服务器都是典型的用户进程，TCP/IP 通常是系统协议栈的一部分。TCP/IP 的正式版本是 IP 版本 4(IPv4)，IP 版本 6(IPv6)在 90 年代兴起，将来有可能代替 IPv4。

这一模型具体的运行架构如图 2.4 所示。客户与服务器处于不同局域网，不同局域网通过路由器(router)连接到广域网。路由器是广域网的架构设备，而当前最大的广域网就是无所不在的 Internet。

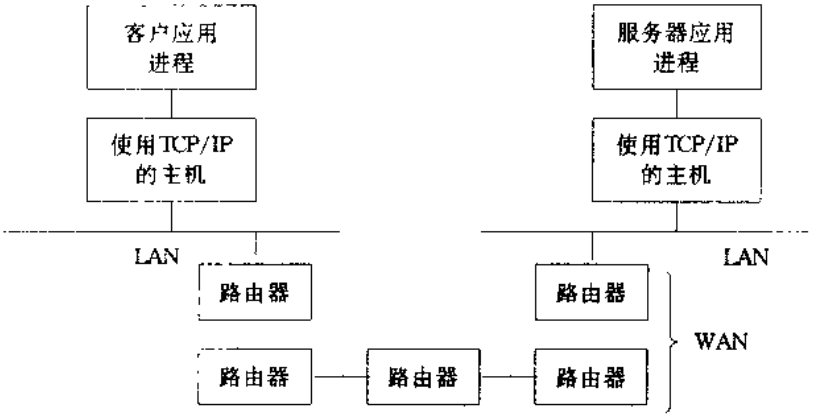

图 2.4　处于不同局域网的客户和服务器主机通过广域网连接

2.3　Linux 网络命令集

在学习网络编程技术的同时，熟练运用一些常用的网络命令会有很大帮助。这样有利于我们检测并调试我们的程序。

在 Linux 的发布中都拥有一些基本的网络命令，在这里描述和罗列出这些网络命令。如果你有一台 Linux 机器的话，希望这一节可以为你配置网络提供一些帮助。

网络配置(Network Configuration)

TCP/IP 测试及查错(TCP/IP Testing and Troubleshooting)

网络客户与服务(Network Clients and Services)

网络监视(Network Monitoring)

拨号网络(Dialup Networking)

1. 网络配置(Network Configuration)

大部分基本命令是用来配置网络接口和一些默认规则。虽然这些命令必须具有管理员的权限才可以使用，但是也并不禁止普通用户使用他们来确定机器的配置情况。注意：通常这些命令是在启动时由 shell 脚本调用，自动地进行网络配置。

Linux 在启动的时候识别网络接口(如果 kernel 配置正确的话)。每一个网络接口自动分配一个名称，例如"lo"、"eth0"等等。这些命令按照设置网络时使用顺序排列。

ifconfig

配置网络接口，或者显示当前网络接口状态。还可以激活和停止某个网络接口。这个命令