

第19章 其他问题

19.1 PLIP的空打印机电缆

为了将空打印机电缆用于 PLIP 连接，需要两个 25 针的连接器（也就是 DB-25）和一根 11 导体电缆。这根电缆至少要有 15 米长。

查看连接器时，应该能看清楚每个针脚上的编号，从位于最左上部的针对应的 1 到最右下部的针对应的 25。对空打印机电缆来说，必须将它和连接器上的针一一对应，具体对应如下：

D0	2	-	15	ERROR
D1	3	-	13	SLCT
D2	4	-	12	PAPOUT
D3	5	-	10	ACK
D4	6	-	11	BUSY
GROUND	25	-	25	GROUND
ERROR	15	-	2	D0
SLCT	13	-	3	D1
PAPOUT	12	-	4	D2
ACK	10	-	5	D3
BUSY	11	-	6	D4

其余的针脚保持未连接状态。如果该电缆被塑料外壳包起来了，这个塑料外壳就应该只和 DB-25 规格的金属外壳的一端连接起来。

19.2 示范 smail 配置文件

本节向大家展示一个示例配置文件，它用于局域网上的一个 UUCP 叶子站点。这些配置文件以 smail-3.1.28 源代码中包含的示例文件为基础。虽然我打算为大家讲讲这些文件的工作原理，但仍然建议大家参考 smail 手册，其中详细讨论了这些配置文件的每一个细节。虽然你已经知道了 smail 配置文件的基本思路，但 smail 手册还是值得一读的，它相当简单，非常出色！

下面展示的第一个文件是 routers 文件，其中描述了 smail 的路由器集。当 smail 必须把邮件消息投递到指定的地址时，它会依次把这个地址交给所有的路由器，直到其中之一的地址与这个指定的地址相符。这里的相符指的是路由器在其数据库内找到了目标主机，其依据是 paths 文件、/etc/hosts 或该路由器与之打交道的任何路由机制。

smail 配置文件内的条目始终以一个唯一性的名字开头，该名字用以标识路由器、传输或执行者。条目之后为定义其行为的属性列表。该列表由一个全局属性集组成，比如采用的驱动程序和只有特定驱动程序才能识别的私用属性等。各属性之间用逗号隔开，而全局属性集和私用属性集之间分别用分号隔开。

为了更好地进行区别，我们以此为例：假设你打算维护两个各自独立的 pathalias 文件；其中之一包含你所在域的路由信息，另一个文件内包含全局路由信息，可能是 UUCP 映射产生的。有了 smail 时，你可在 routers 文件内指定两个路由器，两者都采用 pathalias 驱动程序。该驱动程序在 pathalias 数据库内查找主机名。它希望你能在一个私用属性内为它指定文件名。

前面的两个路由条目中，各自的第二个全局属性都定义了一旦找到与地址相符的路由器，应该采用的传输方式。我们的例子中，邮件消息将采用 uux 传输形式来投递。传输形式是在 transports 文件内定义的，稍后我们将就此进行讨论。

要想具体规定一条消息由谁投递，可指定一个方法文件，而不是指定传输属性。方法文件可在目标主机名与传输之间建立一个对应关系。但这方面的详情，这里不打算深入讨论。

在混合了 UUCP 和 TCP/IP 的一个环境中，我们可能会碰到这样一个问题：对 /etc/hosts 文件中指定的主机而言，可能只有极少数的情况，才会有人要求建立与它的 SLIP 或 PPP 连接。通常，我们仍然想通过 UUCP 发送给它们的任何邮件。为防止 inet_hosts 在与这些主机相符的情况下投递，我们必须将它们置于 paths/force 文件中。事实上，这又是一种“路径别名”（Pathalias）样式的数据库，它会在 smail 查询解析者之前，进行咨询。

对本地地址相关的邮件来说，对它们的控制是在 directors 文件中进行的。它只由路由器文件构成，同时有一个条目列表，定义了每一个定向器。注意定向器本身并不负责一条消息的投递，它们只是执行所有可能的邮件重定向工作，比如通过别名重定向，通过邮件转发重定向……等等。

将邮件投递给一个本地地址比如 janet 时，smail 会将用户名依次传递给所有定向器。假如找到一个相符的定向器，它要么指定一个具体负责邮件投递的传送人（比如用户的邮箱文件），要么马上生成一个新地址（比如在匹配出一个别名之后）。

由于这里牵涉到的安全性问题，定向器通常要执行大量检查，判断它们使用的文件是否有害。凡是用可疑方式得到的地址（比如通过一个全局可写的别名文件），它们都会被作上标记，指出“不安全”。有些传送者干脆将此类地址屏蔽，比如将一条消息发给一个文件的传送者。

除此以外，smail 也会将用户与每个地址对应起来。所有读写操作都会以用户的身份执行。例如，假定将一条消息投递到 Janet 的邮箱，那么地址理所当然需要与 janet 关联到一起。而对其他地址来说，比如自别名文件获得的那些地址，则让其他用户同它们关联，比如一个 nobody 用户。

欲了解这些特性的详情，可参考 smail 手册。

注意 smail 手册放在 Linux 文档项目联机手册的第 8 部分。

成功路由或定向了一条消息后，smail 会将消息传递给由与地址相符的路由器或定向器指定的传送者。这些传送者定义在 transports 文件中。同样地，一名传送者是由一系列全局和私用选项来定义的。

对每个条目定义的选项来说，其中最重要的便是具体负责传输的一个驱动器（driver），比如管道驱动（pipe driver），它的任务是调用 cmd 属性中指定的命令。除此以外，传送者还可使用大量全局属性，它们可对消息头进行方方面面的转换，同时还可对消息的主体进行处理。例如，return_path 属性可令传送者在消息头（邮件头）中插入一个返回路径字段。unix_from_hack 属性则令其位于以一个 > 符号开头的、每一个可能的“From”字样之前。在清单 19-1 中，我们列出了一个示范性的 /usr/lib/smail/transports 文件。

清单 19-1 一个示范性的 /usr/lib/smail/transports 文件

```
# A sample /usr/lib/smail/transports file

# local - deliver mail to local users
local: driver=appendfile.      # append message to a file
```

```

return_path,          # include a Return-Path: field
from,                 # supply a From_ envelope line
unix_from_hack,      # insert > before From in body
local;               # use local forms for delivery

file=/var/spool/mail/${lc:user}, # location of mailbox files
group=mail,          # group to own file for System V
mode=0660,           # group mail can access
suffix="\n",         # append an extra newline

# pipe - deliver mail to shell commands
pipe: driver=pipe,    # pipe message to another program
return_path,         # include a Return-Path: field
from,                # supply a From_ envelope line
unix_from_hack,     # insert > before From in body
local;              # use local forms for delivery

cmd="/bin/sh -c $user", # send address to the Bourne Shell
parent_env,            # environment info from parent addr
pipe_as_user,          # use user-id associated with address
ignore_status,        # ignore a non-zero exit status
ignore_write_errors,  # ignore write errors, i.e., broken pipe
umask=0022,            # umask for child process
-log_output,           # do not log stdout/stderr

# file - deliver mail to files
file: driver=appendfile,
return_path,          # include a Return-Path: field
from,                 # supply a From_ envelope line
unix_from_hack,      # insert > before From in body
local;               # use local forms for delivery
file=$user,           # file is taken from address
append_as_user,       # use user-id associated with address
expand_user,          # expand ~ and $ within address
suffix="\n",         # append an extra newline
mode=0600,            # set permissions to 600

# uux - deliver to the rmail program on a remote UUCP site
uux: driver=pipe,
uucp,                 # use UUCP-style addressing forms
from,                 # supply a From_ envelope line
max_addrs=5,          # at most 5 addresses per invocation
max_chars=200;        # at most 200 chars of addresses

cmd="/usr/bin/uux - -r -a$sender -g$grade $host!rmail (($user)$)",
pipe_as_sender,       # have uucp logs contain caller
log_output,           # save error output for bounce messages
# defer_child_errors, # retry if uux returns an error

# demand - deliver to a remote rmail program, polling immediately
demand: driver=pipe,
uucp,                 # use UUCP-style addressing forms
from,                 # supply a From_ envelope line
max_addrs=5,          # at most 5 addresses per invocation
max_chars=200;        # at most 200 chars of addresses

cmd="/usr/bin/uux - -a$sender -g$grade $host!rmail (($user)$)",

```

```
pipe_as_sender,      # have uucp logs contain caller
log_output,         # save error output for bounce messages
# defer_child_errors, # retry if uux returns an error

# hbsmtp - half-baked BSMTMP. The output files must
# be processed regularly and sent out via UUCP.
hbsmtp: driver=appendfile,
inet,                # use RFC 822-addressing
hbsmtp,              # batched SMTP w/o HELO and QUIT
-max_addrs, -max_chars; # no limit on number of addresses

file="/var/spool/maill/hbsmtp/$host",
user=root,           # file is owned by root
mode=0600,           # only read-/writable by root.

# smtp - deliver using SMTP over TCP/IP
smtp: driver=tcpSMTP,
inet,
-max_addrs, -max_chars; # no limit on number of addresses
short_timeout=5m,      # timeout for short operations
long_timeout=2h,      # timeout for longer SMTP operations
service=smtp,         # connect to this service port
# For internet use: uncomment the below 4 lines
# use_bind,           # resolve MX and multiple A records
# defnames,          # use standard domain searching
# defer_no_connect,  # try again if the nameserver is down
# -local_mx_okay,    # fail an MX to the local host
```
