

第13章 smail的设置和运行

本章打算对 smail 的设置和功能进行一番概述。尽管 smail 的行为在很大程度上兼容于 sendmail，但它们的配置文件却是完全不同的！

主配置文件是 /usr/lib/smail/config。针对每个人建立的站点，必须对这个文件进行编辑，以反映出自己的实际情况。但是，假如建立的只是一个 UUCP 叶站点，那么要做的事情就简单得多。另外，还有可能要用到对路由及传输选项进行配置的其他文件。本章也会简要地介绍它们。

默认情况下，smail 会立即处理和投递输入的所有邮件。但是，假如站点的事务量非常高，也可考虑让 smail 在队列中先收集好所有消息，再定时进行处理，不一定非要“立即”处理。

在一个 TCP/IP 网络中对邮件（消息）进行处理时，smail 会经常用一种 daemon 模式运行。换言之，系统启动时，会自 rc.inet2 调用它。然后，它“默默”地进入后台，等候自 SMTP 端口（通常是端口号 25）进入的 TCP 连接。假如站点非常繁忙，这样做便显得相当有用，因为用不着为进入的每个连接，都单独启动一个 smail。当然，你也可以采取另一种做法，那便是让 inetd 管理 SMTP 端口，并在该端口上建立了一个连接之后，立即调用 smail。

smail 本身有大量标志可供设置，以便对其行为进行全方位控制。但假如要在这里全部列出来，恐怕也没什么好处。幸好，smail 支持大量标准的工作模式。调用它的时候，只需利用一个特殊的命令名（别名），比如 rmail 或者 smtpd 等，就可以启用这些模式。

通常，这些别名是针对 smail 二进制程序设置的一些“符号链接”。以后讨论 smail 的各项特性时，大家会逐渐接触到它们中的绝大多数。

任何情况下，你都应该有两条指向 smail 的链接：即 /usr/bin/rmail 和 /usr/sbin/sendmail（这是根据“文件系统”而定的新 smail 标准位置，另一个常见的位置是 /usr/lib）。在利用 elm 之类的用户代理编写和发送邮件消息时，这一消息应该随命令行上指定的收件人清单一起，送入 rmail 进行投递。对通过 UUCP 收到的邮件来说，同样如此。但是 elm 的几个版本调用的都是 /usr/sbin/sendmail，而不是 rmail，所以对你而言，这两者都必须有。例如，如果你将 smail 保存在 /usr/local/bin 内，在外壳脚本输入下面两行：

```
# ln -s /usr/local/bin/smail /usr/bin/rmail
# ln -s /usr/local/bin/smail /usr/sbin/sendmail
```

如果你想进一步了解 smail 的配置情况，请参考手册 smail(1) 和 smail(5)。如果你喜爱的软件包内没有这两个手册页，则可参考 smail 的源代码。

13.1 UUCP 的设置

要想在一个纯 UUCP 的环境中使用 smail，基本的安装过程很简单。首先，你必须确保建立与前面提到的 rmail 和 sendmail 的两个象征性链接。如果希望从其他站点接收 SMTP 批处理数据，还必须为 rsmtp 建立与 smail 的一个链接。

在 Vince Skahan 发布的 smail 版本中，可找到一个示范性的配置文件。它的名字叫做

config.sample，保存在/usr/lib/smail目录下。必须将其复制到 config文件中，并对其进行恰当的编辑，以反映出与自身站点情况相符的值。

假定你的站点名为 swim.twobirds.com，并在UUCP的映射表中注册为 swim。同时，你的智能主机名为 Ulysses。那么，配置文件的内容应该像下面这个样子：

```
#
# Our domain names
visible domain=two.birds:uucp
#
# Our name on outgoing mails
visible name=swim.twobirds.com
#
# Use this as uucp-name as well
uucp name=swim.twobirds.com
#
# Our smarthost
smart host=ulysses
```

其中，第一条语句告诉 smail你的站台从属于哪个域。请在这里插入它们的名字，相互间用冒号分隔。假如你的站点已在 UUCP映射表中完成了注册，还应在此添加 uucp。传来一条邮件消息后，smail会利用hostname(2)系统调用，来判断你的主机的名字是什么。同时，根据这个主机名，依次跟踪该列表中的所有名字，从而检查接收人的地址。假如地址与这些名字中的任何一个相符，或者与未认证的主机名相符，则认为是一个本地接收人。随后，smail会试图将邮件投递给本地主机上的一名用户或别名。否则的话，就认为接收人在远程主机上，并试图将邮件投递至目标主机。

在visible_name中，应包括你的站台的一个完整形式的域名，我们想在发到外面的邮件中使用这个域名。在所有“外出”的邮件中生成发送者的地址时，便要用到这个名字。必须使用smail能将其与本地主机对应起来的一个名字（比如与 visible_domain属性中列出的域对应的某个主机名）。否则的话，一旦有人直接回复从你那里发出的邮件，邮件便会被发至不知所云的地方！

最后一条语句设置智能主机路由选择要用到的路径。在我们这个示范性的设置中，smail会将发至远程地址的任何邮件都转发给智能主机。smart_path属性中指定的路径将作为到智能主机的一条路由使用。由于邮件会通过 UUCP投递，所以在属性中，必须指定你的 UUCP软件已经知道的一个系统。请参考第 11章，了解如何让UUCP“知道”一个站点。

在上述文件中，还用到了一个我们尚未解释的选项——uucp_name。之所以要使用这个选项，原因是：在默认情况下，smail会将hostname(2)返回的值用于 UUCP特有的一些东西，如 From_header行中给出的返回路径等等。假如你的主机名尚未经由 UUCP映射项目完成注册，那么应告诉 smail，令其换用完整形式的域名（举个例子来说，假定你的主机名是 Monad，但尚未在映射表中完成注册。此时，如果映射表中已经有一个名为 Monad的站点，那么对发给 monad!root的任何邮件而言，包括从你的某个直接 UUCP邻居发出来的邮件，它们都会投递到那个Monad。显然，这种情况不是我们所希望看到的）。要想解决这个问题，唯一的办法就是在配置文件中，设置一个 uucp_name选项。

在/usr/lib/smail目录中，还有另一个值得注意的文件，名为 paths.sample。它同样是一个

示范文件，告诉大家一个路径文件应该如何配置。但是，除非同时拥有到多个站点的邮路链接，便用不着对它进行配置。另一方面，假如确实拥有多条邮路，便必须自行编制这样的文件，或者根据 Usenet 映射表来生成一个。路径文件的详情，本章稍后还会详加解释。

13.2 在局域网环境中的设置

在自己的站点中，假如同时包括几个主机，相互间通过 LAN 连接，便必须指定其中的一个主机，令其负责管理与外部世界的 UUCP 连接。在 LAN 内各主机之间，通常还想通过 TCP/IP 协议，实现邮件的交换。假定我们现在又回到 Virtual Brewery，并将 vstout 设为 UUCP 网关。

在一个连网环境中，最好将所有用户的邮箱都放在单独一个文件系统中。然后，通过 NFS 装载，令其适用于其他所有主机（被它们“看到”）。这样一来，用户便可自由地改变自己使用的机器，同时不必“搬”着自己的邮箱到处乱走（或者更糟，每天早上要检查三、四台机器，看看是否有自己的新邮件）。考虑到这种情况，我们还想让发件人的邮件地址独立于实际写入邮件的那台机器（机器可以不同，地址只有一个）。一种常见的做法是在发件人的地址中，使用域名本身，而不要使用主机名！例如，假定有一位名为 Janet 的用户，它的邮箱地址就应该是 janet@vbrew.com，而不应是 janet@vale.vbrew.com。后面，我们会详细解释如何对服务器进行配置，令其将域名识别成一个在自己站点内有效的名字。

要想将所有邮箱都集中放在一个主机上，另一个办法是使用 POP 或 IMAP 协议。POP 的全称是 Post Office Protocol，即“邮局协议”。用户通过一个简单的 TCP/IP 连接，便能实现对自己的邮箱的访问。而另一方面，IMAP 的全称是 Interactive Mail Access Protocol（交互式邮件访问协议）。IMAP 与 POP 非常相似，具有 POP 的所有功能。但是，IMAP 最重要的特点便是“交互式”。用户可像管理本地文件夹一样，组织与管理自己在远程邮件服务器上的邮件文件夹。无论 IMAP 还是 POP，在 Linux 平台上，都有移植版本。大家可自 sunsite.unc.edu 下载，目录是 /pub/Linux/system/Network。

13.2.1 编写配置文件

对 Brewery 来说，我们打算像这样进行配置：除邮件服务器 vstout 本身之外的所有主机都将“外出”的邮件路由至服务器，使用智能主机路由方式。而就 vstout 自己来说，它将所有外出的邮件发给真正的智能主机，令其对 Brewery 的所有邮件进行路由选择。这个真正的智能主机名为 Moria。

对所有主机（vstout 除外）来说，它们的标准配置文件如下：

```
#
# Our domain:
visible domain=vbrew.com
#
# What we name ourselves
visible name=vbrew.com
#
# Smart-host routing: via SMTP to vstout
smart path=vstout
smart transport=smtp
```

这与我们为纯UUCP站点使用的配置非常相似。最主要的差别在于，用来将邮件发给智能主机的传送层协议是SMTP（这是理所当然的）。visible_domain属性指示smail在外出的所有邮件中使用域名，而不是使用本地主机名。

而在UUCP邮件网关vstout上，配置文件却像下面这样（有少许的不同）：

```
# Our hostnames:
hostnames=vbrew.com:vstout.vbrew.com:vstout
#
# What we name ourselves
visible name=vbrew.com
#
# in the uucp world, we're known as vbrew.com
uucp name=vbrew.com
#
# Smart transport: via uucp to moria
smart path=moria
smart transport=uux
#
# we're authoritative for our domain
auth domains=vbrew.com
```

显然，配置文件利用了一种不同的方案，来告诉smail本地主机叫什么名字。它没有给出一份域列表，令其找出采用了系统调用的主机名，而是显式指定了一个列表。上面的列表表中包含了完全资格的和未经验证的主机名以及各自的域名。这样一来，smail就能够把janet@vbrew.com识别为本地地址，并将邮件消息投递给Janet。

auto_domain参数命名域名，vstout将被视为这个域内的主机。也就是说，如果smail收到其目标地址是host.vbrew.com（其中的主机没有对现成的本地主机进行命名）的任何邮件，它都会将其驳回并将其返回发送端。如果没有这个条目，此类邮件消息就会被发送到智能主机，由它将邮件消息返回vstout，直到超过最大网关跳计数时才将其丢弃。

13.2.2 运行smail

首先，必须决定是将smail作为一个独立的后台程序来运行，还是令inetd来管理SMTP端口，并且只在某一客户机发出连接请求时，才调用smail。通常情况下，都宁愿考虑在邮件服务器上操作后台程序，因为这样的话，远远比一次又一次地针对每次连接调用smail简单得多。由于邮件服务器还会把多数进入的邮件直接投递给用户，所以你要在其他的多数主机上选定inetd操作模式。

不管你针对每台主机选择的操作模式是什么，都必须保证你的/etc/services文件内有下列条目：

```
smtp 25/tcp # Simple Mail Transfer Protocol
```

这个条目定义了TCP端口号，smail将利用这个端口进行SMTP对话。已分配编号RFC对此定义的标准端口号是25。

在后台程序模式下运行时，smail将把自己置入后台运行并等待SMTP端口上发生的连接请求。一有连接，它就转向并利用对等体进程，管理SMTP对话。通常情况下，启动smail后台程序的过程是这样的：利用下面的命令，从rc.inet2脚本调用它：

```
/usr/local/bin/smmail -bd -q15m
```

-bd标记用于打开后台程序模式，而-q15m标记则是令其每隔15min，就对消息队列中累积的消息进行处理。

如果你想用inetd来代替它，你的/etc/inetd.conf文件内就应该有下面这一行：

```
smtp stream tcp nowait root /usr/sbin/smtpd smtpd
```

smtpd应该是一个指向 smail二进制文件的象征性链接。记住，你必须令 inetd重新读取inetd.conf，具体作法是在作出改动之后，向它发送一个HUP信号。

后台程序模式和inetd模式是彼此不相容的。如果你在后台程序模式下运行 smail，就应该确定在inetd.conf文件内批注出用于smtp服务的所有行。同样地，在令inetd管理smail时，一定要保证rc.inet2没有启动smail后台程序。

13.3 故障排除

如果你的配置有错，这里有大量的特征可帮助你找出错误的根源。第一个需要查看的是smail的日志文件。它们保留在/var/spool/smmail/log中，其文件名分别是logfile和paniclog。前者列出了所有的事务，后者则只列出了与配置错误及行为相关的错误消息。

logfile文件内的典型条目如下所示：

```
04/24/94 07:12:04: [m0puwU8-00023UB] received
|           from: root
|           program: sendmail
|           size: 1468 bytes
04/24/94 07:12:04: [m0puwU8-00023UB] delivered
|           via: vstout.vbrew.com
|           to: root@vstout.vbrew.com
|           orig-to: root@vstout.vbrew.com
|           router: smart host
|           transport: smtp
```

这个条目展示了从root到root@vstout.vbrew.com的邮件消息已经通过SMTP被准确投递到vstout。

smail不能投递的消息将在日志文件内生成一个类似的条目，但错误消息取代了已投递部分：message instead of the delivered part:

```
04/24/94 07:12:04: [m0puwU8-00023UB] received
|           from: root
|           program: sendmail
|           size: 1468 bytes
04/24/94 07:12:04: [m0puwU8-00023UB] message instead of the delivered part
|           via: vstout.vbrew.com
|           to: root@vstout.vbrew.com
|           orig-to: root@vstout.vbrew.com
|           router: smart host
|           transport: smtp
```

上面的错误一般发生在这种情形下：smail准确识别出消息应该被投递到vstout，但不能和vstout上的SMTP服务建立连接。如果出现这种情况，就表明如果不是存在配置问题，就是你的smail二进制程序不支持TCP。

这个问题不是很常见的。即使有的发布版本中，有已预先编辑好的 smail二进制程序，这

些二进制程序中没有提供 TCP/IP 连网支持。如果碰到这种情况，你必须自行编辑 smail。安装 smail 之后，就可检查它是否提供 TCP 连网支持了，具体做法是 telnet 到你计算机上的 SMTP 端口。下面展示了一个到 SMTP 服务器的成功连接（你的输入应该像这里标记的一样）：

```
telnet localhost smtp
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^['.
220 smail.twd.de Smail3.1.20.1 66 ready at Sun, 23 Jan 94
13:36:00
QUIT
221 smail.twd.de closing connection
```

如果这个测试不能生成 SMTP 旗标（以 220 代码开头的那一行），在你自行从头编辑 smail 之前，首先要确定你的配置是否真的准确无误，详情稍后讨论。

如果你在使用 smail 时碰到了问题，不能通过 smail 产生的错误消息来找出问题所在，这时，就需打开调试消息。具体作法是利用 -d 标记，后面可选择性地跟上一个比较长的参数（标记和参数之前，不能有空格）。这样，smail 就会在屏幕上打印出其操作报告，你就可以从中了解出错原因了。

如果实在没办法，只有在 Rogue 模式下，调用 smail，具体做法是在命令行指定 -bR 选项。然后，屏幕上就出现这样的消息：“输入 giant 域的邮件消息和 RFC 标准名册。试着令其降到 26 这个协议级别并回滚”。虽然这个选项没有解决你的实际问题，但至少给了你某些安慰。

smail 的编译

如果你确定 smail 二进制程序中没有 TCP 连网支持，只有想办法得到其源代码。它可能包含在你的系统内（如果你通过光盘获得它的话）。不然，就可从网络上通过 FTP，从 <ftp://ftp.planix.com/pub/Smail/> 下载。

编辑 smail 时，最好从配置文件集着手，这些文件源于 Vince Skahan 编写的 newspak 系统。为了用 TCP 连网驱动程序进行编辑，必须把 conf/EDITME 内的 DRIVER_CONFIGURATION 宏设置为 bsd-network 或 arpa-network。前者适用于局域网安装，但因特网要求的则是 arpa-network。两者之间的区别在于后者有一个特殊的用于 BIND 服务的驱动程序，这个服务能够识别 MX 记录，前者却没有。

13.4 邮件投递模式

正如我们在前面提到的那样，smail 能立即投递消息或把它们排成队列，等待处理。如果选择对消息排队，smail 就会把所有邮件保存在 /var/spool/smail 下面的消息目录内。在得到明确提示之前，是不会对队列中的消息进行处理的（这就是所谓的“运行队列”）。

投递模式有三种；前台、后台和排队等候。前台模式下，将立即处理进入的邮件消息；后台模式下，将消息交给接收进程的子进程投递，父进程在分叉之后立即退出。排队等候模式则不言而喻。通过把 config 文件内的 delivery_mode 属性设置为其中之一，就可选择一种投递模式了。如果在 config 文件内设置布尔变量 queue_only 的话，对进入邮件消息来说，不管有没有这个选项，始终都会排入队列等候处理。

如果你打开队列，就必须保证定期检查队列；时间间隔可能是 10 或 15min。如果以后台模

式运行 `smail`，就必须在命令行上增加选项 `-q10m`，表示每隔 10min，就处理一次队列。另一种办法是在这些时间间隔内，从 `cron` 调用 `runq`。`runq` 应该是指向 `smail` 的一个链接。

随 `-bp` 选项调用 `smail`，就可显示当前的邮件队列了。同样地，还可以令 `mailq` 成为指向

```
$ mailq -v
m0pvB1r-00023UB From: root (in /var/spool/smail/input)
Date: Sun, 24 Apr 94 07:12 MET DST
Args: -oem -oMP sendmail root@vstout.vbrew.com
Log of transactions:
Xdefer: <root@vstout.vbrew.com reason: (ERR 148) transport smtp:
connect: Connection refused
```

`smail` 的一个链接，并调用 `mailq`：

以上展示了消息队列中的一条单独的消息。事务日志（如果为 `mailq` 指定 `-v` 选项的话，这是唯一能显示出来的内容）可能会给出为什么该消息还在等候投递的原因。如果以前不曾尝试过投递这封邮件消息，就不会出现其事务日志。

即使你没有采用队列，`smail` 在发现暂时不能立刻投递消息时，有时候会把消息投入队列。对 SMTP 连接来说，其原因可能是因为收件人的主机是不可抵达的；但在文件系统被发现满了时，消息可能会延迟投递。因此，你应该大约每隔一小时（利用 `runq`）就运行一次队列，不然，延迟的消息将一直“赖”在队列中。

13.5 其他配置选项

`config` 文件内，有相当多的选项供你设置，虽然有些选项有用，但与运行 `smail` 没有本质上的联系，所以不是我们讨论的主题。相反地，我们只提少数没有理由不用的选项：

`error copy postmaster`——如果设置这个布尔变量，任何错误都会产生一条发送给邮局负责人的消息。通常情况下，这只针对由于配置不当而引起的错误。这个变量可以被打开，具体做法是把它放入 `config` 文件，并在其前面加一个“+”号。

`max hop count`——如果消息的网关跳计数（也就是说邮件传输经过的主机数）等于或大于这个数，远程投递尝试就会以出错告终，向发件人返回一条错误消息。这是用来防止消息永久循环的。跳计数一般是从邮件头的 `Received:` 字段计算而来的，但也可在命令行上利用 `-h` 选项，手动进行设置。这个变量的默认设置是 20。

`postmaster`——邮局负责人的地址。如果邮局负责人的地址不能解析成一个有效的本地地址，这就是最后的“法宝”。默认设置是 `root`。

13.6 消息路由和投递

`smail` 把邮件投递划分到三类不同的任务中，亦即三个功能模块：路由器、定向器（`director`）以及传输模块。

其中，路由器模块负责解析所有远程地址，判断一条消息接下去应投给哪个主机，以及使用哪种传输协议。以具体的链接为基础，可选用像 `UUCP` 或 `SMTP` 这样的传输协议。

至于本地地址，则发给定向器，由它决定是转发还是别名转换。举个例子来说，指定的地址可能是一个别名，或者是一个邮件列表。另外，用户可能想把自己的邮件转换到另一个地址。假如最终判断出来的地址位于远程，则将其传送给路由器模块，作进一步的路由选择

处理。否则的话，就为其分配一种传输协议，进行本地投递操作。迄今为止，最常见的一种操作是将消息投递到一个邮箱，但也可以将消息（邮件）输送给一个命令，或者为其追加“附件”（随正文一起发送的文件）。

最后，传输模块根据选定的投递方法，完成邮件的实际投送操作。它会试着发出邮件。假如操作失败，那么要么生成一封回函说明情况，要么等待下一次继续尝试。

使用smail，我们可以非常灵活地配置所有这些任务。针对每一种任务，都同时提供了大量驱动程序。可根据自己的实际情况，在这些驱动程序中，选出自己需要的一种。我们要通过不同的文件，向smail描述它们。这些文件的名称分别是路由器、定向器和传输模块，均位于/usr/lib/smail下面。

假如这些文件不存在，就会自动选择一些合理的默认值。这些默认值适用于采用SMTP或UUCP进行邮件传输的大多数站点。假如你想修改smail的路由策略，或打算对一种传输协议进行改动，便需从smail的源码中，找到相应的示范文件（默认配置文件可在源码目录的samples/generic内找到）。随后，将示范文件复制到/usr/lib/smail。最后，根据自己的实际需要，对它们进行修改。本书的附录也给出了这些示范配置文件。

13.7 消息的路由

smail收到一条消息（邮件）之后，首先会检查目的地是本地主机，还是一个远程站点。假如目标主机地址是config中配置好的某个本地主机名，那么消息会被传送给定向器模块。否则的话，smail就会将目标地址传给大量路由器驱动程序，找出可向其转发消息的那个路由器。可在路由器文件中对它们进行描述；假如文件不存在，同样地会使用一系列默认路由器。

目标主机会依次传递给所有路由器，直到发现“最相符”的那个路由器为止。举个例子来说，假定有一条消息投递给joe@foo.bar.com。此时，假设有一个路由器知道bar.com域内所有主机的一条默认路由；但与此同时，另一个路由器却掌握了foo.bar.com本身的信息。由于后者与那条消息的目标地址“更为相符”，所以最终选定的便是它，而不是前一个“广泛意义上”的路由器。假如同时有两个路由器都“最相符”，那么在路由器列表文件中排在最前面的会被选中。

现在，假定这个路由器指定的邮件传输协议是UUCP，同时生成了一个新的目标地址。新地址会传递给传输协议，同时传递的还有下一步负责接收转发邮件的主机名。在前述例子中，smail可能发现要想通过UUCP抵达foo.bar.com，需要使用路径ernie!bert。随后，它会据此生成一个新的目标地址：bert!foo.bar.com@!user。同时，它会指示UUCP传输协议将它作为信封地址使用，再传送给ernie。

若采用的是默认安装，那么可使用下述路由器：

假如目标主机地址可用gethostbyname(3)或gethostbyaddr(3)库调用解析出来，那么消息会通过SMTP传送。唯一的例外是，最终发现该地址引用的是一个本地主机！此时，它也会传送给定向器模块。

smail也能识别以“点分十进制”格式写成的IP地址，并能将其看作一个合法的主机名，只要该IP能通过gethostbyaddr(3)调用解析出来。例如，scrooge@[149.76.12.4]便可能是一个有效的邮件地址，尽管普通人记住的Scrooge的地址在quark.physics.groucho.edu上。

如果你的机器连在Internet上，这些路由器便没有丝毫用处，因为它们不支持MX记录。请参考后文，了解在这种情况下，该如何操作。

假如存在 `/usr/lib/smail/paths` 路径别名数据库，`smail` 就会试着在该文件中搜索目标主机（消除尾随的任何 `.uucp` 字样）。若邮件发给由这个路由器匹配的一个地址，那么它们会通过 UUCP 传送，并利用在该数据库中找到路径。

主机地址（消除任何尾随的 `.uucp` 后缀）会与 `uname` 命令的输出结果对照，核实目标主机是否是一个事实上的 UUCP 邻居。假如答案是肯定的，邮件消息就会用 UUCP 传输协议进行投递。

假如地址不与上述任何一种路由器相符，便会将其投递给智能主机。至于到智能主机的路径，以及打算采用的传输协议，则在 `config` 文件中设置。

这些默认操作适用于大量简单的站点安装，但假如路由情况稍微有些复杂，也很容易出现错误，造成失败。假如你遇到了我们下面描述的任何一种问题，都必须安装自己的路由器文件，令其覆盖默认设定。在附录中，我们给出了一个可首先拿来“开刀”的示范路由器文件。在某些发行版本中，也配套提供了一系列配置文件，它们都经过了仔细的调校，可有效地解决我们碰到的问题。

或许最糟的情况在于，你的主机存在于两个网络区域的交汇处，既有拨号 IP，也有 UUCP 链路。此时，在自己的主机文件中，可能放置了一些只有极少数情况才会通过 SLIP 链接通信的主机名。这样一来，`smail` 就会试图通过 SMTP，投递发给这些主机的任何文件。但是，这往往并非我们所希望的，因为尽管 SLIP 链接并不是经常用到，但 SMTP 的速度要比通过 UUCP 发送邮件慢得多。在采用默认设置的前提下，我们没有办法改变 `smail` 的行为。

要想解决这个问题，可让 `smail` 在查询解析器之前，检查路径文件，并将自己希望强行通过 UUCP 投递邮件的所有主机都放到路径文件中。如果不想通过 SMTP 发送任何邮件，亦可将那些基于解析器的路由器彻底地“剔除”——具体做法是将它们变成“注释”，加上一个注释符号即可。

另一个问题是默认设置并不具备任何真正的 Internet 邮件路由功能，这是由于基于解析器的路由器不会对 MX 记录进行任何处理。要想实现对 Internet 邮件路由的完全支持，要标示出这个路由器并取消使用 BIND 的那个路由器。但是，有的版本中的 `smail` 二进制程序中没有编入 BIND 支持。如果你启用 BIND，但在 `painclog` 文件内却得到这样的消息“`routerinet_hosts:driver bind not found`”时，你就必须获得源码，重新编辑 `smail`。

最后，一般说来，使用 `uname` 驱动程序都不是上上之策。比如，在没有安装 UUCP 时，该驱动程序会产生配置错误，因为无法找到 `uname` 命令。其二就是 UUCP 系统文件内列出的站点多于你事实上与之有邮件链接的站点。它们可能是仅与你交换新闻的站点，或是你偶尔通过匿名 UUCP 从中下载文件的站点，但没有其他通信往来。

为了解决第一个问题，你可用一个外壳脚本来代替执行简单退出的 `uname`。但是最常见的解决办法是编辑路由器文件，并删除这个驱动程序。

路径数据库

`smail` 希望在 `/usr/lib/smail` 下面的路径文件中，找出路径别名数据库。这个文件是可选的，所以如果你从不打算执行任何路径别名路由，只须删除现有的任何路径文件即可。

路径文件必须是一个分类的 ASCII 文件，其中包含一些条目，这些条目把目标站点名映射为 UUCP 路径。这个文件必须分类的原因是 `smail` 利用二元搜索来查找一个站点。该文件内是不允许出现批注的，而且必须用空格符把站点名和路径分开。有关路径别名数据库的详

情，请参见第12章。

如果这个文件是你手动建立的，就应该确保把一个站点的所有合法名称包含在这个文件内。比如，如果一个站点既有一个纯 UUCP 名，又有一个完整形式的域名，你就必须分别为这两个名称增加条目。利用 `sort(1)` 命令把该文件传输一遍，就可对它进行分类。

但是，如果你的站点只是一个叶子站点，就不需要任何路径文件：只须在你的 `config` 文件内设置智能主机属性，把所有的路由交给你的邮件发送机处理。

13.8 将消息投递至本地地址

更为常见的情况是，本地地址只是一个用户的登录名，此时的邮件被投递到他 / 她的信箱——`/var/spool/mail/user`。其他情况还包括别名和邮件列表，以及用户转发的邮件。出现这些情况时，本地地址就扩展成一个新的地址列表，既可能有本地地址，又可能有远程地址。

除了“普通”地址，`smail` 还可以处理其他类型的本地消息，比如文件名和管道命令等。这些不是地址，所以你不能向其发送邮件，比如 `/etc/passwd@vbrew.com`；只有当它们取自转发或别名文件时，才是有效的。

文件名以“/”或“~”开头。后者指代用户的根目录，而且可能是唯一的文件（如果这个文件名取自 `.forward` 文件或信箱中的一个转发条目的话），详情参见后续的讨论。在向一个文件投递消息时，`smail` 会在必要时，把消息添加到这个文件内。

任何以管道符“|”开头的命令都属于“管道命令”。`smail` 碰到这种命令后，会将它传递给外壳程序，同时还会传递它的一些相关参数，只是会将前置的“|”删去。注意消息（邮件）本身会以标准输入的形式，传给命令。

举个例子来说，要想将一个邮件列表送入本地新闻组，那么可使用一个名为 `gateit` 的外壳脚本。同时，设置一个本地别名，利用“`|gateit`”，将来自该邮件列表的所有消息都传给脚本。

假如调用中包含了空白（white space），那么必须将其封闭在一对双引号中。由于这里牵涉到安全性问题，所以千万注意假如是通过某种可疑的方式例如，假如从中取得地址的那个别名文件具有“人人可写”的属性！取得的地址，千万不要执行命令。

13.8.1 本地用户

对本地地址来说，最常见的一种情况便是直接引用某个用户的邮箱。该邮箱保存在 `/var/spool/mail` 中，而且会以用户的名字来命名。注意“邮箱”是该用户拥有的，其中保存有一组邮件，而且采用 660 模式。假如发现该邮箱不存在，`smail` 会自动创建一个。

要注意的是，尽管 `/var/spool/mail` 目前是用来保存邮箱文件的标准场所，但某些邮件软件可能会采用不同的路径，比如像 `/usr/spool/mail` 这样的形式。向本机用户发信时，假如常出现失败的情况，那么便应试着建立一个到 `/var/spool/mail` 的符号链接，看看能否解决问题。

`smail` 要想正常工作，有两个地址（帐号）必须存在：`MAILER-DAEMON` 和 `postmaster`。为一封无法投递的邮件生成反馈消息时，会向 `postmaster`（邮局管理员）帐号发送该邮件的一份拷贝，以检查是否由于配置不当而造成了问题。至于 `MAILER-DAEMON`，则用作反馈消息的“发件人”地址。

假如这些地址没有在你的系统上建立有效的帐号，`smail` 会将 `MAILER-DAEMON` 自动映射到 `postmaster`，而将 `postmaster` 映射到 `root`。显然，无论如何都不应采用这种“默认”的映射

关系，而应为postmaster帐号选择一个恰当的别名，将其映射到负责管理和维护邮件软件的人（帐号）身上。

13.8.2 转发

用户可指示将自己的邮件转发到另一个地址。此时，可选择由 smail提供的两种方法之一。一个办法是在邮箱文件的第一个行中，加入下述字句：

```
Forward to receipient...
```

这样一来，收到的所有邮件都会自动转发给指定的一系列收件人。另外，用户亦可在自己的home目录中，创建一个.forward文件。在该文件中，包含了用逗号分隔的一系列收件人。进行大批量转发时，文件的每一行都会读入，并加以正确的解析。

注意可使用任意类型的地址。也就是说，完全可以像下面这样设置一个 .forward文件：

```
janet, "|vacation"
```

第一个地址会将收到的邮件投递到 Janet的邮箱，而 vacation命令会向收件人返还一条简短的通知。

13.8.3 别名文件

smail能正确控制大量别名文件，它们与 Berkeley的sendmail已知的那些文件兼容。

别名文件的条目有下列的形式

```
alias: recipients
```

recipients（收件人）是一个地址列表，各地址间用逗号隔开，这个列表将被别名代替。如果下一行以空格符开头的话，收件人列表可能会持续到若干行。

有一个特殊的特性允许 smail 处理取自别名文件的邮件列表：如果你指定“:include:filename”为收件人，smail就会读取指定的文件，并将其内容作为收件列表提交。

主要的别名文件是 /usr/lib/aliases。如果你让这个文件成为全球可写文件，smail将不向该文件内指定的外壳命令投递任何消息。下面是一个示例文件：

```
# vbrew.com /usr/lib/aliases file
hostmaster: janet
postmaster: janet
usenet: phil
# The development mailing list.
development: joe, sue, mark, biff
               /var/mail/log/development
owner-development: joe
# Announcements of general interest are mailed to all
# of the staff
announce: :include: /usr/lib/smail/staff,
               /var/mail/log/announce
owner-announce: root
# gate the foobar mailing list to a local newsgroup
ppp-list: "|/usr/local/lib/gateit local.lists.ppp"
```

在向别名文件产生的地址投递消息期间，如果出现了错误，smail就会试图向“别名拥有者”（alias owner）发送该错误消息的副本。比如，如果在把消息投递到开发邮件列表时，投

递屡次失败，错误消息副本就会被邮寄到发件人以及邮局负责人和开发邮件列表拥有者手中。如果拥有者地址不存在，就不会再出现其他的错误消息了。

在把消息投递到文件或调用别名文件内指定的程序时，smail将成为nobody用户，从而避免出现任何安全隐患。特别是在投递到文件时，这种情况会显得非常讨厌。比如在前面给出的文件中，日志文件必须由nobody用户拥有，而且“可写”。否则的话，假若投递给他们，就会失败。

13.8.4 邮件列表

此外，邮件列表也可放弃别名文件，而换用 /usr/lib/smail/lists目录中的文件来进行管理。文件lists/nag-bugs文件描述了一个名为nag-bugs的邮件列表。在这个列表中，包含了所有列表成员的地址，相互间用逗号分隔。该列表亦可以多行形式出现，用一个斜杠符号注明所有的注释性内容。

针对每个邮件列表，都应存在一个名为owner-listame的用户（或别名）。对地址进行解析时，出现的任何错误都会报告给该用户。该地址也作为所有外发邮件的“发件人”地址使用，保存在Sender:这个邮件头字段中。

13.9 以UUCP为基础的传输

目前，有许多传输都被编入smail，其目的是为了利用UUCP套件。在一个UUCP环境内，消息的传递过程是：在下一台主机上调用rmail，再在其命令行上提供作为标准输入的消息和信封地址。在你自己的主机上，rmail应该是指向smail命令的一个链接。

在把消息传递给UUCP传输时，smail把目标地址转换为一个UUCP bang路径。例如，user@host将被转换为host!user。含有“%”地址操作符的地址则会被保留。这样一来，user%host@gateway就会变成%gateway!user%host。但是，smail本身永远都不会生成这样的地址。

此外，smail可通过UUCP收发BSMTP批量邮件。通过BSMTP，在同一“批”邮件中，会封装一条或多条消息（邮件）。而在这个批量邮包中，包含了本地邮件发送程序在建立了一个真正的SMTP连接后要执行的命令。BSMTP常用在大量邮件的“存储与转发”（以UUCP为基础）服务中以节省磁盘空间。在附录中，我们提供的示范传输文件包含了一个示范的bsmtp，它可在一个队列目录中生成部分BSMTP批量邮包。注意稍后必须将它们编译成最终的邮包，这是用一个外壳脚本来完成的，它会自动添加恰当的HELO和QUIT命令。

针对特定的UUCP链路，要想启用BSMTP传输方式，必须使用所谓的“方法文件”（可参考smail(5)手册页了解详情）。假如只有一条UUCP链路，而且正在使用智能主机路由器，那么要想通过SMTP发送批量邮包，可将smart_transport配置变量设为bsmtp，而不是uux。

要想通过UUCP来接收SMTP批量邮包，必须保证目标远程站点可执行一个拆除封装的命令。假如远程站点使用的也是smail，那么必须让rsmtmp建立与smail的一个链接。假如远程站点运行的是sendmail，则还要安装一个外壳脚本，名为/usr/bin/bsmtp，令其执行一次简单的“exec rsmtmp”操作（注意一个符号链接在此是不管用的）。

13.10 以SMTP为基础的传输

目前，smail支持SMTP驱动程序通过TCP连接投递邮件（编者把这个支持称为“简单”。

在将来的smail版本中，他们将提供完整的后端支持，使之能更有效地对这种传输进行处理）。它能够在一台单独的主机上，把邮件消息投递到任何地址，主机名要么被指定为完整形式的域名（能够被连网软件解析），要么被指定为用方括号封闭起来的点分四段式。一般说来，通过BIND、gethostbyname(3)或gethostbyaddr(3)路由器驱动程序解析出来的地址都将被投递给SMTP传输。

SMTP驱动程序将立即试着通过smtp端口（/etc/services中列出的）连接到远程主机。如果不能连接到远程主机或连接超时，邮件投递将在稍后重试。

对因特网上的邮件投递来说，它们要求到目标主机的路由须按照 route-addr格式进行指定（详见第12章），而不是被指定为一个 bang路径（但是，因特网上的路由使用也有限制。所以应该采用完整形式的域名）。由此，smail将把 /usr%host@gateway（gateway是通过 host1!host2!host3!抵达的）转换为 source-addr地址 <@host2,@host3:user%host@gateway>，后者将被当作消息的信封地址发送到 host1。为了启用这类变形（以及内置的 BIND驱动程序），你必须对 transports文件内的 smtp驱动程序条目进行编辑。附录将给出一个 transports文件示例。

13.11 主机名的限制

有时，人们总想在发件人或收件人的地址内采用不完整的主机名（换言之，不想用域名），比如，两个网络进行网关沟通时，其中一个需要完整形式的域名。在 Internet-UUCP中转机上，不完整的主机名应该默认对应于 uucp域。除此以外的其他地址修改都会有问题。

/usr/lib/smail/qualify文件告诉smail哪些域名对应于哪些主机名。qualify文件的组成情况是：第一列是主机名，然后是域名。如果行内包含作为其第一个非空字符的“*”号，就表明这一行是批注。条目的搜索是按其出现的顺序进行的。

有“*”的特殊主机名可对应于任何一个主机名，因此，你可以把以前从未提及的所有主机映射于一个默认的域。此类主机名只能作为最后一个条目出现。

在Virtual Brewery，所有主机都已经被设置为“在收件人地址内，采用完整形式的域名”。不完整收件人地址被看作在 uucp域内，所以，qualify文件内，只需要有一个条目。

```
# /usr/lib/smail/qualify, last changed Feb 12, 1994 by janet
#
* uucp
```