



网络建设与服务技术丛书

E mail

服务器配置和管理

段小华 等 编著

098



清华大学出版社
<http://www.tup.tsinghua.edu.cn>

332

7.07.01/8
980

网络建设与服务技术丛书

Email 服务器配置和管理

段小华 等 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书主要是介绍当前四大网络服务器中十分重要的一种服务器——Email 服务器,详述了目前常用的几种 Email 服务器的安装配置和使用方法,体系划分清楚,内容深入浅出,并提供具体示例和图解,使读者能够方便快捷地掌握 Email 服务器的管理和建设。

本书由四部分组成,第一部分介绍有关邮件服务及邮件服务器的基本概念和基础知识,第二部分介绍 Windows 9x & NT 环境下的 Email 服务器的配置和使用,包括常用的 FTGate、MailMax 和 Exchange 等。第三部分介绍在各种不同的 Unix 环境下 Email 服务器的配置和使用,重点针对当前使用最普遍的 Sendmail 程序进行讲述,同时也介绍了一些其他的服务器。第四部分介绍了有关 Email 服务器的一些高级使用内容,其中包括 FTPMail 服务器的介绍,Email 服务器的安全性问题等。附录 A 和 B 有 smail 和 sendmail 的原样本文件,附录 C 是常用术语的解释。

本书适用于使用不同操作系统的网络管理人员和网站开发人员,同时也适合广大的网络爱好者和邮件用户参考。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Email 服务器配置和管理/段小华等编著.—北京:清华大学出版社,2002
(网络建设与服务技术丛书;4)
ISBN 7-302-05130-5

I. E... II. 段... III. 电子邮件-服务器(软件)-基本知识 IV. TP393.098

中国版本图书馆 CIP 数据核字(2001)第 094226 号

出 版 者: 清华大学出版社(北京清华大学学研大厦,邮编:100084)

<http://www.tup.tsinghua.edu.cn>

印 刷 者: 清华大学印刷厂

责任编辑: 徐培忠

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 **印张:** 19.5 **字数:** 473 千字

版 次: 2002 年 3 月第 1 版 2002 年 3 月第 1 次印刷

书 号: ISBN 7-302-05130-5/TP·3007

印 数: 0001~5000

定 价: 29.00 元

丛 书 序

经过长时间的策划、编写、修改与订正，本套书终于得以与读者见面了。编写本套丛书的目的就是希望为网络事业的发展尽一点微薄之力；同时，我们也非常想与读者们一起交流思想，相互学习。

本套网络丛书主要介绍了网络的构架和一些实用的开发语言和软件，丛书一共有 6 本，分别为：

- 《Apache 服务器配置和管理》
- 《MySQL 高级配置和管理》
- 《PHP 语言进阶和高级应用》
- 《Email 服务器配置和管理》
- 《PostgreSQL 数据库开发和使用》
- 《网络安全和黑客揭密》

使用 Apache 服务器、MySQL 数据库和 PHP 语言可以构架出十分出色的网站，并且可以进行高效的数据查询的信息传递，是目前非常流行和实用的软件。丛书的前 3 本就从这三种工具入手，深入浅出地介绍和探讨了网站构成的方法，适合于网络管理人员和维护人员使用（当然也适合于初学者，不过最好是先了解一些相关的 Linux 知识）。

由于邮件发送关系到信息和数据的交互和传递，因此在网站构成的基础上建立邮件系统是一个重点，进而怎样配置和管理网站的邮件系统就十分重要了。《Email 服务器配置和管理》一书的编纂就基于此。该书对常用的几种 Email 服务器的安装、配置和管理做了颇为详尽的介绍，希望使读者能够快速掌握 Email 服务器的管理和建议，并将关于 Email 服务器的知识提高到一个系统的层次。

PostgreSQL 的应用在我国起步还不久，使用也比较少，但是它具有很好的前景和强大的生命力。作为数据库系统，它在网络上的优点表现得越来越多，因而掌握 PostgreSQL 这一工具也就越来越重要了，为此本套书特意加入了《PostgreSQL 数据库开发和使用》这本书。

随着网络的发展，网络的安全也变得越来越重要，怎样对重要资料进行保护、怎样防止黑客入侵、怎样构建防火墙等等，在组建网站的时候，这些问题都应当全面考虑。《网络安全和黑客揭密》一书对上述问题进行了全方位的探讨，并且补充了前几本书中对于安全问题说明的不足之处。该书从基本的网络和网络安全知识讲起，由浅入深，内容丰富、涵盖面广，既适用于刚开始学习网络和网络安全的初学者，也可供中级和高级的网络管理者和网络编程人员参考。

以上是本套丛书内容的大致介绍，作者相信本套丛书能够使您对网络有个全面的了解和掌握。

前 言

Email 服务器实际上是驻留在服务器主机上的一个程序，它和客户端方面不断地传送着各种信息，它们之间使用一定的协议（如 SMTP、POP3 和其他的一些协议）互相通信。因此，有些时候把 Email 服务器也称做 SMTP 服务器或 POP3 服务器。它是 Internet 应用中使用最早、最多和最基本的服务器，因为只要能与 Internet 连接的人都可以使用电子邮件，而不管连接方式是 UUCP(Unix-to-Unix-Copy-Protocol)连接方式，还是拨号方式，或者是通过网关。而且，电子邮件不需要交互式使用或者快速地响应，也不像别的服务如 Gopher、WWW 服务器那样需要专门的工具，且在使用过程中要等待很长的时间。

Email 服务器是四大网络服务器（指 HTTP 服务器——又称 Web 或 WWW 服务器、Email 服务器、FTP 服务器、Telnet 服务器）之一，其基本结构是采用开放式主从结构（server/client），分成服务器端和客户端两个部分。服务器是指结构中规定了服务器的传输设定、信息传输格式及服务器本身的基本开放结构；客户端是指结构中规定了信息接收格式以构建适当的信息接收工具，例如邮件阅读器。邮件服务的大量信息都存放在服务器上，服务器的作用就是管理这些文档，处理用户发来的各种请求，并将满足用户要求的信息返回给用户。

按其功能，Email 服务器可分为电子邮件信息服务器和邮件清单。电子邮件信息服务器是指能够接受电子邮件请求并且能通过电子邮件返回信息的一个程序。FTPMail 也是一种类型的文件服务器，能通过电子邮件得到那些在 Internet 上用 FTP 方式才能得到的文件。邮件清单是指一组电子邮件地址，通过给一个地址发送信息就可以将这个信息发送到这一组邮件地址（通过别名来实现）。加入清单的人通过给清单地址发送电子邮件来讨论一个话题；每个到达清单的信息都能分发给所有清单订阅者。清单地址能够手工维护或通过专门的清单管理器系统来自动维护。

本书十分适用于使用不同操作系统的网络管理人员和网站维护人员，同时也适合广大的邮件用户和网络爱好者作为参考。阅读本书的读者最好对 Email 服务器有一些基本了解，或者有一个具体的邮件系统的使用经验，并且掌握网络操作系统、通信协议和相关技术的知识。如果你想了解如何安装和配置一个 Email 服务器，以及执行它的基本的邮件服务功能，那么本书不失为一本及时的参考书。该书以具体的操作系统为背景，对常用的几种 Email 服务器的安装、配置和管理做了颇为详尽的介绍，尤其对 sendmail 服务器的介绍分不同章节分别阐述，附有大量示例，以使复杂的内容浅显易懂，让读者逐步地接受。希望在阅读完本书后，使读者能够快速掌握 Email 服务器的管理和建设，将关于 Email 服务器的知识提高到一个系统的层次。

由于作者所识有限，加之时间紧张，书中错误和不妥之处在所难免，欢迎广大专家和读者对本书不吝批评指正。

编 者

2001 年 5 月

目 录

第 1 章 认识电子邮件	1
1.1 邮件消息	1
1.2 邮件的投递	3
1.3 Email 地址	3
1.4 邮件的路由	4
1.4.1 Internet 上的邮件路由选择	5
1.4.2 UUCP 上的邮件路由选择	5
1.4.3 UUCP 和 RFC 822 的混合	6
1.5 路径别名和映射文件格式	7
第 2 章 什么是邮件服务	10
2.1 邮件配置系统	10
2.1.1 系统组成	10
2.1.2 用户代理	13
2.1.3 邮件传输代理	13
2.1.4 邮件处理程序	13
2.1.5 域	14
2.1.6 邮件地址	16
2.1.7 信箱	17
2.1.8 别名	18
2.2 邮件服务剖析	19
2.2.1 邮件服务文件目录	19
2.2.2 邮件服务如何工作	21
2.2.3 邮件如何寻址	21
2.3 规划邮件服务	23
2.3.1 只有本地邮件的配置	23
2.3.2 有一个 UUCP 连接的本地邮件的配置	24
2.3.3 一个域、两个网络和一个路由的配置	25
2.3.4 两个域和一个网关的配置	25
第 3 章 Email 服务器简介	27
3.1 邮件服务器的功能	27
3.2 邮件服务器的结构	28
3.3 邮件服务器的组件	28
3.4 sendmail 程序	30
3.4.1 sendmail 配置文件 (sendmail.cf)	31
3.4.2 sendmail 配置表	32

第 4 章 FTGate 邮件服务器	34
4.1 FTGate 简介	34
4.1.1 什么是 FTGate	34
4.1.2 FTGate 的基本功能	34
4.1.3 FTGate 的特点	34
4.2 FTGate 启动与初始配置	35
4.2.1 快速启动	35
4.2.2 FTGateNT——Windows NT 服务	36
4.2.3 FTGate 应用主窗口	36
4.3 属性	38
4.3.1 拨号属性	39
4.3.2 发送属性	40
4.3.3 进度表	41
4.3.4 防护措施	43
4.3.5 日志记录	44
4.3.6 地址映射	46
4.3.7 域属性	47
4.3.8 SmartPop	48
4.3.9 端口属性	50
4.3.10 路径属性	51
4.3.11 高级发送属性	52
4.3.12 管理属性	53
4.3.13 注册属性	54
4.4 邮箱管理器	55
4.4.1 邮箱管理	56
4.4.2 配置用户邮箱	57
4.4.3 配置列表邮箱	58
4.4.4 列表邮箱的高级设置	59
4.4.5 编辑自动响应器	60
4.4.6 配置组邮箱	61
4.4.7 配置自动邮箱	62
4.4.8 编辑别名或新别名	63
4.5 命令处理器	63
4.5.1 命令处理器管理控制	64
4.5.2 一般命令处理器选项	65
4.5.3 用户邮箱命令选项	65
4.5.4 列表邮箱命令选项	66
4.5.5 自动响应器命令选项	66
4.6 使用邮件客户端	66
4.6.1 Eudora	67
4.6.2 Explorer Mail	68
4.6.3 Windows Messaging	68
4.6.4 Netscape Navigator Mail	69
4.6.5 Netscape Communicator	69
4.6.6 Pegasus Mail	70

4.6.7 Outlook Express.....	70
4.6.8 其他邮件客户端.....	71
4.7 使用代理服务器.....	71
4.7.1 邮件服务器与代理服务器在同一台机器上.....	71
4.7.2 邮件服务器与代理服务器在不同的机器上.....	72
4.8 常见问题 (FAQ).....	73
第 5 章 MailMax 邮件服务器.....	82
5.1 概述.....	82
5.1.1 MailMax 的特点.....	82
5.1.2 MailMax V3.0 的新特征.....	83
5.1.3 若干基本概念.....	84
5.2 系统设置.....	85
5.3 域设置.....	87
5.4 管理设置.....	88
5.5 连接设置.....	90
5.6 用户设置.....	91
5.6.1 用户管理器.....	91
5.6.2 创建用户信息.....	93
5.7 事件日志.....	97
5.8 统计.....	97
5.9 向导.....	98
5.10 队列管理器.....	99
5.11 常见问题 (FAQ).....	99
第 6 章 Exchange 服务器介绍.....	104
6.1 什么是 Exchange 服务器.....	104
6.2 Exchange 服务器的客户端软件.....	105
6.3 客户端和服务器之间是如何工作的.....	105
第 7 章 Unix 操作系统及其网络概述.....	108
7.1 Unix 操作系统简介.....	108
7.1.1 Unix 的历史发展.....	108
7.1.2 网络服务器操作系统的调查.....	110
7.1.3 两大自由操作系统——FreeBSD 与 Linux.....	111
7.2 Unix 网络简介.....	113
7.2.1 UUCP 网络.....	113
7.2.2 TCP/IP 网络.....	115
7.2.3 Linux 连网.....	121
第 8 章 Linux 平台上的邮件服务器.....	123
8.1 配置和运行 smail.....	123
8.1.1 UUCP 的设置.....	124
8.1.2 局域网中的设置.....	125
8.1.3 编译 smail.....	128

8.1.4	邮件投递模式.....	129
8.1.5	其他各种配置选项.....	130
8.1.6	消息路由选择和投递.....	131
8.1.7	路由消息.....	131
8.1.8	往本地地址投递消息.....	133
8.1.9	基于 UUCP 的传送.....	135
8.1.10	基于 SMTP 的传送.....	136
8.1.11	主机名限定.....	137
8.2	配置和安装 sendmail+IDA.....	137
8.2.1	sendmail+IDA 概述.....	137
8.2.2	配置文件简介.....	138
8.2.3	sendmail.m4 文件.....	138
8.2.4	sendmail+IDA 表格通览.....	143
8.2.5	安装 sendmail+IDA.....	148
8.2.6	邮件日常处理工作.....	153
8.2.7	匹配邮件程序.....	155
第 9 章	Solaris 平台上的邮件服务器使用.....	157
9.1	sendmail 概述.....	157
9.1.1	与外界的接口.....	157
9.1.2	sendmail 程序工作.....	158
9.1.3	消息标题编辑.....	159
9.1.4	配置文件.....	159
9.2	sendmail 实现.....	159
9.3	sendmail 参数.....	161
9.4	调整配置参数.....	162
9.4.1	时间值.....	162
9.4.2	发送模式.....	163
9.4.3	负载限制.....	163
9.4.4	日志级别.....	164
9.4.5	文件模式.....	164
9.5	配置文件介绍.....	165
9.5.1	sendmail 配置文件部分.....	165
9.5.2	配置文件方法.....	167
9.5.3	特别的标题行.....	169
9.5.4	地址重写规则.....	171
9.5.5	构造新的配置文件.....	177
9.6	命令行参数.....	179
9.7	配置选项.....	180
9.8	邮件处理程序标志.....	181
第 10 章	FreeBSD 平台上的邮件服务器.....	183
10.1	了解 FreeBSD 操作系统.....	183
10.1.1	FreeBSD 的功能特点.....	183
10.1.2	FreeBSD 的主要应用范围.....	185

10.2	网络基本配置	186
10.2.1	用 Sysinstall 配置网络	186
10.2.2	手工配置网络	188
10.2.3	使用基本网络工具	192
10.3	关于邮件服务器	194
第 11 章	实用 Email 服务程序介绍	196
11.1	邮件客户程序 elm	196
11.2	基于字符终端的 pine	198
11.3	使用 fetchmail 远程下载邮件	199
11.4	使用 procmail 设置邮件过滤	200
第 12 章	深入了解 sendmail	204
12.1	sendmail 安装指导	204
12.2	sendmail 工作原理	208
12.2.1	sendmail 如何工作	208
12.2.2	sendmail 守护进程	209
12.2.3	sendmail 脚本	210
12.2.4	sendmail 相关文件和目录	210
12.2.5	sendmail 命令	212
12.2.6	sendmail 网络设置	213
12.3	sendmail.cf 操作符注解	217
12.4	sendmail 常见问题解答 (FAQ)	220
第 13 章	FTPMail 服务器	228
13.1	如何使用 FTPMail	228
13.1.1	一个会话样本	229
13.1.2	使用不同地址的命令	232
13.1.3	会话命令	232
13.1.4	文件传送命令	233
13.2	安装 FTPMail	234
13.3	测试 FTPMail	237
13.4	管理 FTPMail	239
13.4.1	常规检查	239
13.4.2	限制 FTPMail	240
13.4.3	忽略邮件	241
第 14 章	Email 服务器安全性问题	242
14.1	理解系统安全	243
14.2	关于网络安全	245
14.2.1	防火墙系统	246
14.2.2	身份验证和授权	246
14.2.3	共享文件	250
14.2.4	限制超级用户的访问权限	251
14.2.5	使用特权端口	252

14.2.6 自动安全增强工具	253
14.3 Email 服务器安全理论基础	253
14.4 sendmail 服务器的安全性	255
14.5 Qmail 服务器的安全性	261
附录 A smail 样本配置文件注解	263
附录 B sendmail 样本配置文件	271
附录 C 有关术语表	292

第 1 章 认识电子邮件

自从第一个网络被设计出来，互联网最显著的用途之一就是电子邮件（electronic mail，通常简称为 email）。它是从通信线路的简单服务开始的，即将一个文件从一台机器复制到另一台机器上，并将该文件的拷贝添加到接收者的邮箱（mailbox）文件中。尽管不断增长的网络、复杂的路由需求以及它的不断增大的消息负载量，已经要求更加精心制作的方案。实际上这仍然是现在电子邮件所做的全部工作。不过目前已展开“多媒体邮件”的更多试验，这涉及到在邮件消息中包含图形和声音。人们已经设计出各种不同的邮件交换标准，但 Internet 站点上都采用标准 RFC（Request for comments）822。然而，RFC 822 仅是最伟大的公共奠基石，现有更多的标准已被构想出来以应付不断增长的需求。比如，数据加密、国际字符集的支持，以及多媒体邮件扩展（MIME）。另一个重要的标准 X.400 是由国际电话电报咨询委员会 CCITT（Consultative Committee on International Telephone and Telegraph）所定义。

在本章的开始将首先介绍电子邮件的基本概念，如什么是邮件，邮件地址看上去是怎样的，邮件处理系统是如何将消息送至接收者那里的，以及作为一个网络管理员要涉及一些什么样的问题。

1.1 邮件消息

一个邮件消息是由消息体（它是发送者所写的文本）、指明收信者的特定数据以及传输媒体等组成，很像信件上所见的信息。

这些管理用的数据可以分成两类；第一类是与特定传输介质相关的数据，如发信者和收信者的地址，因此它们被称为 envelope。在消息传输途径中，它们可能会被转换。第二类是处理邮件消息所需的其他数据，它们并不是针对任何传输机制的，比如消息的主题行（subject line）、所有接收者列表、以及消息发送的日期。在许多网络中，将这些数据添置到邮件消息中已成为标准，形成所谓的邮件头（mail header）。它与邮件体（mail body）之间相隔一个空行。通常习惯上会给邮件消息附加一个签名（signature），一般包含有关作者的信息，甚至一个笑话或者座右铭。它与邮件消息之间相隔含有“--”的一行。

UNIX 系统中的很多邮件传输软件使用 RFC 822 中指定的头（标题）格式。它最初是为了在 ARPANET 上的使用指定一个标准，但是由于它被设计成是与任何环境都无关的，它很容易地被其他网络采纳，包括许多基于(UUCP Unix-to-Unix-Copy)，Unix 系统之间的连网的网络。

在所有标准中，标题[头]由几行组成，并由换行符来分割。每行是由从第一列开始的字段名和由一个冒号和一个空格分割的字段本身组成。每个字段的格式和语义是随字段名

的不同而变化的。如果下一行是以一个制表符 (TAB) 开始的, 表示是标题字段的续行。各字段的顺序是随意的。一个典型的邮件标题格式是这样的:

```
From brewhq.swb.de!ora.com!andyo Wed Apr 13 00:17:03 1994
Return-Path: <brewhq.swb.de!ora.com!andyo>
Received: from brewhq.swb.de by monad.swb.de with uucp
        (Smail3.1.28.1 #6) id m0pqq1T-00023aB; Wed, 13 Apr 94 00:17 MET DST
Received: from ora.com (ruby.ora.com) by brewhq.swb.de with smtp
        (Smail3.1.28.1 #28.6) id <m0pqq0r-0008qhC>; Tue, 12 Apr 94 21:47 MEST
Received: by ruby.ora.com (8.6.8/8.6.4) id RAA26438; Tue, 12 Apr 94 15:56 -0400
Date: Tue, 12 Apr 1994 15:56:49 -0400
Message-Id: 199404121956.PAA07787@ruby
From: andyo@ora.com (Andy Oram)
To: okir@monad.swb.de
Subject: Re: Your RPC section
```

通常, 所有所需的标题字段都是由所使用的邮件程序界面产生, 如 elm、pine、mush 或 mailx 等程序。不过有些字段是可选的, 可以由用户自己加上去。例如, elm 允许编辑部分消息头 (标题), 而其他的则是由邮件传输软件加上去的。常用的标题字段以及它们的含义如下所示:

From:	含有发送者的 email 地址, 可能还有“真实名字”。
To:	这是接收者的地址。
Subject:	用几个词描述邮件的内容, 这是起码应该做的。
Date:	邮件发送的日期。
Reply-To:	发送者指定接收者回信所发往的地址。如果有几个账号, 且只想在常用的账号下接收大量的邮件, 此时这个字段是很有用的。该字段是可以选择的。
Organization:	产生邮件的机器所属的组织。如果机器是属于个人的, 可以空着该字段, 或者插入“个人”(“private”)或某些完全无意义的东西。这个字段是可选字段。
Message-ID:	邮件生成系统的邮件传输程序产生的字符串。每条消息只有惟一的 ID。
Received:	处理邮件的每一个站点 (包括发送者和接收者的机器) 在标题中插入的字段, 给出它的站点名称、一个消息 id、收到该消息的时间和日期、从哪个站点来的以及使用了哪个传输软件。据此, 可以跟踪消息走过的路由, 如果出了差错可以追究有关人士的责任。
X-anything:	以 X-开头的标题行对任何邮件相关的程序都是不起作用的。它是用于实现还没有写入或不可能写入 RFC 的额外的特性的。例如, 它用于 Linux Activists 邮件列表中, 是用 X-Mn-Key: 标题字段来选择频道的。

这个结构的一个例外是最开头的一行。它以关键字“From”开始, 紧接着是一个空格而不是一个冒号。为了与普通的“From:”字段相区别, 它通常被引用为“From_”。它包含消息所参与的 UUCP 大宗路径形式的路由, 最后一台接收消息的机器处理消息的时间和日期, 以及一个指明从哪台主机接收来的可选部分。由于每个处理过这个消息的系统都会生成这个字段, 所以它通常包含在信封数据下。

因此, “From_”字段与某些老式的邮件程序是向后兼容的, 但已不再经常使用, 除了

邮件用户界面程序还要依靠它来确定用户邮箱中一个消息的开始处。而且为了消除以“From:”开始的消息体中的行所带来的潜在问题，在它之前放置“>”以避免任何这样的现象出现，这已经成为一个标准方法。

1.2 邮件的投递

一般来讲，要使用一个邮件界面程序如 mail 或 mailx 来编写邮件；或者使用更为复杂的如 elm、mush 或 pine。这些程序统称为邮件用户代理（mail user agents），或简称 MUA。如果发出了一个邮件消息，那么在大多数情况下界面（接口）程序会将它传递给另一个程序去进行投递。这个程序称为邮件传输代理（mail transport agent），或简称 MTA。在某些系统中，对于本地的和远程的投递有不同的邮件传输代理程序，在其他系统中仅有一个。远程投递的命令通常称为 rmail，其他的称为 lmail（如果存在的话）。

当然，本地邮件的投递仅仅是将进来的消息附加到接收者的邮箱里。通常，本地 MTA 是知道别名（将接收者的地址设置成指向其他的地址）和转发的（将用户的邮件重定向到某个其他的目的地）。同样，不能投递的消息通常必须反弹回来（bounced），也就是附带某些出错信息返回给发送者。

对于远程投递操作，所用的传输软件依赖于连接的方式。如果一个邮件必须在使用 TCP/IP 的网络上投递的话，常常会使用 SMTP。SMTP 表示简单邮件传输协议（Simple Mail Transfer Protocol），是在 RFC 788 和 RFC 821 中定义的。SMTP 通常直接连接至接收者的机器上，与远端的 SMTP 后台程序（daemon）协商消息的传输。

在 UUCP 网络中，邮件通常不是直接投递的，而是通过一系列的中间系统转发到目的主机上的。为了在 UUCP 链接上发送一个消息，MTA 通常使用 uux 在转发的系统上执行 rmail，并且将消息通过标准输入送给转发系统。

由于这是对每个消息分别操作的，这将在主要的邮件中心产生可观的工作负载。数百个小文件耗费不成比例的大量磁盘空间，形成混乱的 UUCP 假脱机队列（这是因为磁盘空间的分配通常是以 1024 字节的块为单位的。所以，即使一个最多 400 字节的消息也要吃掉整整 1KB 的空间）。因此某些 MTA 允许以一个批处理文件从远程系统收集几个消息。如果使用了直接 SMTP 连接，那么这个批处理文件通常含有本地主机要发出的 SMTP 命令，称为 BSMTP（batched SMTP）。此时这个批处理会被发送给远程系统中的 rsmtp 或 bsmtp 程序，远程系统将像正常的 SMTP 连接一样来处理输入。

1.3 Email 地址

对于电子邮件，地址是由处理该邮件的机器的名字和这个系统能够识别的用户的代号组成。这可以是接收者的登录名，也可以是任何别的代号。

一个机器解读邮件地址的方法，即如何将这个名字与接收者的用户名结合在一起，消

息在哪个站点最终结束，这很大程度上依赖于所连接的网络。

Internet 站点是符合 RFC 822 标准的，它采用 `user@host.domain` 这样的表示法，这里 `host.domain` 是主机的完全符合标准的域名（fully qualified domain name），当中的符号 `@` 称为“在”（“at”）符号。因为这个表示法不包含有到目的主机的路由，而是给出了惟一的主机名，这称为一个绝对地址。

在最早的 UUCP 环境中，流行的形式是 `path!host!user`，这里 `path` 描述了在到达目的 `host` 之前消息经过的一系列的主机。这种结构称为 `bang path` 表示法，因为一个感叹号近似地被称为一个“bang”。今天，许多基于 UUCP 的网络已经采用了 RFC 822，并且能够理解这种地址类型。

不过目前这两种地址类型还不能很好地融合在一起。假设有一个地址 `hostA!user@hostB`，这时并不清楚这个路径上的“`@`”符号是否是优先的，即不清楚是否必须将消息发送到 `hostB`，再邮递到 `hostA!user`，还是先发送到 `hostA`，再转发到 `user@hostB`。

然而有一种与 RFC 822 一致的方法可以指定路由：`<@hostA,@hostB:user@hostC>` 表示在 `hostC` 上的 `user` 地址，这里将经过 `hostA` 和 `hostB` 到达 `hostC`。这种地址类型常常称为 `route-addr` 地址。

还有一个地址操作符“`%`”，例如 `user%hostB@hostA`。该地址首先发送到 `hostA`，它将最左边的百分号扩展成为一个“`@`”符号（仅在此情况下），地址变成 `user@hostB`，邮件程序将很顺利地消息转发到 `hostB`，再投递给 `user`。这种类型的地址有时被称为“Ye Olde ARPANET Kludge”，它的使用是不正式的。然而，许多邮件传输代理产生这种地址类型。

其他的网络还有不同的地址方案。例如，基于 DECnet 的网络使用两个冒号作为地址的分隔符，产生一个 `host::user` 地址（当从一个 RFC 822 环境试图到达一个 DECnet 地址时，可以使用“`host::user`”`@relay`，这里 `relay` 是一个已知的 Internet-DECnet 中继）。最后，X.400 标准使用了一个完全不同的方案，通过使用一族“属性-值”对来描述一个接收者，就像国家和组织一样。

在 FidoNet 上，每个用户的确定是用一个代码如 `2;320/204.9`，由四个数字组成，表示区（zone）（2 表示欧洲）、网络（net）（320 代表巴黎和郊区）、节点（node）（代表本地 hub）和点（point）（代表个人用户 PC）。FidoNet 的地址可以被映射到 RFC 822 上；上面的地址可以写成 `Thomas.Quinot@p9.f204.n320.z2.fidonet.org`。这样的域名不大容易记忆。

使用这些不同的地址类型有某些隐含方面，这将在以下几节中讨论。然而，在一个 RFC 822 环境中，除了像 `user@host.domain` 这样的绝对地址，很少会用到其他的地址类型。

1.4 邮件的路由

定向一个消息到接收者主机的过程称为路由选择（routing）。除了寻找到一条从发送站点到目的地的路径以外，它还包括错误检测以及速度和代价优化。

UUCP 站点处理路由选择的方法与 Internet 站点有很大的不同。在 Internet 上，定向数据到接收者的主机（由它的 IP 地址确定）的主要任务是由 IP 的网络层来完成的，而在 UUCP 区中，路由必须由用户来提供，或者是由邮件传输代理生成。

1.4.1 Internet 上的邮件路由选择

在 Internet 上，它完全依赖于目的主机是否要执行特定的邮件路由选择。默认的是通过查找目的主机的 IP 地址，直接将消息投递到目的主机去，而将实际数据的路由选择工作给 IP 传输层去做。

许多站点通常会想要指引所有入站的邮件到一个高效稳定的邮件服务器中，这个服务器能够处理所有这些数据流量，并让它在本域分发邮件。为了宣告这种服务，站点在 DNS 数据库中为它们的本地域公布一个所谓的 MX 记录。MX 代表邮件交换器 (Mail Exchanger)，基本的意思是表明服务器主机很愿意为本域中的所有机器充当邮件转发器。MX 记录也可以用于处理自己没有连接到 Internet 上的主机的交通流量，比如 UUCP 网络或者是携带着机密信息的公司网络上的主机。

MX 记录也有一个与之相关的优先权 (preference)。这是一个正整数。对于一台主机如果存在几个邮件交换器的话，那么邮件传输代理将会试图把消息传输到具有最低优先权值的交换器上，如果不行再尝试一台具有高一级优先权值的主机。如果本地主机本身就是目的地址的一个邮件交换器，它就不必将消息转发到优先权值比自己高的任何主机去：这是避免邮件循环 (loops) 的一个安全方法。

假设有一个叫 Foobar 的公司，想要由他们称之为 mailhub 的机器来处理他们所有的邮件，那么就应该在 DNS 数据库中有一个如下的 MX 记录：

```
foobar.com IN MX 5 mailhub.foobar.com
```

这表明 mailhub.foobar.com 是一个 foobar.com 上的有优先权值 5 的邮件交换器。一个希望把消息投递到 joe@greenhouse.foobar.com 的主机将检查 foobar.com 的 DNS，会发现 MX 记录指向 mailhub。如果此时没有优先权值小于 5 的其他 MX 存在，这个消息将被投递到 mailhub，然后被分发给 greenhouse。

上面实际上仅是 MX 记录如何工作的一个轮廓。有关 Internet 上邮件路由选择的更多信息，请参考 RFC 974。

1.4.2 UUCP 上的邮件路由选择

UUCP 网络上的邮件路由选择要比 Internet 上的复杂得多，因为传输软件本身不执行任何的路由选择功能。在早期，所有的邮件得用 bang paths 来寻址。bang paths 指定了一张主机的列表，各主机名用感叹号分开，通过这些主机来转发消息，后接用户名。为了将一封信寻址到名为 moria 的机器上的 Janet 用户，就得使用路径 eek!swim!moria!janet。这将会把邮件从主机发送到 eek，再从 eek 发送到 swim 最后到 moria。

这种技术明显的不足之处是它需要记住很多有关网络拓扑的细节、快速链接等等。更糟糕的是，网络拓扑的变化（比如链接被删除了或主机被移走了）可能会导致消息传输的失败，简单的原因只是由于不知道网络结构变化了。还有如果移到了不同的地方，很可能就要更新所有这些路由了。

然而还有一件事，就是不明确的（多义的）主机名的存在使得路由选择成为必须。例如，假设有两个站点名字都叫 moria，一个在美国，另一个在法国。那么 moria!janet 指的是哪一个呢？这可以通过指明到达 moria 的路径弄清楚。

消除多义主机名的第一步是 UUCP 映射计划组（The UUCP Mapping Project）的成立。它位于 Rutgers 大学，对所有官方的（正式的）UUCP 主机进行登记注册，并记录下他们的 UUCP 邻居和他们的地理位置，确信没有哪个主机名被使用了两次。由映射计划组收集的信息作为 Usenet Maps 公布出来，它会定期地通过 Usenet 进行发布（在 UUCP 映射计划登记的站点的映射是通过新闻组 comp.mail.maps 发布的，其他的组织也会公布他们网络的映射）。Map 中一个典型的系统条目格式如下（其中注释语句已被删除了）：

```
Moria
      bert (DAILY/2) ,
      swim (WEEKLY)
```

这个条目说明 moria 有一个至 bert 的连接——moria 每天对它呼叫两次，和一个到 swim 的连接——moria 每星期对它呼叫一次。下一节将更详细地讨论映射文件的格式。

使用映射文件中提供的连接信息，可以自动地生成从主机到达任何目的站点的全路径。这个信息一般存储在 paths 文件中，有时也被称为路径别名数据库（pathalias database）。假设映射表明可以通过 ernie 到达 bert，那么从上述映射中为 moria 生成路径别名条目应该是这样的：

```
moria          ernie!bert!moria&s
```

如果现在给出一个目的地址 janet@moria.uucp，MTA 就会取得上面的路由，并将消息用 bert!moria!janet 作为信封地址发送到 ernie。

然而，从完整的 Usenet 映射来建立一个 paths 文件并不是一个好主意。其所提供的信息通常是有误导的，而且有时是过时的。因此，只有很少几个主要的主机使用完整的 UUCP 世界映射来建立它们的 paths 文件。大多数的站点仅仅维护着它们周邻站点的路由选择信息，并将需要发送到它们数据库中找不到站点的其他邮件，发到一个有更完整路由选择信息的灵敏主机上。这种方案被称作灵敏主机路由选择（smart-host routing）。只有一个 UUCP 邮件连接的主机，即所谓的页站点（leaf sites），本身不做任何的路由选择，它们完全依赖于它们的灵敏主机。

1.4.3 UUCP 和 RFC 822 的混合

迄今为止，针对 UUCP 网络邮件路由选择问题最好的解决方案是在 UUCP 网络中采用域名系统。当然，不能在 UUCP 上查询一个名字服务器。然而，许多 UUCP 站点已经在自己内部组成了与路由选择相协调的小型域。在映射中，这些域将一台或两台主机作为他们的邮件网关，所以在域中不需要对每台主机都有一个映射条目。网关将处理所有流入和流出域的邮件。在域中的路由选择方案对于外界来说是完全看不见的。

这个方法与上述的灵敏主机路由选择方案配合得很好。全局路由选择信息仅由网关来