

Linux
与自由软件资源丛书

Linux

谢斌 罗勃 李丰君 等 编著

网站建设技术指南



机械工业出版社
China Machine Press

Linux与自由软件资源丛书

Linux网站建设技术指南

谢 斌 罗 勃 李丰君 等编著



机械工业出版社
China Machine Press

本书介绍如何在Linux下建立、运行并维护站点的综合技术,它适合于有一定的UNIX或Linux基本操作基础的网络建立者和维护者,同时也是广大Linux爱好者的一本较全面的参考书。

本书从建站的准备工作开始,写到建立功能强大的Internet网站,层层深入,涉及到网站建设中所遇到的种种问题。从Linux的安装和设置入手,详尽地介绍了如何建立普通站点及具有WWW、E-mail、FTP、BBS等功能的完整的Internet站点,并在最后介绍了网络的安全及网站的升级。本书还提供了笔者在实际操作中积累的一些小技巧,帮助你解决和避免实际工作中遇到的问题。

版权所有,侵权必究。

图书在版编目(CIP)数据

Linux网站建设技术指南/谢斌等编著. -北京:机械工业出版社, 2000. 8

(Linux与自由软件资源丛书)

ISBN 7-111-08156-0

I. L… II. 谢… III. 因特网-网站-操作系统(软件), Linux-基本知识 IV. TP393.092

中国版本图书馆CIP数据核字(2000)第39842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:李云静

北京昌平第二印刷厂印刷·新华书店北京发行所发行

2000年8月第1版第1次印刷

787mm×1092mm 1/16·15.5印张

印数:0 001-6 000册

定价:25.00元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

前 言

Linux 是本世纪90年代在Internet上发展并壮大起来的免费软件，由于其系统软件具有免费获取、硬件费用低廉的特点，近年来发展迅猛。Linux的应用软件已达上千兆。从DOS环境模拟到图像、音响信号的处理，从游戏到中文软件，无所不包。各大软件公司已纷纷开始推出Linux版本的商业软件。1994年底在华盛顿万国会议中心召开了Linux世界年会，各大软件公司都派出了阵容强大的代表团，展出了丰富的Linux产品，Linux开始进入商业应用时代。

据IDC研究表明，全球Linux用户数已超过1000万。1998年，美国有75万用户新安装了Linux操作系统，占中等大小服务器市场的17%。越来越多的软、硬件厂商加入了Linux的支持队伍中。硬件厂商像Intel、Compaq、IBM、Dell、HP等世界巨头都已介入Linux市场，很多硬件产品已经支持Linux；软件厂商也纷纷宣布支持Linux，IBM DB2、Oracle、Informix、Sybase都推出了其数据库的Linux版本，SUN的StarOffice办公软件也越来越受欢迎。

从我国的情况来看，Linux应用不如美国，尤其是在Internet服务器中，只占2%。但是，国家非常重视对Linux技术的研究，将研究以Linux为基础的操作系统及集成应用环境的软件列为国家重点发展项目；而且近日我国已经开发出以Linux为基础的操作系统，人们对Linux的认知度也越来越高。一些中文Linux系统都相继推出，代表作如XteamLinux中文版、TurboLinux中文版，以及去年红红火火的红旗Linux。总之，Linux必将以其独有的优秀特性成为网络服务器的主流。

在本书中，笔者根据自己在使用Linux中积累的经验，从Linux的基本认识和安装起步，逐步介绍了建立一个Linux下的网站所需要的每一步，并向读者提供了一些我们总结出来的技巧，使大家少走弯路。在最后一章中，我们以自己建立的一个Linux技术网站为案例，从最基本的软件安装起步，详细地介绍了一个动态交互式网站建立的全过程。相信每一位读者在阅读全书之后，都能轻松地建立自己的网站，成为这方面的专家。

本书共分11章，其中第1、3、4、5、6章由谢斌编写，第7、8、9、10、11章由罗勃编写，第2章由李丰君编写。另外参加编写的有：北方交通大学陈国勇，中国科技大学王辉、钟庆、李铨、林璟锵、傅文武、薛明东、杨攀、徐浩煜、路中华、张威、李岩、王飞、方正、阮戈、黄和、杨栋、盛大力、刘涛、张志达、侯东晖、叶翔、沈晓丰、吴冀。

由于作者水平所限，书中难免存在一些错误和不足，恳请广大读者批评指正。我们欢迎任何形式的建议和批评。我们的E-mail为 scjkj@163.net, bxywang@public.cd.sc.cn, 电话是：028-5410679, 028-5410306, 028-5412516。

作者

目 录

前言

第1章 了解Linux及其网络结构1

1.1 Linux简介1

1.1.1 何谓Linux1

1.1.2 Linux的功能2

1.1.3 Linux当前的版本3

1.2 Linux的TCP/IP网络结构5

1.3 IPv6简介7

1.3.1 IPv6与IPv4头标的比较7

1.3.2 简化的头标8

1.3.3 参数的修订9

1.3.4 新导入的域9

1.4 站点注册10

1.4.1 何谓站点注册10

1.4.2 注册Linux站点10

1.5 常见问题及解答11

1.5.1 关于运行DOS/Windows上的
程序11

1.5.2 关于Linux的资料来源12

1.5.3 关于获取帮助13

第2章 服务器平台安装和设备管理15

2.1 硬件选择15

2.1.1 CPU、RAM和主板与性能的
关系16

2.1.2 硬盘的选择16

2.1.3 显示卡与监视器的选择16

2.1.4 电源与UPS的选择16

2.2 Linux的安装准备17

2.2.1 Linux的获取17

2.2.2 了解你的计算机配置17

2.2.3 Linux的安装方式19

2.2.4 安装Linux的硬盘分区20

2.2.5 有关LILO的基础知识22

2.3 Linux的安装23

2.3.1 首先应该知道的事23

2.3.2 开始安装: 有关安装启动程序的

选择24

2.3.3 安装过程25

2.3.4 最后的设置工作38

2.3.5 安装LILO40

2.4 Linux下的硬件设备管理43

2.4.1 Linux设备管理概述43

2.4.2 SCSI设备46

2.4.3 硬盘47

2.4.4 CD-ROM驱动器49

2.4.5 调制解调器54

第3章 配置TCP/IP网络58

3.1 开始之前需要的信息58

3.1.1 IP地址58

3.1.2 网络掩码58

3.1.3 网络地址 (Network Address)58

3.1.4 广播地址 (Broadcast Address)58

3.1.5 网关地址 (Router/Gateway Address)59

3.1.6 名字服务器地址
(Nameserver Address)59

3.2 用netcfg配置网络59

3.2.1 操纵名字60

3.2.2 操纵主机60

3.2.3 增加网络接口设备61

3.3 安装网卡62

3.3.1 设置网卡模式62

3.3.2 配置网卡63

3.3.3 有关Intel Etherexpress系列卡的
配置64

3.3.4 网卡配置中的一些疑问65

3.4 路由、网关和IPchains65

3.4.1 路由和网关的概念65

3.4.2 使用IPchains实现数据包过滤和
转发66

第4章 Linux文件及系统管理68

4.1 文件和目录68

4.1.1 文件和目录概述	68	4.9.3 使用对换空间	101
4.1.2 文件类型	70	4.9.4 与其他操作系统共享对换空间	102
4.2 文件权限	70	4.9.5 分配对换空间	102
4.2.1 文件权限的概念	70	4.9.6 高速缓存	103
4.2.2 文件权限的构成	71	4.10 X Window	104
4.2.3 文件权限的依赖性	72	4.10.1 如果没有安装X	104
4.2.4 改变权限	72	4.10.2 如果已经安装了X	105
4.3 文件链接	73	第5章 Linux的网络设置	108
4.3.1 硬链接	73	5.1 配置NFS服务器	108
4.3.2 符号链接	74	5.1.1 NFS简介	108
4.4 文件系统和标准文件系统布局	75	5.1.2 设置NFS 服务器	109
4.4.1 文件系统	75	5.1.3 设置NFS客户机	110
4.4.2 标准文件系统布局	76	5.1.4 NFS的工作原理	112
4.5 基本目录介绍	77	5.1.5 NFS守护程序的启动过程	112
4.5.1 根文件系统	77	5.1.6 各服务进程的作用	113
4.5.2 /etc文件系统	78	5.2 建立PPP连接和配置PPP服务器	113
4.5.3 /dev文件系统	79	5.2.1 有关PPP的基础知识	113
4.5.4 /usr文件系统	80	5.2.2 使用PPP拨号上网	114
4.5.5 /var文件系统	81	5.2.3 配置PPP 服务器	115
4.5.6 /proc文件系统	82	5.3 配置DNS服务器	117
4.6 启动和关机	83	5.3.1 DNS (Domain Name System) 简介	117
4.6.1 启动和关机概述	83	5.3.2 域名服务系统	117
4.6.2 近观启动过程	83	5.3.3 域名服务器和解析器	117
4.6.3 重新启动系统	85	5.3.4 配置暂存专用名字服务器	118
4.6.4 紧急启动(软)盘	85	5.3.5 建立一个简单的领域名字服务器	121
4.6.5 有关关机的更多信息	85	5.3.6 配置实例	129
4.7 登录和注销	86	5.3.7 维护工作	133
4.7.1 通过终端登录	86	5.3.8 拨号网络连线的自动设置	134
4.7.2 通过网络登录	87	第6章 建立Internet 站点	136
4.7.3 login干了些什么	87	6.1 E-mail 服务器的建立	136
4.7.4 存取控制	88	6.1.1 sendmail 服务器的设置	136
4.7.5 Shell启动	88	6.1.2 在Linux机器上设置pop3代理	138
4.8 管理用户	88	6.2 FTP服务器的建立	139
4.8.1 系统管理员	89	6.3 WWW服务器的建立	141
4.8.2 管理职责	91	6.4 建立Samba服务器	154
4.8.3 用户管理	92	6.4.1 Samba是什么	154
4.8.4 用户组管理	96	6.4.2 安装Samba	154
4.8.5 使用LinuxConf管理用户	98	6.4.3 设置Samba的配置文件	156
4.9 内存管理	100	6.4.4 启动Samba	158
4.9.1 什么是虚拟内存	100	6.4.5 Samba常用的重要指令	159
4.9.2 产生对换空间	100	第7章 超文本标记语言(HTML)	162

7.1 HTML概述	162	8.7.2 日期时间函数	192
7.2 HTML文件描述	164	8.7.3 PHP的数据库功能及对MySQL 数据库访问	194
7.3 段落、格式、版面分割标记	165	第9章 建立BBS服务器	197
7.4 文字效果标记	167	9.1 BBS服务器概述	197
7.5 超级链接标记	170	9.2 Firebird BBS系统建站准备	199
7.6 文字列表标记	171	9.2.1 最初的准备工作	199
7.7 表格标记	173	9.2.2 BBS相关帐号的设置	200
7.8 插入图像标记	174	9.3 Firebird BBS服务器系统安装	201
第8章 PHP主页设计	176	9.4 测试BBS并建立系统帐号	202
8.1 内嵌式脚本语言PHP概述	176	9.5 BBS相关配置	202
8.1.1 PHP 发展历史	176	9.5.1 BBS配置文件sysconf.ini	202
8.1.2 PHP的主要技术特点	177	9.5.2 sysconf.ini中定义的主要权限 说明	204
8.2 PHP语句	177	9.5.3 版面的设置	205
8.2.1 初识PHP	177	9.5.4 Firebird BBS主要文件说明	206
8.2.2 PHP语句和HTML分离	178	9.6 BBS转信功能	208
8.3 PHP中的变量	179	9.6.1 主动式和被动式收信	208
8.3.1 变量名和变量类型	179	9.6.2 BBS送信工作原理	209
8.3.2 深入了解变量类型转化	180	9.6.3 安装转信程序innbbsd	209
8.3.3 数组类型	181	第10章 服务器的安全性概述	210
8.3.4 动态变量	181	10.1 服务器安全管理	210
8.3.5 类和对象	181	10.1.1 安全防护的主要内容	210
8.3.6 变量作用域	182	10.1.2 Linux系统的文件安全	210
8.4 外界变量和交互式网页	183	10.1.3 用户访问安全	211
8.4.1 PHP从客户端获取信息的方法: 外界变量	183	10.1.4 日常安全注意事项	213
8.4.2 使用HTTP Form获取信息	183	10.1.5 服务器被侵入后的处理	213
8.4.3 HTTP Cookies与客户端信息	185	10.2 防火墙、IP伪装和代理服务器	214
8.4.4 使用环境变量获取客户端信息	185	10.2.1 什么是防火墙	214
8.5 基本控制流程: 分支和循环	185	10.2.2 防火墙分类	214
8.5.1 条件语句和分支结构	185	10.2.3 Linux防火墙实现策略	215
8.5.2 循环语句	188	第11章 网站建设实例	216
8.6 表达式基础	190	11.1 Apache、PHP、MySQL简介	216
8.6.1 算术运算符	190	11.2 Apache、PHP、MySQL的安装	216
8.6.2 字符串运算符	190	11.2.1 安装和启动MySQL	216
8.6.3 赋值运算符	190	11.2.2 安装PHP+Apache	218
8.6.4 位运算符	191	11.2.3 启动Apache并测试	219
8.6.5 逻辑运算符	191	11.2.4 测试MySQL数据库	220
8.6.6 比较运算符	191	11.3 网站设计和主页面制作	221
8.7 PHP常用函数和MySQL数据库访问 函数	191	11.3.1 网站总体设计	221
8.7.1 PHP内置数学函数	191	11.3.2 网站风格设计	222

11.3.3 需要注意的问题	222	11.4.3 Web-BBS主页面制作	228
11.3.4 “Linux网站建设者” 首页	222	11.4.4 “发表文章” 功能块制作	237
11.4 网站设计和主页面制作	224	11.4.5 几点探讨和功能扩充	240
11.4.1 Web-BBS的结构分析	224		
11.4.2 数据库的建立和测试	227		

第1章 了解Linux及其网络结构

得到这本书，用户一定会尝试着在它的指导下使用Linux建立自己的网站。本书将详尽描述这个过程。用户将会逐步地认识到Linux的强大功能及利用Linux建立网站的便利之处。

1.1 Linux简介

1.1.1 何谓Linux

Linux是UNIX克隆(UNIX clone)或UNIX风格(UNIX alike)的操作系统(OS)，它在源代码级上兼容绝大部分UNIX标准(指的是IEEE POSIX、System V、BSD)，是一个支持多用户、多进程、多线程、实时性较好的功能强大而稳定的操作系统。它可以运行在x86 PC、Sun Sparc、Digital Alpha、680x0、PowerPC、MIPS等平台上，是目前运行硬件平台最多的操作系统。Linux最大的特点在于它是GNU(简单地说，GNU是一种自由软件体系)的一员，遵循公共版权许可证(GPL)，秉承“自由的思想，开放的源码”的原则。成千上万的专家及爱好者通过Internet在不断地完善并维护它，可以说，Linux是计算机爱好者自己的操作系统。

Linux诞生于1990年，Linus Torvalds，芬兰赫尔辛基大学的一名学生，用汇编语言写了一个在80386保护模式下处理多任务切换的程序，后来从Minix(Andy Tanenbaum教授所写的很小的UNIX操作系统，主要用于操作系统教学)得到灵感，进一步产生了自认为狂妄的想法——写一个比Minix更出色的Minix，于是便开始写了一些硬件的设备驱动程序，一个小的文件系统。这样，0.0.1版本的Linux就诞生了，但是它只具有操作系统内核的雏形，甚至不能运行，必须在有Minix的机器上编译以后才能运行。这时候Linus已经完全着迷而不想停止，决定抛开Minix，于是在1991年10月5日发布了Linux 0.0.2版本。在这个版本中已经可以运行bash(the GNU Bourne Again Shell——一种用户与操作系统内核通信的软件)和gcc(GNU C编译器)。从一开始，Linus就决定自由扩散Linux，包括源代码，他在comp.os.minix新闻讨论组里发布Linux 0.0.2时写到：

“Do you pine for nice days of Minix-1.1, when mem were men and wrote their own device drivers? Are you without a nice project and just dying to cut your teeth on a OS you can try to modify for your needs? Are you finding it frustrsting when everything works on Minix? No more all-nighters to get a nifty program working? Then this post might be just for you.

“As I mentioned a month ago, I'm working on a free version of a Minix-lookalike for AT-386 computers. It has finally reached the stage where it's even usable(though may not be depending on what you want), and I am willing to put out the sources for wider distribution. It is just version 0.0.2 ... but I've successfully run bash, gcc, gnu-make, gnu-sed, compress, etc.under it.”

随即，Linux引起了黑客们(hacker)的注意，通过计算机网络加入了Linux的内核开发，Linux倾向于成为一个黑客的系统——直到今天，Linux内核的开发被认为是真正的编程。由

于一批高水平黑客的加入,使Linux发展迅猛,到1993年底1994年初, Linux 1.0终于诞生了。Linux 1.0已经是一个功能完备的操作系统,而且它的内核写得紧凑高效,可以充分发挥硬件的性能,在4MB内存的80386机器上也表现得非常好,至今人们还在津津乐道。不过自从2.1.xx系列的内核出现以来, Linux开始走高端的路子——硬件的发展实在太快了。但是Linux不会失去它的本色。Linux具有良好的兼容性和可移植性,大约在1.3版本之后,开始向其他硬件平台上移植,包括当前最快的CPU,所以不要总把Linux与低档硬件平台联系到一块, Linux发展到今天,这一直是一个误区,它只是将硬件的性能充分发挥出来而已, Linux必将从低端应用横扫到高端应用。

在Linux的发展历程上,还有一件重要的事: Linux加入GNU并遵循公共版权许可证(GPL)。此举大大加强了GNU和Linux,几乎所有应用的GNU库/软件都移植到Linux上,完善并提高了Linux的实用性。而GNU也有了一个根基,更重要的是遵循公共版权许可证,在继承自由软件精神的前提下,不再排斥对自由软件的商业行为(如把自由软件打包以光盘形式出售),不排斥商家对自由软件进一步开发,不排斥在Linux上开发商业软件。从此Linux又开始了一次飞跃,出现了很多的Linux发行版。如Slackware、RedHat、Suse、TurboLinux、OpenLinux,支持中文的有Xteam、Bluepoint以及刚刚出炉的红旗Linux等。Linux主要是指操作系统内核,对所有发行版内核源代码都是一样的(但集成的内核版本可能因发行时间不同而有所不同)。还有一些公司在Linux上开发商业软件或把其他UNIX平台的软件移植到Linux上来,如今很多IT业界的大腕,如IBM、Intel、Oracle、Infomix、Sysbase、Corel、Netscape、CA、Novell等都宣布支持Linux!商家的加盟,弥补了纯自由软件的不足和发展障碍。Linux迅速普及到广大计算机爱好者,并且进入商业应用,正是打破某些公司垄断文化圈的希望所在!

Linux是爱好者们通过Internet协同开发出来的,当然它的网络功能十分强大,比如可以通过ftp、nfs等来安装Linux,用它来做网关等等。随着Linux的发展衍生出来的应用恐怕出乎Linux本人最初的预料,如有人用它来做路由器,有人用来做嵌入式系统,有人来做实时性系统……常有新手问Linux能做什么,其实它不像那些中看不中用的操作系统,不在于它能干什么,而在于用户想干什么。

1.1.2 Linux的功能

1. 个人Linux工作站

无论在家中,还是在办公室里, Linux与基于Intel芯片PC的结合都会创造出一台功能强大的Linux机器。Linux对于那些负担不起Sun或是HP工作站的公司,对于每一个工程师和所有在X终端上遇到困难的人来说,都是极棒的产品,它同时也是希望保留一些旧格式文件和继续使用原有程序的DOS和Windows混合环境下用户的最佳选择。

2. Internet服务器

Linux是Internet的产物,而且擅长提供Internet服务。Linux缺省提供WWW、Usenet新闻、电子邮件、FTP等许多功能,可以访问网络内部用户,也可以通过整个Internet发布消息。如果与拨号的调制解调器相连(使用多个串行口), Linux就可以变成强大的Internet访问接入点。大量的ISP因为Linux的可靠性和性能而选择了它。

3. X终端客户

Linux对X11的优秀支持简化了应用服务器上远程运行应用(从Linux机器到一个多处理

器的超级计算机)的过程,并且可以在本地的Linux工作站上显示信息。而且,Linux比传统X终端更占优势,因为Linux提供了全方位的多媒体服务,包括声音、动画和高级图像支持。如果用户需要其Linux机器用于多台X终端,则可以非常快速而方便地实现。考虑到目前硬件速度越来越快,越来越可靠,价钱却越来越低,有谁不愿意花最少的钱买更多的实惠呢!

4. 终端服务器、传真服务器、Modem服务器

Linux也能很好地支持串行设备和电话。昂贵的定制的终端服务器(如Annex)通常支持SLIP、PPP、Direct Connection(直接连接)、Dial up Apple Talk 和The Internet Adapter。Linux不但能提供上述功能,还可以提供定制的安全性、身份验证和登录过程。一个ISP的中等系统可以连接200多个调制解调器,提供并维护可靠的拨号服务。

5. 网络服务器

与商业组织类似,教育机构也热衷于将Linux用作企业服务器。用于文件及打印共享时,可将Linux配置为使用NFS、Apple Talk及NetBIOS协议。其性能优价格廉的特点使它颇具吸引力。由LAN桥接而生成WAN也是Linux的一个很好的用途。当Linux运行于RISC和SMP(对称多处理)硬件系统时,它可为严谨的后台处理带来便利及易用性。

6. X应用服务器

任何作为X终端的机器都能体会到Linux支持X应用的范围之大。使用Linux作为应用服务器去加快RISC工作站与使用Linux作为工作站一样,都是非常合理的选择。如果把Linux配置成应用服务器,就能通过运行DOS X服务器把286、386等低档次的机器作为哑终端重新利用起来。这也说明了Linux是低预算的X Window网络的选择。

7. UNIX开发平台

Linux能够支持UNIX开发,不但支持主流语言,也支持其他语言;其跨平台的环境还可以为其他UNIX操作系统平台产生二进制代码;GNU C、GNU C++、GNU Fortran 77、ADA、Pascal、Modula 2和3、TCL/Tk、Scheme 和SmallTalk/X 均可免费使用,并附带详细的工作代码库;流行的Motif Widgets也能从几个供应商手中买到;多种语言的编译器有助于节省开发时间。所有这些,再加上灵活的shell语言编辑器(大小写敏感)、源代码包和详细文档都给了编程者充分的可定制环境。另外,它也能作为一个理想的计算机学习系统,在不打扰用户共享设施的环境中,尽情控制这个复杂的系统。

8. 商业开发

在商业开发系统中,执行CA/Clipper(基于dBase和Fox超集的面向对象编译器的含义是,Clipper的开发者只需对软件做很少的改动就能适应Linux的运行。其结果是功能相同,而性能提高了,在一样的硬件条件下,速度比DOS版本快了20~200倍。

1.1.3 Linux当前的版本

Linux说到底只是个kernel(内核)而已,一些公司和爱好者在不断地开发和推出新kernel,而相应的应用工具(utilities)和软件(software)则多用GNU的和其他的自由软件(freeware)。由于Linux基本遵循POSIX标准,因此大多数UNIX软件的编译移植不会有太大的困难(UNIX可移植性相当好,同一套源代码在小至微机大至巨型机上都可编译)。而将Linux kernel和这些外围的程序整合起来的任务是由Linux发行商(distribution)完成的,他们推出不同的发行版本(distribution)。

常见的Linux发行版本如下。

1. RedHat 6.0 <http://www.redhat.com>

基于rpm的包结构，核心代码约有200MB左右，其好处是安装使用简单方便，并且rpm包的结构也使包的安装和卸载方便多了。另外RedHat的更新升级版(updates)出得也很及时，被评为Best Network OS。

曾经受到抨击的RedHat 6.0的问题是，由于采用了glibc 2.0(libc 6)，其C库头文件等有一些改动，导致一些软件产生编译时出错问题。但只要改动一下#include就好了。Official RedHat 6.0是双CD的：

CD 1: Binary CD (RPMS和live filesystem)

CD 2: Source CD (SRPMS)

Official CD中带以下几个ftp version没有的应用工具：MetroLink X Server、RealVideo Server/Client、BRU(Backup/Restore Utility)。另外每个RedHat每个版本还会有个powertools，是一些候选进入正式发行版本的packages。还有个很大的contrib archive，一般是一些软件的升级版等。不过安装时要正确选择目录，hurricane的才是5.0的(btw: RedHat 4.2代号Biltmore, 5.0代号Hurricane)。

2. Slackware 3.4 <http://www.cdrom.com>

基于tgz的包结构，历史很悠久的Linux distribution，其核心部分有120MB左右。Slackware在国内用得很多，比较公认的是，它用来做服务器时性能最好。一般推荐新手入门使用RedHat，因为它安装容易，几乎是一路回车就能装上。但是若使用Linux已经有一段时间，那就无所谓了，因为自己有了评价不同版本的能力，就可以根据自己的习惯、喜好和用途来选择。

Official 4 CD set:

CD 1: Slackware 3.4 binary & source

CD 2: sunsite.unc.edu archive 1

CD 3: sunsite.unc.edu archive 2

CD 4: tsx-11.mit.edu archive

Slackware 3.4 是基于libc5的，因此源码兼容性好些，但其缺点是binary中东西太少，许多东西还要自己下载、编译。

3. Debian 1.3.1 <http://www.debian.org>

基于deb的包结构，这也是一个很不错的distribution，也称GNU/Linux，与GNU的关系紧密。其特点是，收集的软件很全，binary近400MB，而且其deb包安装删除也很方便，据介绍，在业余卫星上还用过Debian做OS。

4. SuSE Linux 5.2 <http://www.suse.com>

SuSE是从其X Server开始的，它和XFree 86合作开发x86上的X Server。SuSE也有自己的distribution。由于这是家德国公司，该发行版在欧洲用得更为广泛。X11R6.4的Linux测试平台就是SuSE Linux 5.0。

5. TurboLinux 4.0 简体中文标准版 10CD

TLC 4.0标准版10CD的内容：

Disc 1: Installation(625MB)

Disc 2: Source (607MB)

Disc 3: Application (553MB)	Disc 4: Console<1> (600MB)
Disc 5: Console<2> (563MB)	Disc 6: X11<1> (610MB)
Disc 7: X11<2> (548MB)	Disc 8: Themes (560MB)
Disc 9: Perl<CPAN>(495MB)	Disc10: Games (626MB)

6. 红旗Linux

由中国科学院软件所、北大方正电子有限公司、康柏电脑公司三家合作共同推出的国产中文操作系统红旗Linux，预装了炎黄中文平台和方正TrueType字库，是目前国内唯一一套在Linux上支持大字符集(GBK)的中文操作系统，实现了Linux上的TrueType显示和打印功能，并且从安装到使用提供了全中文文化的操作环境；红旗Linux支持基于Intel芯片的各类PC和服务器的，并且支持基于ALPHA芯片的工作站和服务器等，目前该平台产品在硬件平台支持方——康柏公司——的PC服务器及ALPHA机上均能稳定运行，红旗Linux彻底支持Informix-SE、Oracle 8以及Sybase等多种流行数据库，并提供了Linux上数据库解决方案的实例，为用户开发基于Linux的数据库应用开创了先例。

1.2 Linux的TCP/IP网络结构

网络和Linux是密切相关的。在某种意义上Linux是一个针对Internet和WWW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自身常常被用来支持各种组织机构的网络需求。这一节讲的是Linux如何支持如TCP/IP等网络协议的。TCP/IP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(NSF1 NET和Internet)却变得更大了。现在我们所熟知的万维网(World WideWeb)就是从ARPANET演变过来的，它自身支持TCP/IP。UNIX TM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 TM 平台。

在IP网络中，每台机器都有一个IP地址，一个32位的数字，它惟一地标识这台机器。WWW是一个非常巨大并且迅速增长的网络，每台连在上面的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16.42.0.9这个IP地址实际上分成两部分：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16.42.0.9为例，网络地址是16.42，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16.42.0.9为例，子网地址是16.42.0，主机地址是16.42.0.9。这样就可以允许某部门划分他们自己的子网络。例如，如果16.42是ACME计算机公司的网络地址，则16.42.0可能是子网0，16.42.1可能是子网1。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，而域名则容易记，像linux.acme.com，就比16.42.0.9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。这些名称被静态地定义在/etc/hosts文件中，或者Linux能请求域名服务器(DNS)来解析它。这种情况下，本地主机必须知道一个或一个以上的DNS，并且这些服务器要将其名称指定到/etc/resolv.conf 中。

当希望与另一台计算机连接时，比如说想阅读一个Web页，用户的IP地址就会用来与那台

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、校验和以及其他的有用信息。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，譬如由于电话线路的问题而引起的错误。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，它们会转发送从子网内来的IP包。例如，如果子网16.42.1.0和16.42.0.0之间通过一个网关相连，那么任何从子网0发往子网1的包必须由网关指引，网关可以帮这些包找到正确的路线。本地主机建立路由表用以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是一个传输层的协议，其他协议可以用它来传输数据。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包的时候必须知道该IP包中的数据是哪个上层协议的，因此IP包头中有个一字节包含着协议标识符。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。一个端口地址惟一地标识一个应用，标准的网络应用使用标准的端口地址，如Web服务使用80端口。这些已登记的端口地址可在 /etc/services 中看到。

这一层的协议不仅仅是TCP、UDP和IP。IP层本身用很多种物理媒介将IP包从一个主机传到其他主机。这些媒介可以加入它们自己的协议头。以太网层就是一个例子，但PPP和SLIP不是这样。一个以太网(Ethernet)允许很多个主机同时连接到同一根物理电缆上。传输中的每一个以太网帧可以被所有主机看见，因此每一以太网设备有个惟一的地址。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SRAM中了。以太网地址有6B长，如：08-00-2b-00-49-A4。一些以太网地址是保留给多点传送用的，送往这些地址的以太网帧将被网上所有的主机接收。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)来允许机器将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该IP地址的ARP请求包发给网上所有节点,拥有该IP地址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。那些不支持ARP的网络设备会被标记出来, Linux将不会用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。

1.3 IPv6简介

随着Internet用户的迅速增加,网络地址不足的危机日益严重,按目前入网主机的增长速度预计,到2005年左右IP地址将被耗尽。Internet面临的另一个问题是路由表规模的急剧膨胀,如不采取措施,可能在IP地址枯竭之前就会导致网络瘫痪。正是在这一背景下,提出了新一代的Internet协议——IPv6。

IPv4是非常出色的设计,如果它存在着重大缺陷,Internet就不会发展成今日的规模。IPv6继承了IPv4的优点,并根据10年来对IPv4运用的经验进行了大幅度的功能扩充。

1.3.1 IPv6与IPv4头标的比较

首先介绍新IP的头标格式。IPv6的头标在起始64比特之后是128比特的源地址和信宿地址,全长为40B。图1-1展示了IPv6头标的格式。

起始的64比特由以下的域构成:

- 版本域(version field) 4比特
- 优先级值(priority value) 4比特
- 流标记(flow label) 24比特
- 净荷长度(payload length) 16比特

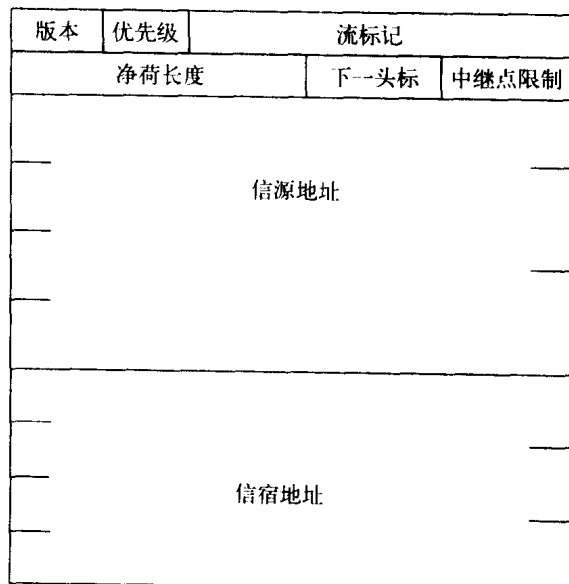


图 1-1

- 下一个头标(next header) 8比特
- 中继点限制数(hop limit) 8比特

新头标的结构比IPv4简单得多。IPv4中有10个固定长度的域、2个地址空间和若干个选项, IPv6中只有6个域和2个地址空间。图1-2展示了IPv4头标的格式。

在IPv6中起始4比特的版本号和IPv4的意义相同, 其位置保持不变。这是由于考虑到同时利用IPv4和IPv6进行通信的情况。也就是说, 原本设想生成数据链路帧时进行相同的封装, 使IPv4和IPv6能够共用数据链路层的驱动程序。

网络层利用此版本域来决定对分组的处理。当版本值为4(0100)时, 就认为是IPv4的分组, 版本值为6(0110)时, 就看作IPv6的分组。然而实际上舍弃了这种想法, 只作有限的使用, 这是由于在底层可以将IPv4和IPv6分离的缘故。例如, 在以太网上传送IPv6分组时, 以太网的帧类型(frame type)是十六进制数的86DD, 而不是IPv4的8000(十六进制数)。

版本	头标长度	服务类型	分组总长	
标识符			标 志	报片偏移
生存时间	协议		头标校验和	
源IP地址				
信宿IP地址				
选项				填充域

图 1-2

IPv4中的头标长度(header length)、服务类型(TOS: type of service)、标识符(identification)、标志(flag)、报片偏移(fragment offset)和头标校验和(header checksum)这6个域被删除。分组全长(total length)、协议类型(protocol type)和生存时间(TTL: time to live)3个域的名称或部分功能被改变, 其选项(option)功能完全被改变, 新增加了2个域, 即优先级和流标记。

1.3.2 简化的头标

IPv4中的头标是基于1975年的技术设计的, 至今已有20余年历史。IPv6的头标主要是在以下3个方面作了简化:

- 1) 所有的头标长度固定。
- 2) 删除头标校验和功能。
- 3) 删除各路由器的分拆成报片的处理功能。

在IPv6中没有IPv4处理特殊分组时使用的可变长度选项。这并不是说IPv6不能处理特殊的分组。在IPv6中不用IPv4的变长选项域, 而是在基本头标之后加上扩展头标(extension header)来处理特殊分组。由于基本头标的长度固定, IPv6自然就不需要头标长度域了。

删除头标校验和的优点是, 每做一次分组的中继转发时不必进行校验和的确认和更新, 因而能够减轻头标处理的负荷。其缺点也是显而易见的, 由于不能检测出差错, 可能出现转发。但是这一缺点是无足轻重的, 因为几乎所有的数据链路层都进行封装(capsule)处理、计算帧的校验和。例如IEEE802网络的媒体访问控制处理、ATM线路的适配层、串行线路的PPP都具有差错校验的功能。

IPv4具有报片拆/装功能, 因此发信方不必考虑中继点的传输容量, 能够发送很大的数据

报。如有必要,在中继点上将这些数据报分割成适当大小的报片,接收方等待所有报片的到达,重组数据报。但是,若要在只能传送小报片的网上发送大的分组,只有当全部报片到达接收方时才能认为分组发送成功,即使只丢失了一个报片,也需重发整个分组。可见这种方式不能有效地利用网络。IPv6利用了“路径MTU发现(path MTU discovery)”的功能,由主机学习能够传送报片的最大长度。如果传送很大的分组,此分组也只是在网络上被抛弃。其结果,IPv6就不需要IPv4中使用的分组标识符、报片偏移等拆/装控制域。但是在IPv6中,终端主机间还是需要进行拆/装处理。此外,假定所有的IPv6网络至少能传送净荷为536B的分组,当主机不想知道路径MTU或没有记住路径MTU时,只发送小分组即可。

IPv6简化后删除了TOS域。在IPv4中主机指定大带宽、最短、最便宜或最安全的路径等优先级设置TOS。但是此域在应用中几乎未被采用。

1.3.3 参数的修订

和IPv4一样,IPv6头标中也含有描述分组长度、TTL和协议代码等功能的域。但是这些域的定义是根据运用经验修订的。

IPv4的分组总长度在IPv6中变为净荷长度(payload length)。净荷长度是指接在头标之后的数据长度,两者之间没有太大的差别。例如,考虑由20B的TCP头标和400B的应用数据组成净荷。IPv4在TCP头标前添加20B的IPv4头标,分组长度值变为440B。IPv6添加了40B的头标,净荷长度不是460B而是420B。在IPv6中净荷长度域的大小与IPv4的总长域相同,都是16比特,故分组的尺寸限制在64KB。但当IPv6传送不能用净荷长度域描述的那种特大分组时,可以利用“特大净荷(jumbo payload)”选项。

IPv4协议类型域在IPv6分组中被定义为新的结构,它变为“下一头标类型(next header type)”。在IPv4中接在IPv4头标之后是UDP和TCP的传输层协议数据。IPv6数据报具有与其完全相同的结构,在下一个头标类型中可设置成UDP(17)及TCP(6)等协议类型。在下一节将说明在IP头标和作为净荷的TCP/UDP之间可以插入扩展头标的结构。些时在下一个头标号码域中可以设置首先出现的扩展头标类型。

将TTL改为“中继点限制(hop limit)”使其设计名副其实。在IPv4中TTL以秒为单位表示分组直至从网中消灭为止的剩余时间。如果允许分组在网中无限制地存在下去,旧分组无目的地出现时,将引起协议出错。例如,TCP中连接结束后,有时暂时保持“连接状态”。这是为了保证属于原有状态的所有分组从网络中消灭。此机制仅当传输层协议能够检测到分组在网络中的生存时间的情况下起作用,这就是TTL域的目的。在IPv4的规范中要求每个路由器减去1秒或在路由器队列驻留1秒以上时减去相应的秒数。但是实际上,正确推算分组的等待时间是十分困难的,并且,实际的时间不是以秒而是以毫秒为单位,因此几乎所有的路由器只对TTL作减1的处理。这一习惯成为IPv6的规范,因而域的名称也作了相应的改变,中继点限制不是秒数而是中继点的数目。以此为前提,可以认为传输层协议具有保护滞留在网中的分组的功能。例如,在TCP中设置了很大的序号(sequence number)域,因序号不会立即循环一周,故滞后分组到达时,也不会产生误动作。

1.3.4 新导入的域

IPv6的头标中流标记(flow label)和优先级(priority)这两个域在IPv4中是没有的。它们都是为