ADMINISTRATION GUIDE

InterMail® Post.Office™ EDITION

Version 3.5

Software.com™

THE INTERNET INFRASTRUCTURE COMPANY™

version 3.5 980429A

Table of Contents

Preface

Welcome to Post.Office!

The *Post.Office Administration Guide* is the primary Post.Office manual. It includes instructions for performing almost all mail server operations and provides in depth coverage of system architecture and mail flow through your server. This volume is supplemented by the complete suite of Post.Office manuals, including an *Installation Guide*, a *List Owner's Guide*, and a *User's Guide*. Together they provide the reference material and instruction sets required for a complete understanding of the Post.Office mail server software.

Structure of the Manual

This manual is organized by function. Operations are presented in order of probable use, but feel free to review the information in whatever manner you desire - even skip sections if the content is familiar.

- Chapter 1 provides an overview of e-mail.
- Chapter 2 describes the role Post.Office plays in the e-mail universe.
- Chapter 3 describes the web interface to Post.Office, your primary point of contact.
- Chapter 4 covers system configuration.
- Chapter 5 discusses account management.
- Chapter 6 explains the Program Delivery feature in detail.
- Chapter 7 deals with mailing list management.
- Chapter 8 describes recommended system monitoring.
- Chapter 9 provides instructions for backing up and restoring the mail system.
- Chapter 10 covers troubleshooting.
- Chapter 11 explains the Post.Office command line utilities.

Happy reading!

Style and Conventions

Consistency is the key. In order to make this manual as easy to use as Post.Office, we've adopted the following conventions:

Icons

Occasionally, an icon will appear in the left margin. Each icon has a specific meaning. The paragraphs that follow identify the icons and their intended use.

Note: *Notes alert you to information of special interest or provide clarification on the use of a particular Post.Office feature. Notes supplement standard content and are not required reading.*

Warning! Warnings contain critical information. Typical warnings include cautions about maintaining system security and avoiding overburdening your mail server. Failure to read a warning may have serious consequences.

Hint: *As you may have guessed, the helpful hints suggest ways to make your life easier. The tips are based on suggestions from other Post.Office users, including the Software.com "Postmasters."*

Security Feature: The security features of Post.Office (and there are many) are highlighted by the appearance of a lock. Look for the locks when reviewing the security aspects of your mail server installation.

UNIX: Certain comments and instructions apply to UNIX users only. The UNIX computer icon provides a simple means of recognizing such items. Post.Office users whose system is installed on Windows NT should ignore these discussions.

Windows NT: Other comments and instructions apply to Windows NT users only. The NT icon marks the discussion of such items. Any comments associated with the NT icon can be safely ignored by UNIX users.

Terminology and Type

- Fields and forms are referenced by their proper names.
- Literal entries (commands and such) appear in `monospaced` type.
- **Links** are underlined and in boldface.
- Important new terms appear in *italics*.
- Variable names appear in *`monospaced italics`*.
- Optional entries appear in [square brackets].

Standard Examples

Generic Term	Standard Examples	Meaning in this Manual
domain	software.com	a partial domain name (host name excluded)
host.domain	sparky.software.com	a fully qualified domain name (with host name included[1])
user@domain	john.doe@software.com	a sample user's e-mail address
list@domain list@host.domain	biking@software.com biking@sparky.software.com	a mailing list address; the address to which messages are submitted for posting
list-request@domain	biking-request@software.com	a mailing list request address; the address to which commands and requests for subscription or unsubscription are sent.
owner-list@domain	owner-biking@software.com	the list owner alias address; the address used to correspond with the mailing list owner

Questions and Comments

Copies of this manual can be obtained by anonymous FTP to `ftp.software.com` or from our web site at `http://www.software.com`. If you can't find an answer to your question in the manual, check the list of Frequently Asked Questions (FAQ), also located on our web site at `http://www.software.com`.

To suggest improvements or provide feedback on the content of this manual, send e-mail to `Post.Office.Manual@Software.com`

Legal Notices

1 Host names often involve a theme such as colors, animals, or cities. We've used common pet names as the theme for our sample hosts.

Trademarks

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this documentation, and Software.com was aware of a trademark claim, the designations have been printed in initial caps or all caps.

Post.Office and Software.com are trademarks of Software.com, Inc.

Licensing Agreement

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL SOFTWARE.COM BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MD5 Message-Digest algorithm

The MD5 Message-Digest algorithm used in Post.Office is ©1991-92 RSA Data Security, Inc. Created 1991. All rights reserved.

License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software.

The Regular Expression Routines

The Regular Expression Routines used in Post.Office are © 1992-94 Henry Spencer. All rights reserved. This software is not subject to any license of the American Telephone and Telegraph Company or of the Regents of the University of California.

Permission is granted to anyone to use this software for any purpose on any computer system, and to alter it and redistribute it, subject to the following restrictions:

1. The author is not responsible for the consequences of use of this software, no matter how awful, even if they arise from flaws in it.

2. The origin of this software must not be misrepresented, either by explicit claim or by omission. Since few users ever read sources, credits must appear in the documentation.
3. Altered versions must be plainly marked as such, and must not be misrepresented as being the original software. Since few users ever read sources, credits must appear in the documentation.
4. This notice may not be removed or altered.

The Regents of the University of California Copyright

Post.Office includes software that is © 1990, 1993, 1994. The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by Mike Olson.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Re-distributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Re-distributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: This product includes software developed by the University of California, Berkeley and its contributors. 4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ‘‘AS IS’’ AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1

E-mail in a Nutshell

This chapter presents a non-technical overview of the basic concepts and common conventions of electronic mail (e-mail), including:

- Elements of a message
- MIME
- Mail clients
- Mail servers
- Addressing protocols
- The Postmaster
- Directory services
- Junk mail and other e-mail abuses

This chapter is not about the Post.Office mail system; rather, it is a brief introduction to some of the basic concepts that apply to most e-mail systems. It will be useful if you are new to the world of e-mail administration.

A more specific discussion of Post.Office follows in the next chapter, so if you already know what e-mail is and what kinds of programs comprise a mail system you may want to skip ahead.

1.1 It Starts as a Message

E-mail is all about *messages*; that is, e-mail provides a way for two or more people to exchange messages. Just as the postal service is used to send postcards, letters, and magazines, e-mail is used to send various kinds of messages. An electronic message can range from a simple memo or letter to a complex multimedia presentation designed to overload and delight your senses. Regardless of its content, the message is the fundamental currency of electronic mail.

1.1.1 Evolution of the Electronic Message: E-mail is Born

The most rudimentary method of leaving a message for someone who uses a computer is to tape a hand-written note on their monitor. The next step, electronic messaging in its most basic form, occurs when you type a few words in an open window on the computer screen hoping the next person who comes along will find it. This basic electronic

message system works if nobody else needs to use that particular computer, and if you don't mind leaving the computer and monitor on.

However, if more than two people are using this computer, then the electronic message must be stored (as a file on a disk) until the recipient comes along. Only when the message is safely put in a file can the computer be used for other purposes, employed by other users,[2] or shut off. As long as the two users who wish to communicate agree upon a common file where they will store messages for each other, this system works.

However, using a large, single file is clumsy. Instead, users may agree upon a directory in which to store messages. Each message could then be stored as an individual file with a descriptive file name. Even so, a large volume of messages between several users can fill up a directory awfully fast, and it can become difficult to make heads or tails out of the resultant mess.

When confronted with a large number of message files, it would be nice to know several things about the message without opening the file, such as: who a message is for, what it's about, and when it was sent. If more than two users share this system, it's also useful to know who a message is from before reading it.

Historically, postal mail solved this problem of message organization with various conventions of encapsulation. Letters almost invariably began with an indication of who they were for. Often other information, such as where the letter was written and what the letter was about, preceded the main body of the message. It's the same in e-mail. Messages are composed of headers and bodies. Headers include information such as sender, recipient, and subject, which allows recipients – and the programs that serve them – to sort and prioritize their mail before taking the time to examine the body of the message.

In addition to facilitating the handling of mail upon receipt, encapsulation allows mail to be delivered more efficiently. In postal mail, messages are placed in envelopes which contain only the information required for delivery (as well as a return address in the event mail cannot be delivered). Likewise, e-mail programs use electronic envelopes, which are marked with a destination and return address and "contain" the electronic message (header plus body). Basically, the principles behind e-mail are as simple as the postal mail that you've been using all along – relax, it's not all that complicated.

1.1.2 Electronic Envelopes

When delivering a message from one user to another, however, an e-mail program only needs to know two things:

- where and who the message is going to
- who it was from, in case it needs to be returned

---

2 *User* is a generic term for anybody who uses a computer.

This information is used to create an *envelope*, which is directly analogous to a postal envelope: both are labeled with "to" and "from" addresses and contain the message (headers and body) within.

Only programs use envelopes – all that users ever get to see are the headers and body of a message. Still, it's good to know that envelopes exist in case we ever need to really pin down an ontological definition of e-mail.

1.1.3 Message Headers

Delivery is only the first step in the process. There is still a need for easy organization or messages and the recognition of message content.

Corporations and other institutions have long used messages (often called memos) that have key pieces of information laid out in a series of headers at the beginning of the message (see Figure 1-1). Header information allows institutional mail services to deliver memos efficiently and gives memo recipients an initial idea of what the message is about before delving into the full content.

```
To:       Jane D.
From:     John S.
Subject:  Toga contract termination
Date:     July 27, 1994
-----------------------------------------------------
Jane,

I have decided to terminate our contract with the Toga
company. The togas don't seem to convey the corporate image
which we require.

Let's meet at 3:30 to discuss the details. OK?

John
```

Figure 1-1 Message headers provide key information about who a message is for and who it is from, as well as what it is about and when it was drafted or sent.

Headers can be just as effective in managing a gaggle of electronic messages in a directory. One can make sense of a message file only by opening it and checking the header information to see who a message is for, what it's about, who sent it, and when. This is tedious work. Fortunately, the dull, tedious, and repetitive work of opening a large number of message files and examining their headers is exactly the kind of thing that computers are good at.

As long as headers are consistently formatted, an e-mail program can easily scan through a pile of messages and find all the messages that begin with, for example, the line "To: Jane." Similarly, if other header information indicates when messages were written, a computer can organize these messages and present them to Jane in chronological order.

The key to headers, as far as computers are concerned, is that they be absolutely consistent. E-mail interoperability depends on an agreement (or standard protocol, as the people working on such things like to call agreements) for the formatting of the headers.

While different e-mail systems do things differently, all have some kind of header information, and for two systems to be interoperable, any differences must be eliminated or somehow resolved.[3]

It is the tight regulation of the use and format of headers which allows users with disparate e-mail programs to send each other electronic mail. At the same time, headers provide users with valuable information about their e-mail.

1.1.4 The Body of a Message

Just as the body of a letter tends to make up the bulk of traditional postal mail, in general the body makes up the bulk of an e-mail message. While users must cater to the needs of computers and programs when writing headers, there are no such restrictions on the body of a message. As a result, the body of an e-mail message tends to look a lot like the body of a postal letter.

Although often limited to the rather rudimentary ASCII character set,[4] newer and more sophisticated electronic mail programs are increasingly allowing users to send each other elaborately formatted text and even graphics, sound, and video clips (multimedia). The more complex the message medium, the larger the amount of data that must be transferred when a message is sent from one user to another. In some systems, sending a large, complex file can create a bottleneck – a sort of traffic jam on the information superhighway. As increasing bandwidth[5] allows larger data streams on networks, e-mail users will be able to make more and more use of data-intensive e-mail features.

MIME: Multipurpose Internet Mail Extension

Back in the days when people were still relieved about not having to use punch cards to talk to computers any more, sending any kind of text message was considered pretty cool. E-mail evolved without any allowances for video and audio files, or even rich text, the highly formatted text with **bold,** *italics,* and all the other spiffy stuff we've gotten used to since the word-processor consigned the typewriter to the antique shop.

In order to incorporate formatted text and multimedia into the Simple Mail Transfer Protocol (SMTP), which is currently the most common existing e-mail protocol, a new protocol called MIME (Multipurpose Internet Mail Extension) was developed. MIME

---

3 Programs which are able to transfer messages between systems using different protocols are called gateways or switches. Generally, transferring messages from one system to another through a gateway is a tremendous hassle, since the header information which allows a message to be delivered on one system may not include the necessary information to deliver a message in another system.

4 ASCII offers all the letters and numbers as well as standard punctuation symbols, but not such features as bold, underline or different fonts.

5 Bandwidth is a way of measuring how much data can pass through a given network in a given period of time. Multimedia features such as graphics, sound, and video require more bandwidth than certain systems can provide. Like everything in the computer world, bandwidth capacity is increasing rapidly.

allows you to incorporate anything from a recording of your newborn's voice to a short movie in an e-mail message.

As long as both parties have MIME-enabled e-mail programs (not all e-mail programs support MIME), people can exchange any kind of multimedia file they want by simply appending the file to their message. Since multimedia is still in the early stages, you should check to make sure that someone has MIME capabilities before you send them a pile of stuff.

1.2 E-mail Software: Mail Clients

Mail clients are programs which assist users in carrying out tasks related to electronic mail. These include creating and submitting messages for delivery, checking for new incoming mail, reading received messages, and organizing the volumes of saved messages generated by high usage. The mail client is the only element in the maze of networks and e-mail handling programs with which most users have any contact (see Figure 1-2).

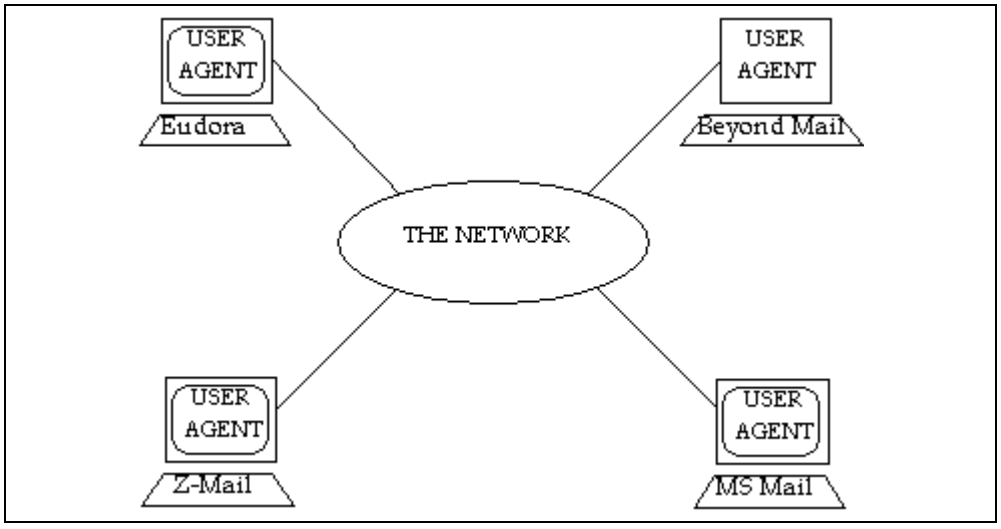

Figure 1-2 A mail client is the only e-mail program with which most users have any contact (network not shown to scale).

If you ask users what e-mail program they use, they will generally tell you the name of their mail client. This is because the mail client takes care of most of the tasks required to send and receive messages. Other tasks are delegated to programs which are hidden from the user.

In its simplest form, a mail client is a program that allows you to create a text message for another person which can be read by that person with the help of a similar program. Mail clients which can send and receive graphics and multimedia work along the same lines as their simpler cousins. While all mail clients can handle plain text messages, the exchange of complex multimedia messages requires that the mail clients follow an agreed upon standard format, such as MIME, for the body of such messages.

1.2.1 Creating Messages

To create a new e-mail message, mail clients frequently provide users with a message template, so that users need only fill in the blanks in order to complete the headers. Mail clients leave the body blank so that users can fill it in as they please. The mail client operates in this manner as a simple word processor, and most offer at least minimal editing functions. Together, these editing features allow users to include any amount of supplementary textual (and often multimedia) information in their messages. Once a message is complete, the mail client will forward the message to a more specialized e-mail program (a mail server) for delivery.

Mail clients typically list incoming messages on a menu or in a window which may be called the “In Box,” or something similar. Often this message list shows who the message is from, when it was sent, and what the subject of the message is. Users select the message they wish to read, and it is displayed for them on the monitor (junk mail you throw away without reading, just like you already do with postal mail, but for once it doesn’t end up in a landfill).

Frequently users will want to save messages that they may need to look at again at a later date. Rather than maintain a single directory filled with a random assortment of messages, many mail clients help users organize their messages into a set of directories or mailboxes, so that messages can be organized by subject or according to whatever taxonomy the user prefers.

1.2.2 Sending and Receiving Messages

In today’s world of huge networks connecting bazillions of users, mail clients are not up to the task of transferring messages to their recipients. This task is relegated to specialized programs called *mail servers* (which are described in Section 1.3).

To send a message, a mail client needs only to give it to a mail server. This generally requires only a single command on the part of the user to whisk the message away across the networks. As long as networks are functioning correctly, a message can be delivered around the world in a matter of seconds.

Mail clients can receive messages in a couple of different ways. One option is to have the mail server place messages directly into a specific directory (on the same computer as the mail server, but in the user’s directory). This directory functions as the user’s mailbox. In this case the mail client consults this directory any time that a user asks it to check for mail. Any messages found in this directory are retrieved and listed for the user. This type of delivery is often used on UNIX machines.

A second and increasingly popular method of message delivery is directly supervised by the mail server. Rather than place incoming messages in an externally-controlled mailbox, the mail server itself manages the mailboxes, holding onto messages until the user checks for mail. When the user does this (remember, the user probably doesn’t even know that he has a mail server secretly working on his behalf), he uses the check mail command on his mail client, and the mail client checks with the mail server. If the mail

server has any messages for that user, they are made available to the mail client, which in turn makes them available to the user. This second method employs the Post Office Protocol (POP).

The advantage of the POP delivery method is that it does not require that the mail server have access to the user's mailbox directory (which means that the user need not have a system account on the server). This is advantageous in today's networked environment where more and more often the mail client and the mail server are located on different computers. In this case, the mail server is at the disposal of the mail client, which contacts it when it wants to send out a message or check for mail (see Figure 1-3).

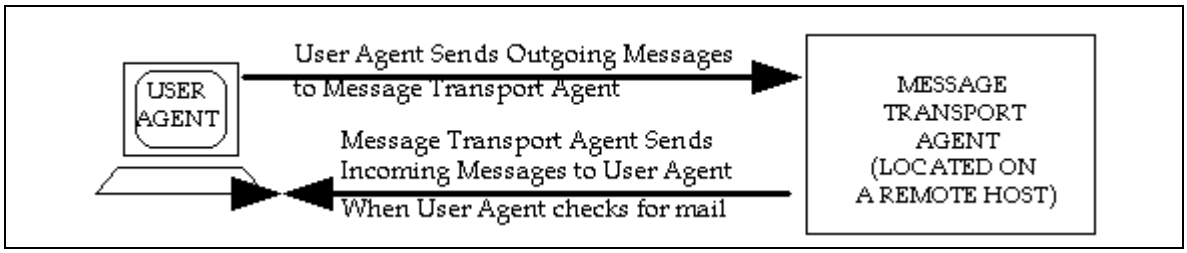

Figure 1-3 Often, when the mail client and the mail server are located on two different computers, the mail server does not deliver messages until the mail client checks for mail.

Another advantage of POP delivery is that it does not require the user to have a system account on the same computer where the mail server is running.

1.3 E-mail Software: Mail Servers

A *mail server* is a specialized program designed to deliver messages across today's increasingly large networks. Mail servers generally interact with other programs – primarily mail clients and other mail servers – rather than with users. The most common task which mail servers carry out is to accept a message from one mail client and deliver it to another.

Post.Office is a mail server.

1.3.1 The Role of Mail Servers in an E-mail System

Mail servers do most of the work in the e-mail universe, including the sorting, forwarding, storing, and delivering of mail. The function of a mail server is analogous to the postal service; late at night, in post offices around the globe, thousands of insomniacs sort through mountains of bills, catalogs, and coupon mailers, so that in the morning postal carriers can deliver these missives to our doors. If we think of a mail client as a personal secretary who helps us write our messages, we can liken mail server to the thousands of letter-sorters and others who work behind the scenes to ensure that we get our mail.

A mail server is also a database which stores information about your e-mail account – not the least of which is your e-mail address. All information regarding your account is stored in your mail server, including your password, instructions for how your mail should be delivered, and some other items that you probably never knew existed. Having an e-mail account really just means having a mail server account, so when your system administrator mentions “setting up an e-mail account,” what they’re really talking about is adding a new user account to the mail server database.

Mail servers are daemon programs, which means that they are running 24 hours a day, ready and anxious to serve. When a mail client (or another mail server) wants to give a message to a mail server, it contacts it and gives it the message. In contrast, mail clients are usually only active when a user is interested in writing, sending, receiving, or perusing e-mail.

When messages need to travel between mail clients, it is commonly one or more mail servers that carry out the task. Figure 1-4 illustrates how mail servers such as Post.Office and sendmail transport messages between mail clients such as Eudora and Pine:

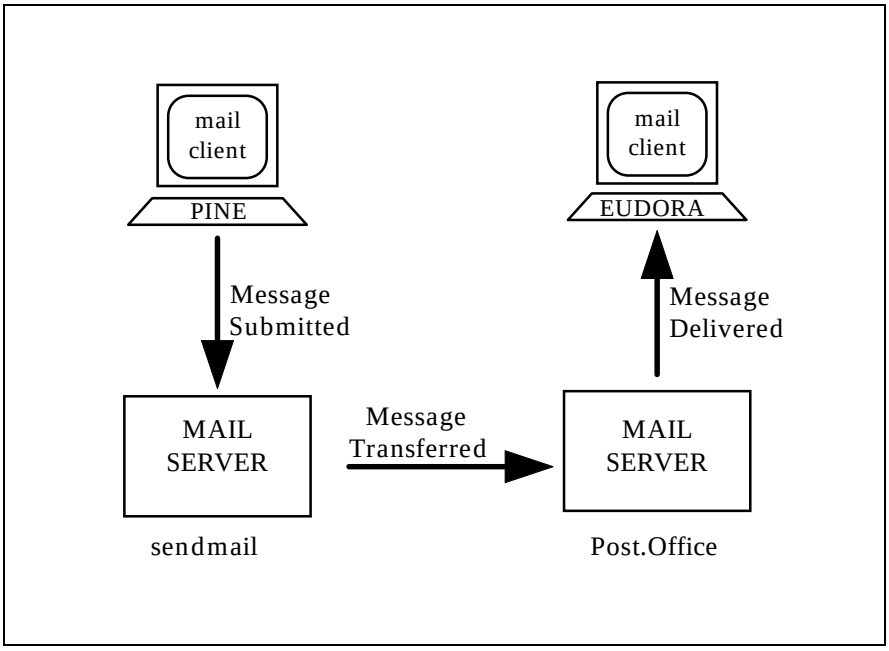

Figure 1-4 In a direct transmission, a message is forwarded between the two mail servers (in this example, Post.Office and sendmail) which are closest to the mail clients (in this example, Pine and Eudora). Message travel is indicated by the arrows.

1.3.2 Sorting and Forwarding

We know that when mail is placed in a postal mail box, its next stop is a post office. There it is sorted and a forwarding decision is made. For local mail this may mean putting the letter in a mail delivery person’s mail pouch, while mail destined for a distant city may require that the process of sorting and forwarding be repeated several times at various post offices in different cities along the way.

This is the same method that is used in e-mail delivery. As an e-mail message travels from one mail client to another via one or more mail servers, decisions are made as to the routing of the message at each step along the way. Using the addressing information provided on the electronic "envelopes," mail servers sort through messages and make decisions about where to forward them. In some cases a message needs to be sorted only once and can then be forwarded directly to its recipient. In other cases this process must be repeated several times along the way.

If a message must travel from a writer at Software.com to his brother who is attending the University of Washington, the writer and his mail client create a message together. The mail client then gives the message to the mail server at Software.com. This mail server then forwards the message to another mail server at the University of Washington, and this second mail server then delivers the message to the writer's brother. Broken down, this process consists of five steps, shown in Figure 1-5:

```
1.  User to client          (sender creates message)
2.  Client to server        (message forwarded)
3.  Server to server        (message forwarded)
4.  Server to client        (message forwarded)
5.  Client to user          (recipient reads message)
```

Figure 1-5 An example of the steps involved in getting e-mail to its destination.

Sometimes a message must be relayed by an intermediate mail server. In such a case it would be handled by three (or more) mail servers: one mail server that accepts the message from a mail client, another that relays it, and the last mail server which delivers it to the recipient's mail client. In such a case, Step 3 of Figure 1-5 would be repeated one or more times:

1.3.3 Forwarding a Message to a Mail Server

All mail servers are daemons waiting for other e-mail programs to contact them and give them a message. Like all e-mail transactions, the conversation is a client/server transaction, a kind of call and response conversation in which one computer asks for something and another provides it. In this case the mail server that is accepting the message is a server, while the client that is transmitting the message could be a mail client or another mail server. The conversation can look something like the illustration shown in Figure 1-6.[6]

6 This is a parody of a Simple Mail Transfer Protocol (SMTP) conversation.


```
Hello this is Computer1 (mail client or mail server)
>>> Hello this is Computer2 (mail server)
I want to send you a message from Bob
>>> OK
It's for Jane
>>> OK
Here's the message, ending with our secret handshake
>>> OK
Data, data, ... data, secret handshake
>>> Message received
Good-bye.
>>> Good-bye.
```

Figure 1-6 Example of what a mail client (or mail server) might say to a mail server when giving it a message.

1.3.4 Delivering A Message to a Mailbox for Client Retrieval

When a mail server places a message in a user's mailbox, it is simply creating and saving a file in the user's directory. The alternative is for the mail server to hold onto a message in a mailbox of its own until a mail client retrieves any accumulated messages from another computer. The mail server will again act as a server while the mail client literally impersonates the recipient it is representing (Figure 1-7):[7]

```
Hello? Anyone there?
>>> Hi, this is your mail server
This is Jane
>>> Jane, your mailbox has 2 new messages
Give me the first one
>>> Data, data, data... secret handshake
Give me the second one
>>> Data, data, data... secret handshake
Thanks, I got them. Good-bye
>>>You're welcome, Good-bye
```

Figure 1-7 What happens when a mail client retrieves two messages from a mail server (more or less).

Besides delivering a message to another mail server or to a mail client, there are a couple of other ways that a mail server can deliver a message: to special programs or to an error handling routine.

In the first case a mail server can be told to deliver all messages addressed in such-and-such a way to a special program. This could be, for example, a mail sorting program or a mailing list exploder. When it receives a message for this type of program, the mail server starts the program and gives it the message. Mail sorting programs are especially popular with people who receive large volumes of e-mail and want, for example, to separate personal messages from mailing list messages.

7 This is a parody of a Post Office Protocol Version 3 (POP3) conversation.

The second case involves the disposition of messages that the mail server cannot decide what to do with. While some mail servers reject such messages outright, others forward them to the Postmaster (the mail system administrator) who must then decide what to do with them. An error handler is a program which assists the Postmaster in sorting through, responding to, and disposing of these "problem" messages.

1.3.5 Mail Servers and Addresses

Often a mail server can function as a local post office for a network or a portion of a network. When used in this manner the mail server has a list of local recipients for whom it receives messages. This list translates, from the mail server's perspective, into a list of addresses which includes everybody in that mail server's "electronic neighborhood" (often a neighborhood like this is called a *domain*). A mail server accepts a message when it recognizes the address as a local address.

Often a single user will receive mail at multiple addresses, all of which the mail server funnels to that user's mailbox. The reason for having more than one address can stem from wearing several hats (so that a user may receive all the e-mail addressed to the Sales Department as well as to her own name) or simply because she wants it to be as easy as possible to get mail to her. E-mail programs are rather neurotic about how addresses are written. One way to compensate for this is to try to guess how people will mess up your address – you can then tell the mail server that mail sent to any of these "guesses" is really meant for you. The specifics of e-mail addresses are illustrated in more detail below.

1.4 Addressing Protocols

Addressing protocols are the key to allowing users and computers to contact each other across networks. This section describes addressing in general as well as providing some details on the most commonly used addressing protocol, the Domain Name System (DNS).

Back when there were only three channels to watch on television and computers were big enough to squash people, addressing systems were simple. You gave every computer a name, and then you compiled a list of those names and information about where each computer was. If you added a computer to your network you would add the name and location of the new computer to the address list maintained on every other computer on the network.

Nowadays there are too many channels on TV, and far, far too many computers on networks for lists like this to be maintained on each computer. Networks like the Internet are growing at exponential rates and there is simply no possible way for all computers to keep constant track of each other. Addressing systems were developed to allow computers to find each other when they needed to, and these addresses are the key to today's electronic mail systems.

There are two common addressing systems (X.400 and DNS) which resolve the difficulty of providing addresses to millions of computers. Because DNS addresses are simpler (see Figure 1-8) and more common, we will describe DNS addressing in this section.

```
A Sample X.400 Address:

     /PN=SMITHJ/O=ORG/PRMD=COMPANY/ADMD=TELCOM/C=US

A Sample DNS Address:

     Jane.Doe@Software.com
```

Figure 1-8 While X.400 addresses tend to be complex, DNS addresses are fairly simple. Both X.400 and DNS addressing systems are capable of supporting millions of hosts.

1.4.1 The Domain Name System (DNS)

With the DNS, each computer actually has two "names": one that is useful for people (a DNS address), and a second that is better suited for computers (an IP address[8]). For example, there is an entry in the DNS for a computer named sparky.software.com. The other name in the DNS that refers to this computer is `[198.17.234.1]`. This string of numbers is the computer's IP address (and is used almost exclusively by other computers).

The address `sparky.software.com` can be used to illustrate the hierarchical nature of the DNS. The right-most word (an abbreviation) indicates the type of organization (or sometimes the country) in which the computer is located. In this case the abbreviation `com` indicates a commercial organization.[9] Next comes the name of the organization which is unique within the `com` domain (for example, there is also a `software.org`, but there can be only one `software.com`). The name given to the computer is sparky.

Rules dictate there be only one organization named software in the com domain, and only one computer named sparky in the software.com domain. These rules ensure that every DNS address is unique.

In this way, a message that is addressed to Jane, who uses the computer sparky, could be addressed:

```
To: Jane@sparky.software.com
```

8 IP stands for Internet Protocol.

9 The other types of organizations are: "gov" for government, "edu" for education, "mil" for military, "net" for network resource, and "org" for other organization. Country codes are generally two digits: "ca" for Canada, "us" for United States, etc.

Larger organizations may decide to further divide their network by departments such as sales or support in order to keep track of what's what. Within such a scenario the address for sparky could be:

```
sparky.sales.software.com
```

Jane's address would become:

```
Jane@sparky.sales.software.com.
```

Although messages always travel from one computer to another when crossing a network such as the Internet, it is between users rather than computers that messages are addressed. Often specific computer names are not even included in the e-mail addresses people use. For example:

```
Jane@Software.com
```

This address indicates that the message is for Jane, who has some affiliation with the software.com domain. There is no computer with that name since software.com is the organization's domain name (the abbreviation `com` is always immediately preceded by the name of an organization). Yet the above address works because the DNS is a directory service, which allows computers to transform the above e-mail address into the address of a specific computer.

In order to deliver a message to Jane@Software.com, an e-mail program must determine the name of a computer that accepts mail for the `software.com` domain by asking the DNS. The DNS responds to this query by providing a list of computers that accept mail for that domain, one of which is sparky. The message is then sent to sparky.software.com, where Jane will find it the next time she checks her mail.

1.4.2 Multiple Addresses

There are a variety of reasons that you might want to assign more than one address to a single user:

- You may want an address which indicates what department users are associated with in their organization.
- You may want both first name (casual) and last name (formal) addresses.
- You may want to include common misspellings as valid addresses just in case.

For all the aforementioned reasons, Jane Doe has all the following addresses registered as valid e-mail addresses:

```
Jane.Doe@Software.com
Jane.Dough@Software.com
Jane@Software.com
Sales@Software.com
```


1.4.3 Other Types of Addressing

Networks other than the Internet often use different addressing systems and directory services. When a network consists of only several dozen or even a few hundred machines, it is fairly easy for each computer to maintain a list of where all the other computers (and even all the other users) in that network are located. It is only with the advent of the huge Internet that it became impossible for every computer to keep track of the millions of other computers in the new virtual neighborhood.

For example, on some networks e-mail messages are simply handed from one computer to another until they reach the recipient, so that an address might look like this:[10]

```
computer3!computer2!computer1!recipient
```

The above address indicates that the message needs to travel to computer3, then to computer2, and finally to computer1, where it is delivered to the intended recipient. This kind of addressing is clumsy and limited in comparison to the DNS system described in Section 1.4.1.

1.5 Protocol Proliferation

E-mail works beautifully – most of the time. This section provides a quick overview of some of the defined standards, or *protocols*, which enable e-mail to work as smoothly as it does.

As e-mail has proliferated, it has done so in a variety of ways. The communications protocol used most often over TCP/IP[11] links is the Simple Mail Transfer Protocol (SMTP). SMTP has been tremendously successful as the protocol of choice on the popular Internet. The UNIX-to-UNIX Copy Protocol (UUCP) remains established on many older networks. X.400, part of the more recent Open Systems Interconnection (OSI) suite is used more widely in Europe, and can be used over TCP/IP connections.

X.400 has been less successful than SMTP, but some proprietary X.400, LAN-based e-mail systems have been successful on a smaller scale than SMTP. Such systems usually require a fairly homogeneous network and can in some cases become a liability when trying to communicate with the outside world.

While all of the various systems for message delivery work “domestically,” trying to transfer e-mail between two locally disparate networks using different protocols can be difficult. In general it can be done once you know how (and in many cases if you can afford to pay for some fancy software). Learning the trick will remind you of how often Mr. Gore’s information superhighway is still a dirt road where you have to experiment with various addressing formats to see what works.

---

10 This is an address used on UUCP (UNIX to UNIX Copy) networks.

11 TCP/IP is a network protocol which allows reliable delivery of data from one computer to another and is the basic building block of the Internet. It stands for Transfer Control Protocol/Internet Protocol.

Within a network such as the Internet these problems have been resolved. As long as mail clients and mail servers are configured correctly (and this has historically been a fearsome task), mail transfer should be a cinch.

1.6 Directory Services for Users

The addressing services used by computer programs to resolve an e-mail address were discussed in Section 1.4. But how do users find the correct address of the person they're attempting to write to in the first place?

This has been a considerable problem for e-mail, especially for large and rapidly growing networks like the Internet. Unfortunately, the problem is not fully resolved yet. There are a few basic tools available, and although none provide a comprehensive "network phone book," they do provide limited assistance in locating someone's e-mail address or other information about them. The most widely available of these tools up until now has been the finger service, but a relatively new directory protocol known as LDAP is currently gaining popularity, and could emerge as the standard for directory services in the near future.

1.6.1 Finger Service

While the DNS is a directory service that helps computers to find information about where other computers are located, there are also directory services which allow users to obtain information about other users. The most widespread such directory service is the *finger* service, which allows you to "finger" someone - *if* you already know their e-mail address.

Many mail clients can initiate finger queries, which can unearth interesting and sometimes useful information about the person who is queried. This could be their phone number and mailing address, some kind of humor, or whatever other kind of information that person wants you to know.

For example, a finger query for Jane.Doe@software.com could return the following information:

```
Jane Doe
Jane.Doe@Software.com

525 State Street
Santa Barbara, CA 93101
USA
Tel: (805) 882-2470

What do you call a thousand developers at the bottom
of the sea...? (write to me if you want to know the answer)
```

Figure 1-9 An example of the kind of message you get if you request finger information for Jane at Software.com

Currently there is no completely comprehensive directory service that can find Jane Doe if you don't know her address at Software.com. Excellent directory services exist within certain small local networks, but none have yet achieved widespread success. However, by submitting multiple finger queries under just about every address where you might find the person you're looking for, this service at least gives you a "trial and error" directory service.

It is, of course, only a matter of time until some kind of virtual white pages arrives, and a server demurely asks you "what network please?" whenever you seek an address. But not today.

1.6.2 LDAP

An early attempt to create a comprehensive directory service was X.500, the general name given to directory services that are designed to the X.500 specifications. These specifications emerged (along with the second version of the X.400 specifications) as international standards in 1988, but were not widely adopted by the Internet community.

However, recently a directory service based on X.500 has begun to emerge as a possible Internet-wide standard. This directory service is the Lightweight Directory Access Protocol (LDAP), which is a sequel to the X.500 Directory Access Protocol (DAP). LDAP was originally designed as a front-end for an X.500 directory, but can also be used independently of X.500 to create a distributed directory service. This means that LDAP is a viable solution to the problem of creating a directory of Internet users and organizations.

1.7 E-mail Abuse

While the majority of this chapter focuses on the features and goodies that make e-mail so useful in the modern world, it is worth mentioning that e-mail is not without its problems. This section focuses on the abuses of e-mail in general, and Internet e-mail in particular. Although these issues – like all things Internet – are rapidly changing, anyone who is going to administer an Internet e-mail system should be aware of the possible problems and vulnerabilities of the technology.

1.7.1 Spamming

The most common abuse of the electronic mail systems of the world is the same as the most common abuse of the postal systems of the world: junk mail. Like most postal junk mail, the majority of junk e-mail is commercial advertising or some other unsolicited – and unwanted – garbage. On the Internet, sending out junk mail like this is popularly

known as “spamming,” with the junk mail itself called “spam” or unsolicited commercial e-mail (UCE).[12]

What it Is

When people in the Internet community talk about spamming, they’re usually talking about a particular type of junk e-mail, and not just any piece of unwanted e-mail. The messages that get labeled as spam are typically commercial advertising for highly questionable (and often illegal) products and services: get-rich-quick schemes, miracle diets, and the like. The following example is only slight exaggeration of a typical spam message:

```
To:       people-who-love-money@freecash.net
From:     FreeCash, Inc.
Subject:  FREE MONEY!!!!
------------------------------------------------------------
Dear Friend -

Would you like to have free money? Yes?! Then call us now!

FreeCash, Inc. has just patented an AMAZING new form of LEGALLY
generating FREE MONEY! You can take advantage of this INCREDIBLE new
service by simply calling our TOLL FREE(*) phone number, which will
get you in touch with our WORLD FAMOUS FINANCIAL EXPERTS! They will
MAKE YOU RICH!


(*) only $19.95 per quarter minute! Wow!
```

Figure 1-10 A prime example of unsolicited commercial e-mail; a.k.a. “spam”[13]

Of the millions of computer users who get garbage like this in their daily e-mail, practically nobody is asking for it.[14] They get these messages for the same reason that they get junk mail from the postal service: somehow their name and address got put on a

12 Why is junk e-mail called spam? According to Internet legend, the name comes from a famous sketch performed by the British comedy troupe Monty Python’s Flying Circus. In the sketch, a group of Vikings drown out the conversation in an English coffee shop by loudly and repeatedly chanting “spam, spam, spam...” The repeated and unwanted nature of the chanting is considered the inspiration for using the term to describe unsolicited commercial e-mail.

Meanwhile, the word SPAM© remains a trademark of Hormel Foods Corporation. We reluctantly use it here to describe e-mail only because the Internet community has adopted it as the standard terminology.

13 This particular message is also part of a sub-category of spam known as “MMF.” The acronym stands for “Make Money Fast,” a common subject for these messages. MMF messages typically advertise pyramid schemes and other illegal endeavors, and would probably constitute mail fraud if sent through the postal service.

14 Firms who distribute e-mail like this insist that the users who don’t want it are outnumbered by users who are clamoring to receive even more junk mail. However, taking an informal poll of your computer-using friends and coworkers will probably convince you otherwise.

mailing list, and the firms that buy the mailing list think that they can make money off these people by offering various services to them.

Note a couple of things about the above message. First, notice that the To: address does not include the address of a particular user; the destination address (`To: people-who-love-money@freecash.net`) is either the address of the mailing list that includes the recipient's address, or a totally meaningless placeholder. In most cases, the sender hides the true destination addresses of these messages by using the blind carbon copy (`BCC:`) feature available in most e-mail clients.

Second, notice that the return address (`From: FreeCash, Inc.`) does not conform to the address protocols that we looked at in Section 1.4 – you can't reply to this message, because there is no valid return address. Seldom does junk e-mail have a legitimate return address, because the senders know that most recipients will simply reply to their advertisements with notes like "Take me off your list right now!" Spammers don't want to download thousands of such messages whenever they distribute their ads, so they deliberately send their messages with only bogus addresses.

The nicer folks who send junk e-mail will at least include instructions that you can use to request to be removed from their mailing lists, but not all do. For that reason, it can be virtually impossible for you to stop the flow of junk to your e-mail mailbox once it begins.

It's worth mentioning that receiving garbage advertisements in your daily e-mail is only half of the story with spamming. A second manifestation of this same problem involves Usenet newsgroups, the Internet's community billboards. With newsgroups, spammers can send a single message to a multiple newsgroups and have it seen by all of the folks who regularly read messages posted there.[15] Although this type of spamming is indeed annoying and decreases the usefulness of newsgroups, it is not as problematic to the Internet as e-mail spamming.

Why it's Bad

E-mail spamming is worse than a simple annoyance that requires you to delete unwanted messages from your e-mail client. To understand why, again consider its paper alternative, junk mail.

While we think of postal junk mail as bad, it really isn't. Senders of junk mail buy stamps to pay the postal service for each piece of junk that they send; the more junk mail is sent, the more money the postal service is receiving, and the less they have to charge the rest of us to send our letters and packages. Just as television commercials fund the networks and allow us to watch sitcoms for free, junk mail funds the postal service and allows us to pay a mere 19 cents to have a postcard carried thousands of miles across the continent and personally delivered.

---

15 It was in Usenet groups that spamming originated. Because of the hostile response of Usenet users, and the proactive crackdown on junk postings by newsgroup administrators, senders of junk advertisements switched to direct e-mail as their medium of choice. However, newsgroup spamming continues.

However, the opposite is true for junk e-mail: in this case, it's the recipient of the message – as well as the folks whose systems carry it across the Internet – who foot the bill. If you're paying your Internet Service Provider (ISP) for your time online or the amount of server resources that you are using, then more junk e-mail means more time online to download it, more storage space on the server to store the messages, and more money out of your pocket – all for e-mail that you don't want! Meanwhile, your ISP's modems are being tied up by users like you who are downloading all of these unwanted messages, meaning that the ISP may have to add modems and phone lines or field complaints from customers who can't get on line.

The damage of junk e-mail increases with its volume. In one widely-publicized case, a particular ISP was receiving 1.8 million messages *per day* from a single spammer; not surprisingly, that volume negatively affected the quality of the ISP's service. As more and more of the Internet's bandwidth is taken up by millions and millions of junk e-mails, connection speeds for everyone will suffer, and the usefulness of e-mail itself will decrease because of the shrinking percentage of legitimate messages.

Note: *Another problem associated with spamming is the practice of using a stranger's mail server to distribute junk e-mail. This problem, known as relaying, is covered in Section 1.7.2.*

It should be noted again that not all advertising via e-mail should be considered "spam." Like all other mediums, the Internet and e-mail can be useful methods to distribute information about legitimate commerce. However, the dramatic increase in recent years of e-mail ads that are just plain garbage – as well as the widespread Internet backlash to that garbage – have caused a great deal of debate about what commerce does and doesn't belong online. Like the rest of the Internet, the spam debate continues to evolve.

1.7.2 Mail Relay

Another topic of concern to e-mail administrators is something known as *relaying*. Relaying can be a little difficult for beginners to understand, since there's "good" relaying and "bad" relaying. The relative goodness or badness of relay is generally measured by the types of messages being relayed, and whether or not the administrator of the mail server that is used for the relay deems it acceptable.

What it Is and Why it's (sometimes) Bad

The simplest definition of mail relay is that it happens whenever messages are given to a mail server which are destined for some other mail server. You do this with your mail client every time that you send e-mail to someone whose e-mail account isn't stored on the same mail server as yours; since you're only giving it to your mail server so that it can pass it along to some other mail server, you're using your mail server to *relay* that message.

Consider again the following illustration, which was shown in Section 1.3.1:

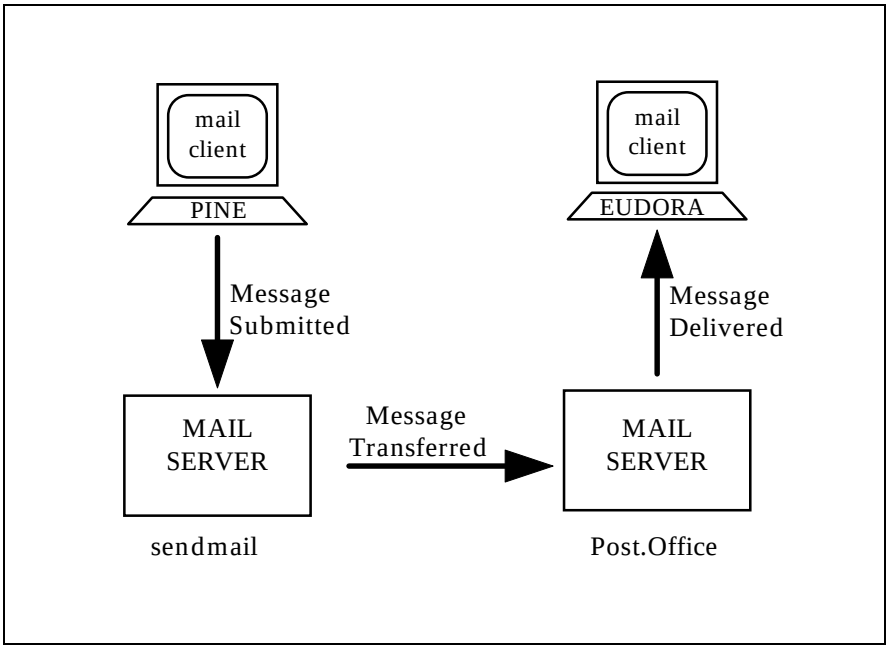

Figure 1-11 The Pine mail client relaying a message through a mail server.

In this example, the Pine mail client is giving a message to its mail server (in this case, sendmail) which the mail server can't personally deliver, since it's addressed to a user whose account is stored on another mail server. The mail server asks the DNS where the message should go, and then hands it off to the recipient's mail server. The first mail server has simply relayed the message.

So what's wrong with this? In theory, nothing – relaying is one of a mail server's primary functions, and e-mail could never get routed through the Internet if mail servers couldn't relay. However, relay becomes an issue when an extremely large number of messages are being relayed, or when the relayed messages are unwanted by their recipients. In other words, relaying becomes a problem when it becomes a tool for spamming.

To understand why this is a problem, imagine that you are an e-mail administrator and that some of your users are sending spam to users throughout the Internet. When your mail server is given a junk e-mail message addressed to 500,000 different people, it immediately gets to work delivering the message to all 500,000 recipients – just as it does for any other message addressed to any number of recipients:

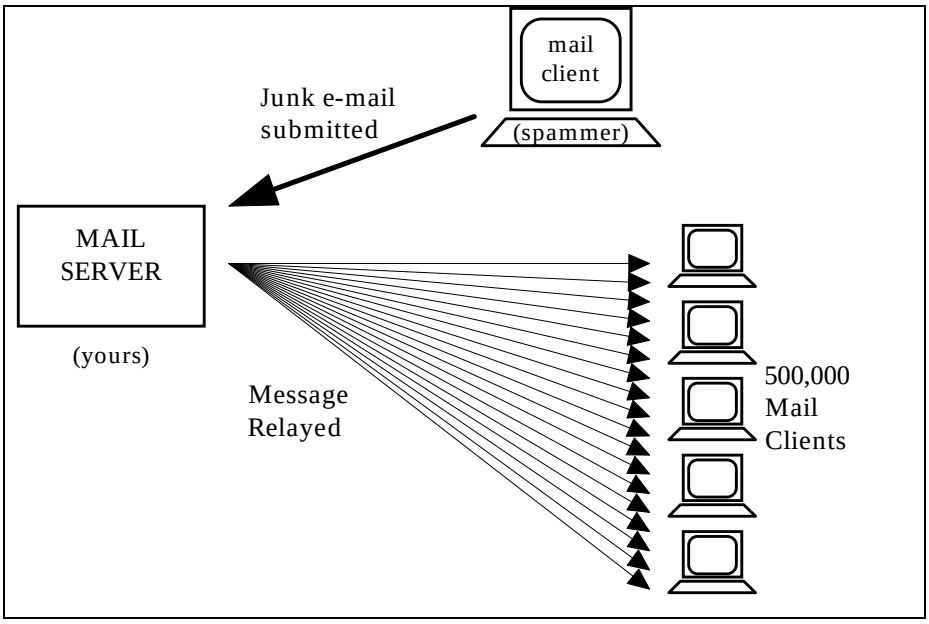

Figure 1-12 When a mail server is used to relay junk e-mail.

As you can imagine, sending out an e-mail message to half a million different users will take a fair amount of time. Also, since people tend to change their addresses now and then, this one outgoing message will generate a large number of undeliverable copies which will each be returned to – and handled by – the same mail server that is occupied with distributing the other 499,999 copies of this message. Not surprisingly, the performance of your mail server will become very, very slow while it is grinding through these tasks, and it may be temporarily unable to send or receive any other mail.

Notice another thing about the above illustration: from the perspective of the 500,000 recipients of the spam message, *this message is coming from your mail server!* Like any legitimate piece of e-mail, this spam message will include a header that indicates the name of the mail server that sent it. To the recipients of this e-mail, it appears that *you* are a spammer, since your mail server is obviously the instrument by which the junk e-mail was delivered. Because of this, sites that are trying to combat spam may send you unpleasant letters of protest, or decide to block any and all messages coming from your mail server.

Note: *That you and your mail server end up looking like the distributor of junk e-mail instead of the original sender is no accident: this is precisely why spammers like to use other folks' mail servers to distribute their stuff. That way, if their messages cause a lot of problems to the targeted mail server, or other sites block all mail from the distributor, or if recipients get mad and say unkind things about the sender, it's your problem and not theirs.*

In a nutshell, abusive relaying can result in your valuable mail server resources being hijacked by spammers and used to distribute junk e-mail – whether you like it or not.

Why it's Possible

Mail relaying is made possible by the openness of the SMTP protocol, which defines the way that computers exchange electronic messages. By definition, SMTP mail servers accept network connections from mail clients and mail servers from around the network (or the Internet), receive whatever e-mail messages that the connecting system gives it, and processes the delivery of the messages – regardless of the content of these messages, the number of recipients, or the likelihood that the recipients will want the message.

The following example shows the type of transaction that led to the mail server in Figure 1-12 attempting to deliver 500,000 copies of a junk mail message (a variation of the SMTP conversation shown in Figure 1-6):

```
Hello this is Computer1 (mail client)
>>> Hello this is the mail server on Computer2
I want to send you a message from Sir Spamalot
>>> OK
It's for these 500,000 people: (list of names)
>>> OK
Here's the message, ending with our secret handshake
>>> OK
Data, data, ... data, secret handshake
>>> Message received
Good-bye.
>>> Good-bye, I'm off to deliver this message to everybody ...
```

Figure 1-13 A paraphrase of a spammer's SMTP transaction.

Again, accepting messages and sending them to their ultimate destinations is the primary function of a mail server, so they're *supposed* to work this way. The folks who designed e-mail systems and the SMTP protocol never imagined that the technology would eventually be used to send "Make Money Fast!" messages throughout the world, so there is nothing inherent in the protocol for deciding what message should and shouldn't be accepted. Until a clear standard for combating abusive relay emerges, solutions to the problem will vary from mail server to mail server.

1.7.3 Denial of Service Attacks

A particularly destructive abuse of the e-mail systems of the world is something known as a denial of service attack. Unlike spamming and abusive relaying, which create problems for mail servers as a side-effect of distributing junk e-mail, a denial of service attack is created solely for the purpose of disrupting or stopping mail activity. These wanton acts of destruction are both unethical and illegal.

There are different kinds of denial of service attacks. The most common type that targets e-mail servers involves a program that opens many network connections to a server and maintains these connections despite having no interest in the services provided by the server. For example, a program can open multiple network connections to port 25 of a mail server (which is the port that servers "listen to" for receiving incoming messages),

and maintain those connections for the sole purpose of occupying the mail server. Such an SMTP transaction might look like this:

```
Hello this is Computer1 (denial of service attack program)
>>> Hello this is the mail server on Computer2
I have nothing for you to do right now.
>>> OK
Tell me again who you are.
>>> I'm the mail server on Computer2
I have nothing for you to do right now.
>>> OK
Tell me again who you are.
>>> I'm the mail server on Computer2
I have nothing for you to do right now.
>>> OK
( ... and so on)
```

Figure 1-14 A denial of service attack.[16]

It's obvious from the flow of the above conversation that the connecting client has no intention of transmitting a message, and that it is simply wasting the server's time. However, the server is designed to satisfy requests from various clients, not to decide the worthiness of each connection, so it will remain connected to this client until the client closes the connection or goes an extended period of time without issuing a command to the server.

Although the above example seems non-threatening, imagine if hundreds of clients were simultaneously executing the same pointless transactions on the same mail server. The result would be a server that is so busy giving useless information to clients who won't go away that it cannot be reached by the e-mail clients and servers that actually want to give it mail. True to its name, this kind of attack denies services to other computers and users.

Note: *Denial of service attacks are not unique to SMTP transactions, or even e-mail servers. Just about any type of server, including web and FTP servers, can be (and have been) targets of such attacks.*

16 In case you're interested, the SMTP commands paraphrased in this example are NOOP, which requests NO OPeration, and HELO, which asks the mail server to identify the host on which it is running.

2

E-mail with Post.Office

If your goal is to get Post.Office set up as fast as you can and you don't really give a hoot about the finer details, you may want to skip ahead to Chapter 3. The information presented here is not absolutely required for the operation of Post.Office. Rather, it is provided for those who prefer to start with an overall view of the Post.Office system, as opposed to a more hands on, trial and error approach. Contents include:

- A list of Post.Office features
- Profiles of the types of Post.Office users and what they do
- An overview of Post.Office system architecture

2.1 Features of Post.Office

As you read through the Post.Office features you will discover that Post.Office does a lot more than just trade messages with mail clients and other mail servers.[17] Although Post.Office is first and foremost a mail server, features such as mailing list management, finger service, and automatic replies do a lot to enhance the functioning of your e-mail system.

2.1.1 Versatile Mail Accounts

Of course, the heart of any mail server is its support of e-mail accounts. Post.Office mail accounts are more flexible and easier to use than what you're likely to find with other mail servers. While e-mail is complicated, on the whole Post.Office is not.

All mail accounts in Post.Office include the features described in the following sections.

Multiple Mail Addresses

An unlimited number of e-mail addresses may be assigned to a single account. You can even include addresses with different domains in the same account. Post.Office easily accommodates systems with multiple addressing formats and/or the requirement to host multiple domains.

17 If you don't know what mail clients and mail servers are, it's probably a good idea to go back to Chapter 1 and familiarize yourself with the basic terms and concepts of e-mail.

Multiple Mail Delivery Options

Every mail account in Post.Office can support the following types of mail delivery:

- **POP3 delivery**. The most common method of mail delivery, POP3 delivery stores messages in a "mailbox" on the server system until the user logs in with their mail client to retrieve the messages.
- **Forwarding**. As with ordinary postal mail, users can request to have their e-mail forwarded to another address. When forwarding is enabled, a copy of each message that arrives to the account is immediately sent on to the forwarding address.
- **Program Delivery**. For the vast majority of users, having e-mail delivered to a mailbox or forwarded to another Internet e-mail address is sufficient. However, there are situations for which advanced users need e-mail processed in some special way – archived, sorted, faxed, etc. Post.Office offers the ability to deliver mail to external programs that can carry out these additional tasks.

On UNIX platforms, Post.Office also supports a fourth delivery option:

- **UNIX delivery**. This option allows for the delivery of e-mail to a UNIX maildrop file, which allows pick-up of mail using legacy mail clients.

Every account must have at least one delivery option, but multiple options may be selected for a single account. For example, a corporate user may choose to have POP3 delivery to his mailbox on the company server *and* forward all mail to the e-mail address for a personal account accessed from a home computer.

Users can select their own delivery options (unless specifically prohibited by the Postmaster), so the Postmaster is not required to modify this information every time a user wants to change their delivery method.

Automatic Replies to Incoming Mail

The Post.Office Auto-Reply feature allows you to set up an automatic reply message to be delivered in response to all mail sent to a given account. There are three different auto-reply options available:

- **Reply**. The Reply mode sends an e-mail message to anybody who sends mail to a particular address. For example, you can use this feature to send a "virtual" brochure to anyone who contacts your company at the address `sales@your.company`.
- **Echo**. The Echo mode is the same as Reply, but returns the sender's original message along with the auto-reply message. You can use this to let people know that "so-and-so is no longer at this address, so stop mailing him stuff here, and no, we don't know where he is!" (or something more civil).
- **Vacation**. The Vacation mode is useful for users to set when they go out of town. It automatically responds to all messages with the user's vacation message (which all users can write themselves). Unlike the Reply mode, senders get only one copy of the vacation message regardless of how many messages they send to the account.

Mail Account Directory

Post.Office allows e-mail accounts to be optionally listed in a Mail Account Directory. This directory allows users to browse through a list of e-mail accounts and get names, e-mail addresses, and home page information for other users with Post.Office accounts. This information can be made available only to users with accounts in Post.Office, or can be shared via a web interface with all users on your network (or the Internet).

Postmaster Control Over End User Account Editing Options

By default, all users with e-mail accounts in Post.Office can execute the following account-related operations through the end user web interface:

- Change their password
- Select their mail delivery method(s)
- Enable and set their vacation message
- Edit their finger information
- Edit their directory information
- View the e-mail addresses that exist for their account
- View the access restrictions on their account
- View the Mail Account Directory

However, the Postmaster has the option of restricting access to any or all of these options in the web interface. For example, you can "switch off" the Select Mail Delivery Method option if you don't want users to have access to this functionality. This allows you to customize Post.Office behavior to match your organization's needs. See Chapter 4 for more information.

2.1.2 Mailing List Manager

In addition to the management of e-mail accounts, Post.Office includes a mailing list manager. A *mailing list* is a group of users who share information on a common topic. Mailing lists allow electronic messages to be distributed to all of the list's *subscribers* by submitting a message to a single address.

The mailing list manager is completely integrated with the rest of Post.Office – no extra options must be added for your installation of Post.Office to support mailing lists. The Post.Office mailing list manager offers a number of nice features, some of which are described in the following sections.

Compatibility With Existing Mailing List Managers

In addition to a web browser interface, the Post.Office mailing list manager includes an e-mail interface which is similar to existing mailing list manager programs, including the popular Majordomo mailing list manager. Instead of making experienced mailing list

users learn a bunch of new commands and operations, Post.Office has adopted many of the Majordomo conventions.

Remote User Access

The mailing list manager offers a limited interface to users who do not have e-mail accounts in Post.Office (that is, the rest of the teeming masses on the Internet). If you decide to allow it, users from outside of your system can request subscription to your mailing lists, and send and receive messages from those mailing lists. To ensure system security, external users are allowed these options exclusively and are prevented from further access to your mail server.

Not All Things to All People

Each mailing list has a series of policies that define how subscription requests, unsubscription requests, and messages submitted for posting should be processed. The policies for subscription requests are further divided by users who have Post.Office e-mail accounts and everyone else (those teeming masses again), so you can create a mailing list that is open to all of your users, but closed to everyone outside of your system. With posting policies, a distinction is made between the users who are or aren't subscribed to the mailing list, so you can choose to reject (or closely scrutinize) messages from non-subscribers while letting all subscriber mail go through.

Mailing List Limits

To control mailing list activity, as Postmaster you can control the number of subscribers allowed for each list and the amount of mail traffic you will permit per day. Reasonable default limits are established upon installation of the software, but you can customize those values to suit your particular situation.

Moderation Galore

Moderation is the practice of closely scrutinizing user requests relating to a mailing list. Moderated requests are held for the attention of the list owner, who will periodically sort through them and approve or reject the requests as he or she sees fit. Mailing list moderation is an option, not a requirement. For each mailing list in Post.Office, any or all of the following items may be moderated:

- Subscription requests from users who have local Post.Office mail accounts on this server
- Subscription requests from users who do *not* have mail accounts on this server
- Messages submitted by list subscribers
- Messages submitted by non-subscribers
- Unsubscription requests

2.1.3 Security

Security has been a major consideration throughout the design of Post.Office. There are five basic security mechanisms:

- **Limited Permissions**: Permission for Post.Office to run as root or administrator (super-user) is strictly limited to start-up. Once Post.Office is running, the program has no root or administrator privileges, which prevents users from compromising your system's security through the mail server. Also, users with e-mail accounts in Post.Office are not required to have logon accounts on the server system.
- **Passwords**: Any configuration change requires that the Postmaster (the mail administrator) supply a password. Users have their own passwords which allow them to retrieve their mail and make changes to their personal accounts.
- **General Access Restrictions**: Post.Office operations can be limited to specific locations so that configuration changes and mail retrieval can be made only from a specific host, or from within broader boundaries (for example, within a partially specified DNS address which does not include a host) as set by the administrator.[18]
- **Message Limits**: Both accounts and mailing lists in Post.Office have a series of associated limits and quotas. You can use these to control the amount of mailing list activity, limit the size of account mailboxes, and other items on a per-account or per-list basis.
- **Warnings**: Post.Office warns the system administrator in the event that it detects attempts made to break into the e-mail system and documents any such attempts.

For more detailed information on security options, see the discussions on general server security and mailing list policies in Chapter 4 and Chapter 7, respectively.

2.1.4 Support for Open Standard Protocols

Post.Office supports specific "open standards" protocols and is designed to accommodate mail transfer between non-compatible protocols. Post.Office incorporates the Simple Mail Transfer Protocol (SMTP), which allows message transfer around the world via the Internet.

You can also use Post.Office to route non-SMTP messages to a gateway (i.e., to route UUCP messages to a UUCP gateway). See the discussion of mail routing options in Chapter 4.

18 A host is a single computer and a broader boundary could be a set of computers. You can limit operations to as few or as many computers as you like. Anyone who tries to use Post.Office from an unauthorized computer is barred access to your e-mail system.

2.1.5 Remote Configuration and Management

All interactions with Post.Office are carried out via World Wide Web (WWW) forms: you request a form, fill it out, and submit the form to save your changes in the Post.Office database. There is no complicated syntax. You don't need to learn any programming languages to install or operate the system. You don't even need to do your configuration and management from the host where Post.Office is installed.

All forms include instructions on how to fill them out, so you should be able to complete them without even referring to this manual.

Post.Office allows ordinary users to make certain changes to their e-mail accounts using special forms of limited scope. While unable to make changes which would jeopardize the mail system, end users do not have to disturb the Postmaster to make changes which affect only their account, such as modifying their preferred method of mail delivery or resetting their account password.

2.1.6 Wide Area Network Design

Since Post.Office is designed to be a wide-area network messaging system, you aren't tied to a single local network. When your organization expands, so does Post.Office.

In fact, Post.Office is Internet-ready, so your messages can travel easily around the planet. If you are already on the Internet, Post.Office will handle mail for any number of Internet domains on the same machine.

2.1.7 Operating System Independence

From the perspective of the Postmaster, operating Post.Office on a Sun computer running the Solaris 2 version of UNIX is no different than operating Post.Office on a Windows NT machine. Since configuration is handled via universal web forms, system administrators are free to work from their favorite platform. You can commit to Post.Office without committing to any specific operating system or brand of computer.

2.1.8 Directory Information via the Finger Query Server

The finger server allows people to find limited information about one another if they know each other's e-mail address. With Post.Office, modifiable finger information exists for each mail account, which provides directory information (in addition to the Mail Account Directory) for every user who has an e-mail account with Post.Office. Users can modify their finger information without assistance from the system administrator.

2.1.9 sendmail Emulation

In addition to fully supporting SMTP, Post.Office supports features offered by sendmail, a freeware mail server which has achieved widespread use among UNIX users. In most cases, Post.Office acts as a drop-in replacement for sendmail (although there are differences). Mail system administrators who have customized sendmail extensively should refer to Chapter 11 to ensure a smooth transition to Post.Office.

Although sendmail is available exclusively on UNIX platforms, this functionality is duplicated on NT with the postmail utility, also described in Chapter 11.

2.2 Who Uses Post.Office

Just about anyone who does anything with e-mail can use Post.Office to do it. The Postmaster can use Post.Office to easily administer his/her organization's e-mail system; users with e-mail accounts can use Post.Office to specify how their mail should be handled, join mailing lists, and manage mailing lists; even computer users who don't have Post.Office e-mail accounts can use it to join mailing lists that have been made available to the public.

Each of these types of users is described in the following sections.

2.2.1 The Postmaster

There is always somebody who supervises the electronic mail system. This person is known as the *Postmaster*.

As e-mail systems have evolved from a few users to the millions who currently send messages across a variety of public and private networks, a need has evolved for institutions such as corporations and universities to appoint individuals to supervise day-to-day e-mail operations. These Postmasters are entrusted with maintaining the software (mail clients and mail servers) required to support e-mail, as well as providing certain information and services to users.

In addition to installing mail servers and providing users with mail clients, a Postmaster must in general keep an eye on the e-mail system to ensure that messages are being properly delivered in a timely manner. Postmasters receive error messages from the mail server when things go wrong. For example, the Postmaster is usually the first to know that a disk is full, or when a portion of the network is down, since messages start to queue up abnormally.[19] They can be notified of incorrectly addressed messages that the system

19 For a variety of reasons, messages often cannot be delivered immediately and often must be queued until they can be delivered, sometimes for a few minutes or hours, even for a day or two in some cases. When the time a message is queued becomes too long, it is a good bet that there is some kind of kink in the system. For example, in the aftermath of the 1994 earthquake in Southern California, some messages were queued for several days. Messages queued beyond a specified time limit are returned to the sender as undeliverable.

failed to deliver. The Postmaster can then try to correct the problem so that the message can be delivered, or have the message returned to its sender.

The Postmaster also functions as an e-mail guru for users, both by opening and closing e-mail accounts and by helping users with questions. Other e-mail related tasks that they may supervise include maintaining mailing lists and establishing aliases (additional names under which users can receive mail). Many Postmasters are also responsible for general system maintenance, system security, and user training.

What the Postmaster Does

The Postmaster's duties include the following:

- Define the domains for which Post.Office is responsible
- Define any additional routing options required for proper flow of mail through the server
- Create and delete e-mail accounts
- Create and delete mailing lists, and assign list ownership
- Set limits on user mailbox size and mailing list activity
- Set mail server performance and security parameters
- Set rules for the processing of undeliverable mail
- Respond to errors caused by mail activity, such as undeliverable or unreturnable mail
- Define the account management operations that are available to end users

Although it is not required, the Postmaster also typically serves as the e-mail "guru" for users, answering questions about their accounts and assisting them with whatever problems they encounter.

What the Postmaster Doesn't Do

We've made a real effort to make things simple for the Postmaster. Your users are able to change certain aspects of their own accounts, as described in the next section, so users will generally leave you alone. This should minimize the administrative aspects of your job as Postmaster.

Users can choose how their messages are delivered and change their password on their own. When leaving town for a few days, they can set up an auto-reply message to let people know that they're on vacation.

There are no arcane and apparently nonsensical configuration files to puzzle through, such as the `sendmail.cf` file. All changes can be made in simple web or e-mail forms, all of which come with instructions on their usage. We tried to do everything in English – if you catch us being overly technical or using esoteric jargon and abbreviations, e-mail us, and we'll fix it.

2.2.2 People With Post.Office Accounts

After the Postmaster, the next class of users are those who have e-mail accounts on the Post.Office server. These users are referred to as *local users*, and they can perform a wide range of operations related to their mail accounts. These users can also advance to the status of *list owners*, a class of local users with nearly Postmaster-like privileges over one or more mailing lists.

What Local Users Can Do

By default, all users with e-mail accounts in Post.Office can perform the following account-related operations:

- Change their password
- Select their mail delivery method(s)
- Enable and set their vacation message
- Edit their finger information
- Edit their directory information
- View the e-mail addresses that exist for their account
- View the access restrictions on their account
- View the Mail Account Directory for local users.

Note: *The Postmaster can choose to disable any or all of these account operations for local users.*

In addition to the above account options, all local users can also perform the following mailing list-related operations:

- Request a list of mailing lists available for subscription
- Review descriptions of the available mailing lists
- Request subscription to one or more mailing lists
- View their current mailing list subscriptions
- Request unsubscription from one or more mailing lists
- Request the list of subscribers for a mailing list (available only in the e-mail interface and only if permitted by the list owner)

A Higher Level of Local User: List Owners

List owners are a special class of local users to whom the Postmaster has delegated authority over one or more mailing lists. Any local user is eligible to own any number of mailing lists, over which he/she has wide-ranging administrative authority, including the ability to:

- Set policies for subscription, unsubscription, and posting
- Add or remove subscribers
- Approve or reject subscription and unsubscription requests
- Approve or reject submitted messages for posting
- Edit submitted messages before they are posted
- Set the available delivery options and delivery schedule

The many facets of list ownership are discussed in detail in Chapter 7.

What Local Users Can't Do

Despite the seemingly wide range of operations available to them, local users are restricted from doing any of the "important" things that have a large impact on Post.Office – these activities are the jurisdiction of the Postmaster alone. For example, local users cannot create or modify addresses for their accounts. They cannot modify the attributes of an account other than their own, cannot create accounts or mailing lists, and cannot set or modify ownership over a mailing list.

Local users are not required to have login accounts on the server system where Post.Office is installed. This was done deliberately to provide an additional level of security and prevent unnecessary clutter on the server by requiring the creation of new accounts solely for the purpose of allowing e-mail access. Mail users can, of course, have system accounts if desired – it simply isn't required.

2.2.3 People Without Post.Office Accounts

To facilitate the sharing of directory information, as well as the creation of public mailing lists, Post.Office can make certain information available to all users, whether or not they have accounts in your installation of Post.Office. This means that if you choose to share this information, users outside of your system – possibly anyone in the world with an Internet connection – can view information on your system's mail accounts, and subscribe to your mailing lists. While this may sound alarming, be assured that these outsiders are confined to a relatively modest and limited area of the interface, and can *never* modify any part of Post.Office. So relax.

A special part of the Post.Office interface is provided for these "public" users, who are also known as *remote users*. A public Mail Account Directory allows remote users to view the names and e-mail addresses of mail accounts on your system (if the accounts have been specifically listed there). Meanwhile, a special mailing list manager interface allows remote users to carry out the following mailing list operations:

- Get a list of the mailing lists available to them
- View descriptions of available mailing lists
- Request subscription to mailing lists
- Request unsubscription from mailing lists
- View their current list of subscriptions
- Request the list of subscribers for a mailing list

Refer to Chapters 5 and 7 for more information on remote users and the interface they use to interact with Post.Office.

2.3 Post.Office Architecture

Mail server functions are distributed among a number of software modules which work together to carry out message handling and other activities. This section gives you a bird's eye view of the Post.Office architecture. If this is the kind of thing you really dig, you can wallow in the nitty-gritty details in Appendix A, which is devoted exclusively to a discussion of the Post.Office system software architecture. If you just want to know how the stuff works and don't much care why, you may want to skip ahead to the chapters that discuss Post.Office operations.

Figure 2-1 shows Post.Office broken down into five functional chunks: the Post.Office managers, the MTA, a finger server, a password server, a POP server, the MTA, and the database which contains all of the information required to run the mail server. These items are all discussed in the sections that follow, along with the omnipotent Dispatcher which tells them when to run.

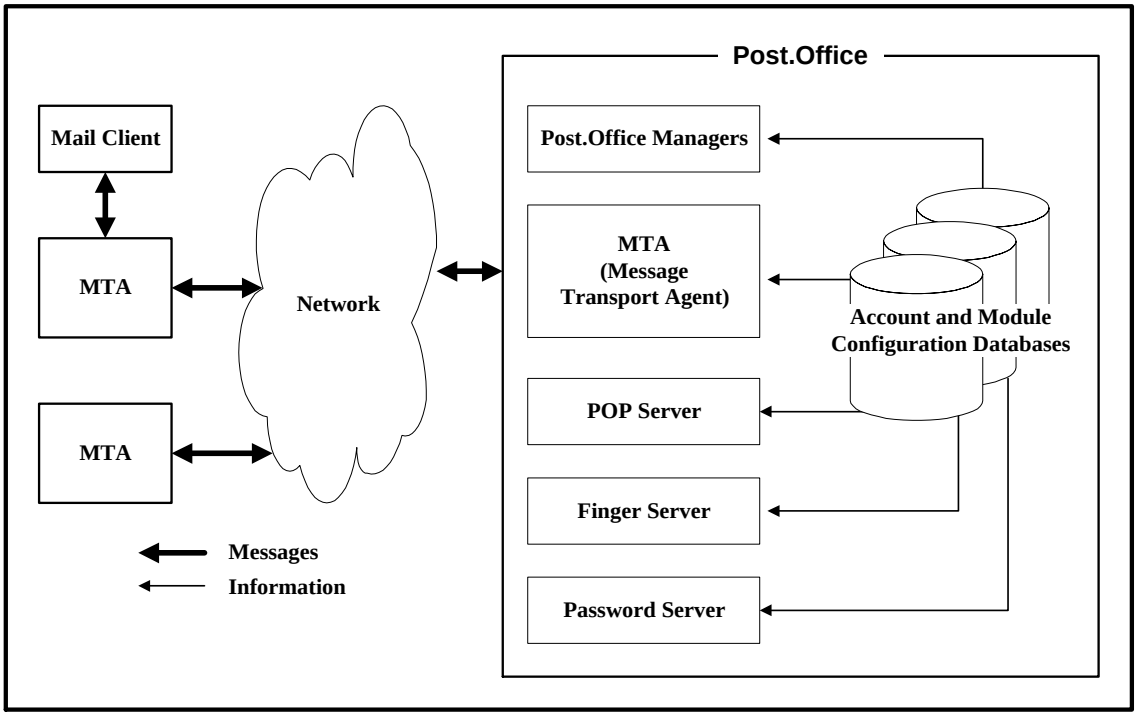

Figure 2-1 The primary components of the Post.Office mail server (which run under control of the Dispatcher). Arrows indicate the flow of messages between Post.Office and external mail servers and the flow of information from the account and configuration databases to the various Post.Office modules.

2.3.1 The Dispatcher

All mail server modules are coordinated by the Post.Office *Dispatcher* (on Windows NT this is a service; on UNIX platforms, a daemon[20]). The Dispatcher monitors all network ports[21] related to e-mail and starts up the appropriate modules to handle incoming connections. The Dispatcher also controls the number of processes which can be run simultaneously, thereby limiting the computer resources required to process e-mail.

The Dispatcher determines to whom an item should be sent based on the port on which the item was received.

- E-mail messages sent via SMTP are received on port 25 and directed to the MTA.
- Requests for access to account and configuration via the World Wide Web (WWW) are dispatched to the Web Manager. The port on which such requests are received may vary as it is configurable by the mail system administrator. Common selections are port 80, port 81, and port 8080.
- Requests for access to a POP mailbox are received on port 110 and directed to the POP server.
- Finger requests are received on port 79 and directed to the Finger Server.
- Password change requests from the Eudora mail client are received on port 106 and dispatched to the Password Server.

2.3.2 The MTA

The role of the MTA (which stands for Mail Transport Agent) is to exchange messages with other mail servers, accept messages directly from mail clients, and deliver messages to the appropriate location (as specified by the Post.Office delivery instructions for a particular account). To run the Post.Office system successfully, no further knowledge of this module is required. However, if you feel the need to delve further into this topic, feel free to review the detailed discussion in Appendix A.

---

20 A daemon is a program that is always running. Except for the Dispatcher, Post.Office modules run only while they are carrying out a task.

21 All e-mail transactions begin when one computer contacts another. Different ports are assigned to different types of transactions, so that when a certain port is contacted, the Dispatcher immediately knows what kind of transaction is involved and activates the appropriate module.

2.3.3 Account and Module Configuration Databases

The account and configuration databases play a crucial role in the Post.Office picture. They store the data that the rest of the modules rely on to carry out their tasks.

The account database holds all user account information, so it can be quite large. All modules refer to the account database whenever they need account information in order to process a message or otherwise carry out a task. By keeping all user information in one place, a single configuration change updates all modules at once.

Every Post.Office module has a database which contains the configuration information for that module. For most modules, this database is fairly small, containing only a few configuration options and a list of error messages.

Warning! When backing up your Post.Office mail server, remember to save the accounts and configuration databases in addition to your mailboxes. Failure to do so will result in the loss of critical information.

2.3.4 Post.Office Managers

Post.Office managers are the interface between the Postmaster and the mail system. These modules process the forms that you submit, following your instructions to update and maintain the account and configuration databases.

Note: *End users also communicate with these managers, but in a more limited sense. They are only allowed to edit database information for their personal mail account and for those mailing lists that they own.*

There are four Post.Office managers: the Account Manager, the Configuration Manager, the List Manager, and the WWW-Server.

The Account Manager handles e-mail forms requesting information from or changes to the accounts database only. The Configuration Manager handles e-mail forms requesting information from or changes to any of the various configuration databases. The List Manager handles e-mail messages sent to any mailing list request address, and provides information about or makes changes to mailing list information only. The WWW-Server responds to all requests received via web forms. It can retrieve and modify both account information and mailing list information, in addition to general system configuration options.

2.3.5 The POP Server

The POP server answers client requests to download mail. These requests are made by POP3-compatible mail clients, which retrieve and store the messages that Post.Office has received for their respective users.

2.3.6 The Finger Server

The finger server answers finger queries, a widely used feature which allows users to find out information about someone whose e-mail address they know. For example, if you query "Jane.Doe@Software.com," you'll get her phone number and address (as illustrated in Chapter 1).

2.3.7 The Password Server

The Password Server allows the Eudora mail client to communicate with Post.Office for the purpose of updating a user's POP3 mail account password. Refer to the Post.Office FAQ for information on using this particular feature.

2.3.8 Further Readings

The brief descriptions given above should provide most people with a more comprehensive outline of the Post.Office design than they will ever need in order to operate the system. However, if you're one of those people who needs or wants to know the polarity of the coil, the size of the plug gap and the required torque on the bell housing before driving the car, further details are available in Appendix A, which is devoted solely and exhaustively to Post.Office architecture.

3

Using the Web Interface

This chapter introduces the Post.Office web browser-based user interface. Among the topics discussed in this chapter are:

- Instructions for logging in to the web interface
- An introduction to Post.Office menus and forms
- Instructions for moving through the interface
- Tips for getting help through the interface (and elsewhere)
- Troubleshooting tips

3.1 Logging In

Although just about anybody with a web browser and a network connection to your mail server system can access the Post.Office web interface, they're not going to get very far unless they have the proper access. Access in Post.Office is verified by requiring users to supply their e-mail account address and the password defined for their account. Only if a user can supply an address and password for an existing account will they be able to enter Post.Office.

Note: *There is actually a small area for public directory information and mailing list activity available to all users, even if they do not log in to Post.Office. However, the options available in this public area are limited to viewing information specifically made available to remote users, and that doesn't really count as being "in" Post.Office.*

3.1.1 Know Where You're Going

The first step in the login process is getting the URL (that is, the web address) of the system where Post.Office is running. You should have received an e-mail message when your account was created that contains the appropriate URL. Enter this address in your web browser to access the Post.Office login form.

22 That is, an interface that you use through a World Wide Web browser.

If you didn't get the confirmation message, or if you deleted it without noting the web address for your Post.Office server, you'll need to contact your system administrator (the person who installed Post.Office) to get this information. If you are able to connect to the right server, but your web browser is getting something other than the Post.Office login form, refer to Section 3.5 for troubleshooting information.

3.1.2 Authentication Information Form

When you get your web browser pointed to the right URL, you'll see the Post.Office Authentication Information Form. This is a login screen that requires users to enter their e-mail address and password before letting them poke around the system.

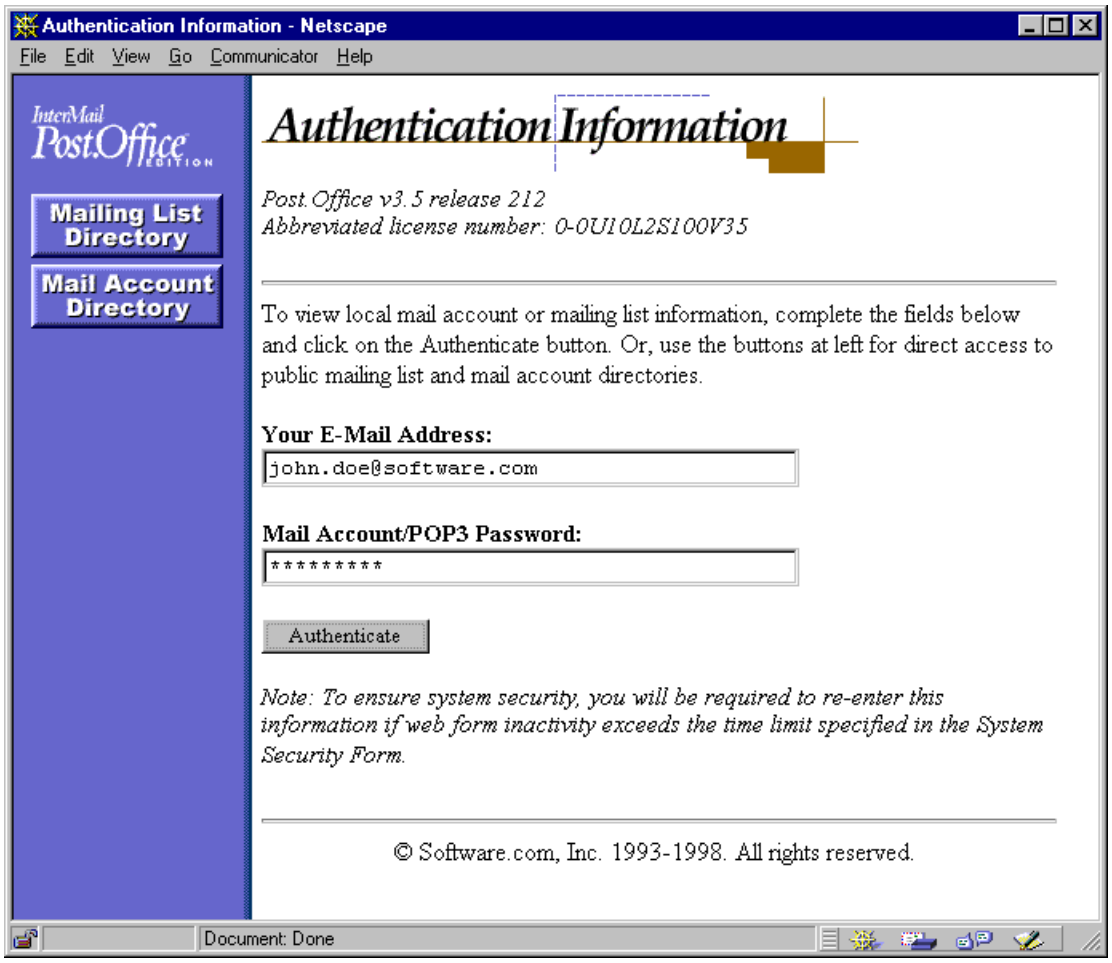

Figure 3-1: Authentication Information Form

You probably noticed the **Mailing List Directory** and **Mail Account Directory** buttons at the left of the form. These navigation buttons allow users who don't have e-mail accounts in Post.Office (and who subsequently cannot log into the system) to access information on the accounts and mailing lists hosted by Post.Office. These options are discussed in Chapters 5 and 7.

To log in to Post.Office, you must enter an e-mail address of an existing Post.Office account – as well as a password that corresponds to this address – in the Authentication Information Form. Once you've supplied this information, click the **Authenticate** button to enter Post.Office. Post.Office will verify that this information is correct before allowing you to access additional mail server menus. If the authentication check fails, you will be requested to re-enter your address and password.

3.1.3 Your Multiple (Login) Personalities

Post.Office is like NT or UNIX in regards to logging in to the system. As an administrator, you can log into these operating systems using your own personal account, which leaves you with restricted access to the system, or using the system administrator account (**root** in the case of UNIX, **administrator** in the case of NT), which gives you virtually unrestricted access.

Likewise, as the administrator of Post.Office, you can log in either as yourself (that is, your own personal mail account) or as the Postmaster (the administrative account), depending on the level of access you require. To log in as yourself, enter your personal e-mail address in the **Your E-mail Address** field of the Authentication Information Form. To log in as the Postmaster, enter `postmaster@`*`host.domain`*[23] in this field.

Since this is the Postmaster's administrative manual, we're assuming here that you want to log in as the Postmaster and carry out administrative duties. However, this will not always be the case, since logging in as your personal account is more convenient for modifying the attributes of your own e-mail account, and particularly when working with mailing lists. However, logging in as yourself restricts you to the same operations that any ol' end user can do. Just remember that when you want to operate on the system level (configuring Post.Office, creating accounts, handling undeliverable mail, etc.) you should log in as the Postmaster, and when you want to carry out operations specific to your own account (change your mail delivery options, subscribe yourself to a mailing list, moderate a mailing list that you own, etc.), log in as yourself.

3.1.4 Passwords

For each of the two accounts that you can use to log in to Post.Office, there is a password that must be given when logging in with that account. This means that when you log in to your e-mail account, you must supply your account's password, and when you log in to the Postmaster account, you must supply the Postmaster password. The administrator who installed Post.Office defined your e-mail address and password during installation, and also set the Postmaster password. If you don't know the appropriate passwords, ask your system administrator.

---

23 With your own host name and domain name, of course.

3.2 Of Menus and Forms

Before we get any further, you should know a couple of things about the Post.Office user interface. First, you should understand the difference between the two types of pages in the web interface: *menus* and *forms*.

3.2.1 Menus

Menus display lists of actions which can be performed (such as creating an account) or objects which can be viewed, edited, or acted on in some way (such as mailing lists). You generally can't "do" anything in a menu, but you can use a menu to get to a form for carrying out whatever task you have in mind. The navigation buttons **Account Admin**, **Mailing Lists**, **Deferred Mail**, **System Config**, and **Help** at the left of all menus allow you to easily move between the available top-level menus.

The Account Administration menu, which is the first thing you see after logging in from the Authentication Information Form, is an example of a menu. Like other menus, it shows you a set of links to forms that you can use to carry out specific operations. See Chapter 5 for a description of the options available from this menu.

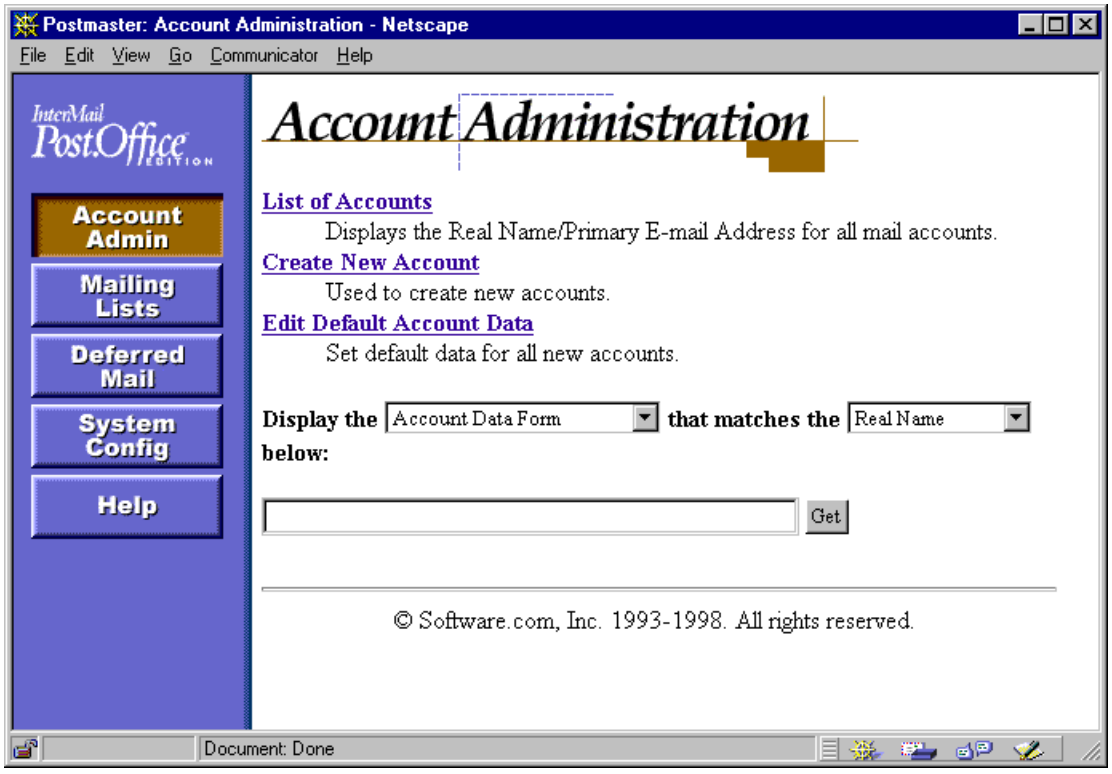

Figure 3-2: Account Administration menu

Like the Account Administration menu above, most menus display some predefined set of options. However, other menus – such as the list of available mailing lists – display lists of objects that may number in the tens of thousands. To avoid making you wait forever to see the entire list of objects, these menus break it up into easily digestible chunks of up to 50 objects. The following illustration of the List of Mailing Lists menu demonstrates this type of menu:

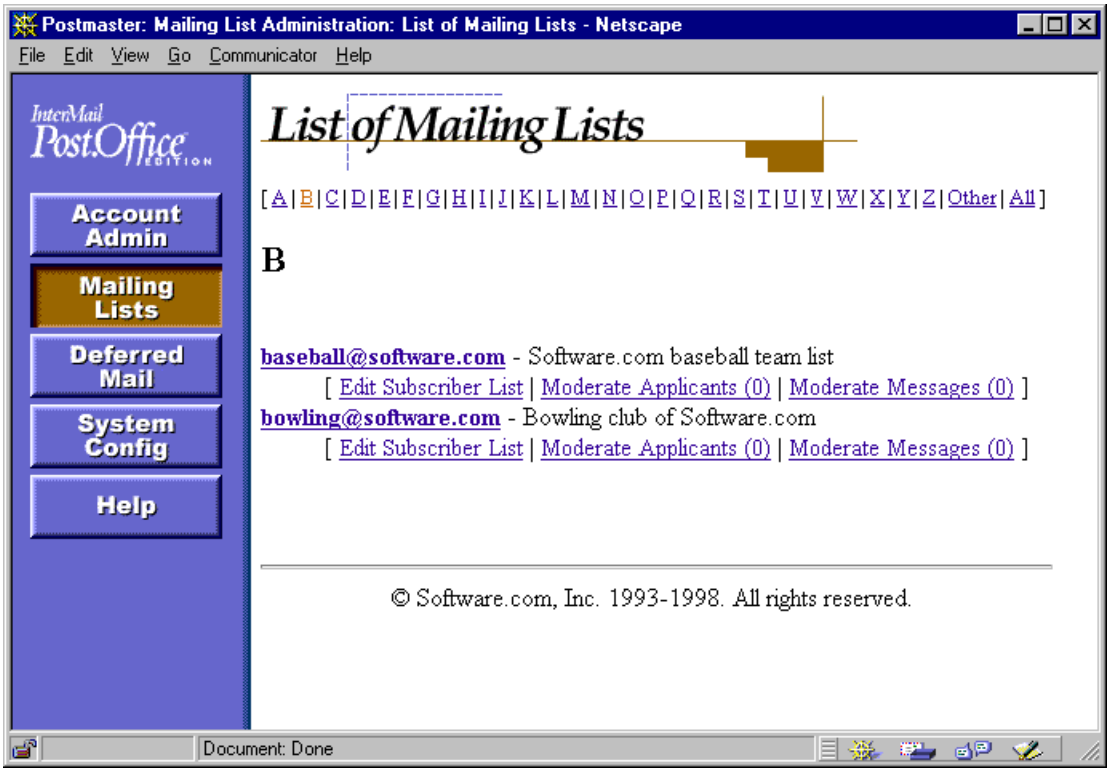

Figure 3-3: List of Mailing Lists menu

Notice that individual **A**-**Z** links at the top of the display area allow you to skip to other entries in the alphabetical list (i.e., click on the **B** link to view a list of mailing lists whose addresses begin with the letter B). Links labeled **Previous** and **Next** (not shown in the illustration above) are used to move forward or backward in the alphabetical list in groups of 50. The **all** link at the top of the menu, which displays the entire list of objects, is the only option that causes more than 50 entries to be shown at a time.

3.2.2 Forms

Forms, meanwhile, contain the data related to an object, such as an e-mail account. Most of the data displayed in a form can be modified and saved, and almost all of the actions that you perform in the interface take place in forms. Forms are distinguished by the lack of menu navigation buttons, as well as by the appearance of execution buttons that allow you to save or discard your changes.

Forms are typically invoked from menus. The Default Account Data Form, for example, is invoked when you click the **Edit Default Account Data** link on the Account Administration menu (Figure 3-2). Like other forms, it allows you to modify specific information and save the changes by submitting the form.

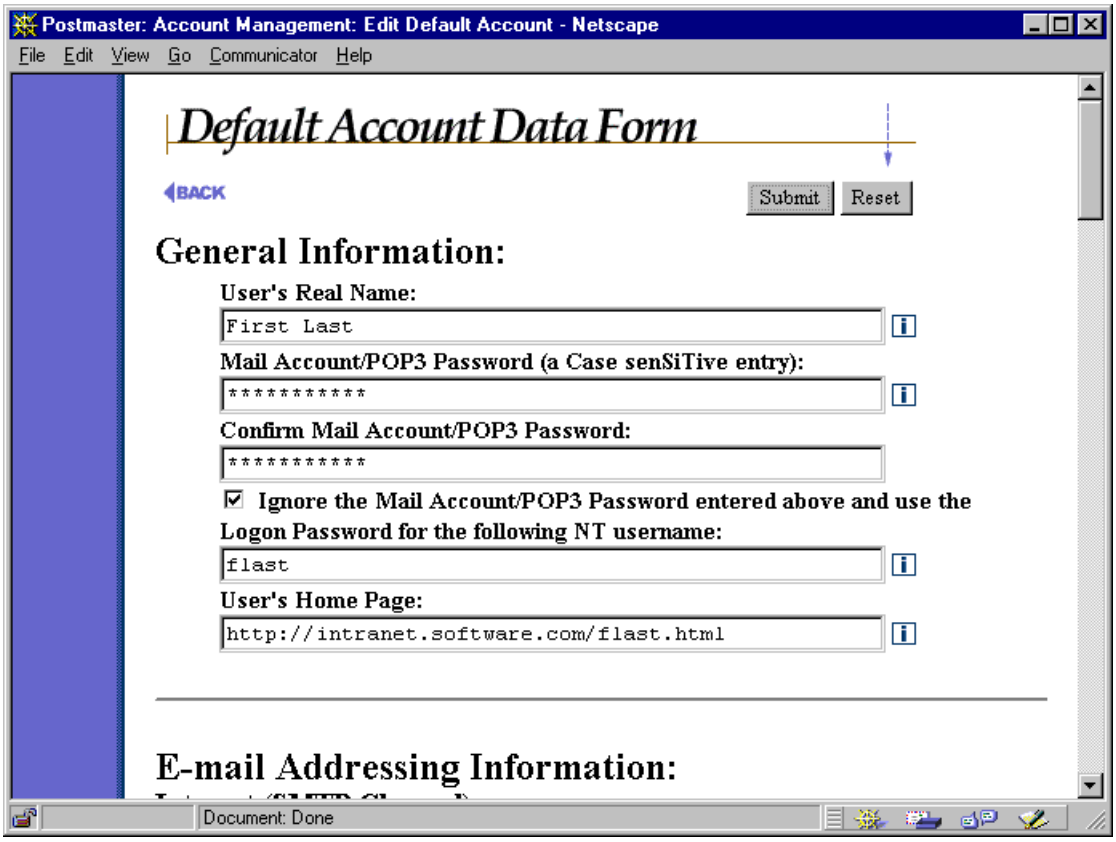

Figure 3-4: Default Account Data Form (only the top portion is shown; the complete form is much larger than what you see here)

Both the Account Administration menu and the Account Data Form are described in greater detail in Chapter 5.

3.3 Getting Around in the Interface

Along with the Account Administration menu mentioned above, there are four other top-level menus available to you: Mailing List Administration, Status of Deferred Mail, System Configuration, and Online Documentation (Help). These menus can be displayed at any time by clicking on the appropriate menu button (**Account Admin**, **Mailing Lists**, **Deferred Mail**, **System Config**, or **Help**) at the left side of any menu screen. You can switch from menu to menu at any time by clicking one of these menu buttons.

The Post.Office web interface is like a web site, which means that you'll be maneuvering through a series of pages that don't let you see everything at once. Unfortunately, this may get you lost if you don't remember how you happened to get to a certain form. For just this reason, we've given you a BACK link on every form that lets you move up a step or two in the form/menu hierarchy. The BACK link is visible in Figure 3-4 at the top left of the form. Unlike the browser's built-in "back" button, which may get you to a form with out-of-date information, this option returns you to the appropriate form or menu with all data updated for whatever modifications you've been making.

Along with the BACK link, most forms also include the execution buttons **Submit** and **Reset**. Clicking on the **Submit** button commits whatever changes you have made to data on the form, and typically closes the form and returns you to the top-level menu (Account Administration, Mailing List Administration, etc.). The **Reset** button allows you to cancel your changes by resetting all form fields to their previous values, and leaves you in the current form.

3.4 Getting Help

There are several ways to get information if you need help with something you're doing in Post.Office. First, there are online versions of all Post.Office manuals (including this one), as well as a list of frequently-asked-questions (FAQ), available to you in the web interface. Second, there is field-specific online help available in most forms. Finally, when all else fails, you can contact the technical support department of your Post.Office vendor, which exists specifically to help you with problems that you experience with Post.Office.

3.4.1 Online Documentation

To view the available online documentation, click on the **Help** menu button on the left side of any menu (if you're at a form that doesn't show the menu buttons, use the BACK link to move up the interface hierarchy until you see them).

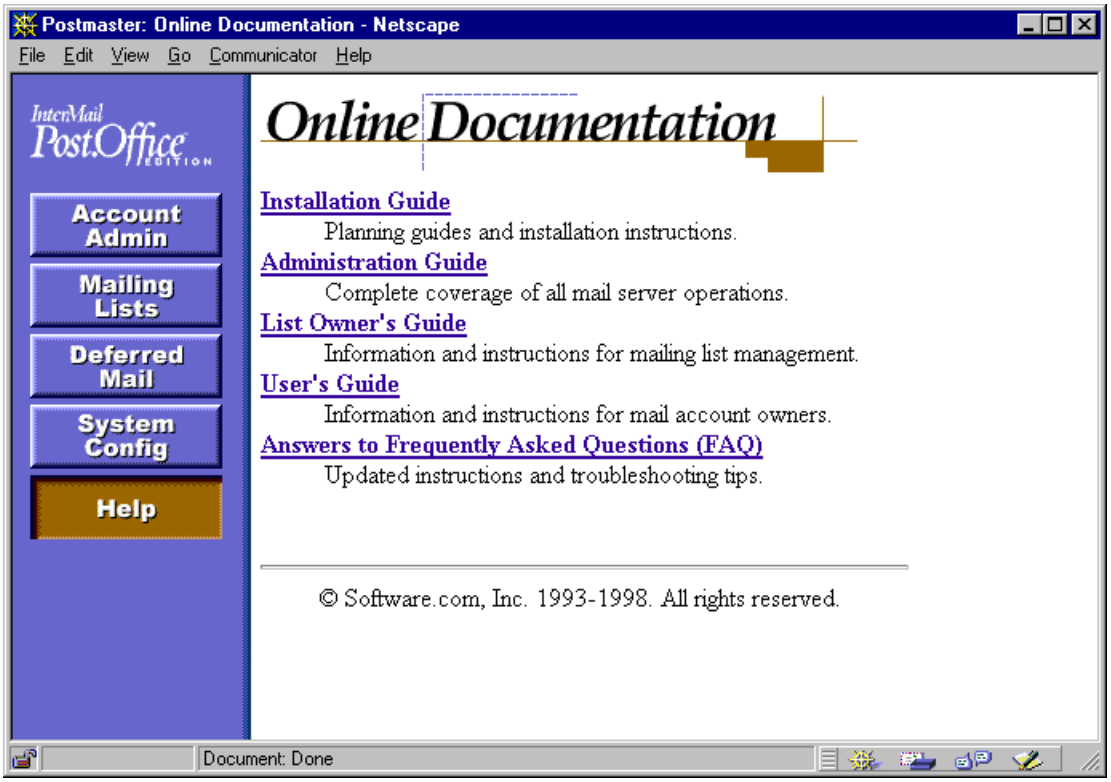

Figure 3-5: Online Documentation menu

The documents available from this menu are the following:

- Installation Guide
- Administration Guide (this very manual)
- List Owner's Guide
- User's Guide
- Answers to Frequently Asked Questions (FAQ)

The FAQ link takes you to the Software.com web site, where a frequently updated version of the Post.Office FAQ is kept for your convenience. The other links takes you to electronic versions of each manual, which are installed with Post.Office. All of the documents available from this menu contain a table of contents, which you can search if your browser supports word searching. Click on a link in the table of contents to view the corresponding information.

3.4.2 Help Links

Most forms include links to online help, which can be handy if you don't understand how to use a certain field. For example, if you are setting mail delivery options and have no idea what the Program Delivery option is all about, you can click on the help link for additional information. The help link is the graphic to the right of form fields that looks like this:

3.4.3 Technical Support

If you've looked through the documentation, read the help text, and still can't find out the information that you're looking for, you can call or e-mail the technical support department of your Post.Office vendor. Those folks are there to help you.

3.5 Troubleshooting

This section describes some common login problems, with some suggestions for dealing with them.

Password Doesn't Work

Check to see if the Caps Lock of your keyboard is currently on. The Post.Office password is CaSe-senSiTive, so accidentally setting the Caps Lock can leave a user shut out of the mail system. A simple mistake, but you'd be surprised how often this very miscue generates questions and concerns from users.

If you're logging in to your personal e-mail account, remember that if the NT Integrated Password option is enabled for your account, you must enter your NT login password – not your POP3 password – in the Authentication Information Form. The POP3 password defined for your account is saved in Post.Office, but is completely unused as long as the NT Integrated Password option is set. Refer to Chapter 5 for more information on this feature.

Bounced Back to Authentication Form

As a security measure, Post.Office will sign you out of the web interface if there is no activity for a period of time. You can set the specific number of minutes for this timeout period in the System Security Form (described in Chapter 4). This feature prevents others users from making modifications to the system if you log into the interface and then leave for the day with your web browser still running. If you get bounced back to the Authentication Information Form, simply log in again and continue your activities.

Correct URL goes to wrong web pages

When attempting to access the Post.Office web interface, you may find that entering the URL to the correct server is getting you to web pages other than the Authentication Information Form shown in Figure 3-1. This occurs when the same computer that is being used as a mail server is also being used as a WWW server; instead of connecting to Post.Office, you're connecting to the web site hosted by this system.

The greeting message that you received when your account was created contains the appropriate URL for logging in to the Post.Office web interface. If you didn't get a greeting message, or you unwisely deleted it and no longer have a copy, you should contact your system administrator (whoever installed Post.Office) to get the correct URL.

If solving the problem just isn't enough for you, and you need to know *why* you're solving the problem, what follows is a description of the situation.

This gets into the pretty technical areas of client/server computing, but here's what's going on: Server machines use "ports" to match server processes (such as a web server) to the client programs (like your browser) that will be interacting with them. Ports are simply numbers used to identify a process and distinguish it from the other thousands of processes that may also be running on the same computer. Whether you realize it or not, every time you ask a program on your client system to interact with a server machine, you are asking to use a specific port; otherwise, the server would have no idea which of its many available services you were trying to use.

Web servers generally use port 80 of the server system, so this is where your web browser is looking unless you say otherwise. So if you ask your web browser to go to the address

```
http://sparky.software.com
```

what you're really asking is to connect to port 80 of this computer and interact with whatever server process it finds there. Port numbers can be specified in URLs by using the ":#" notation at the end of the address, so the above address is equivalent to:

```
http://sparky.software.com:80
```

So far so good. Post.Office includes its own web server for its web-based interface, and like other web servers, it will run on port 80 by default. However, if the server system on which the mail server is installed is already running a web server, the Post.Office web server process must run on a different port number (otherwise, it would prevent all access to the web site). The default in these cases is port 81, but the administrator who installs Post.Office may choose any unused port on the system.

Therefore, if the server `sparky.software.com` is running both a regular web server (port 80) and the Post.Office web server (say, on port 81), you would point your browser to

```
http://sparky.software.com
```

or

```
http://sparky.software.com:80
```

to access the web site, and

```
http://sparky.software.com:81
```

to access the Post.Office web interface.

Again, in cases where the server is hosting both Post.Office and a web site, the Post.Office web server may be running on just about any unused server port. You still need to contact your system administrator for the complete web address (including port number) for logging in to the Post.Office web interface. But now you know why.

4

System Configuration

This chapter contains instructions for setting up Post.Office after it has been installed and is up and running. Included in this chapter are the following topics:

- A checklist of system configuration options that you should set immediately after installation.
- Complete descriptions of the System Configuration menu and all system configuration forms in the web interface.

4.1 Setup Checklist

There are a very large number of available Post.Office system configuration options, dealing with everything from security to performance to your preferences for the handling of undeliverable mail. However, many of these are fine-tuning parameters that you'll be revising only after you gain more experience with your mail system. If you've installed Post.Office for the first time, here's a checklist of items that you should set up immediately before you get your e-mail system going:

1. **Set your local mail domain(s)**. This is a list of all the mail domains and specific individual hosts whose mail is handled exclusively by your server machine. Entries in this list declare that your system is the definitive destination for these hosts and/or domains. The list of local mail domains is defined in the Mail Routing Form (Section 4.4).
2. **Set your address completion domain**. When mail arrives for a recipient whose address doesn't contain a domain, the specified address completion domain will be added to the incomplete address. This parameter is useful for enforcing desired addressing conventions, and is also defined in the Mail Routing Form (Section 4.4).
3. **Set the account options available to end users**. By default, all Post.Office users can access eight different options related to their accounts from the web interface: change their password, select their mail delivery method(s), set a vacation message, edit their finger information, edit their directory information, view their account's e-mail addresses, view their account's access restrictions, and view the Mail Account Directory. We recommend that you allow them access to all of these functions, but if your organization requires otherwise, you can remove access to any or all of these functions for your users through the End User's Account Options Form (Section 4.8).

4. **Restrict the hosts and/or domains from which Post.Office will accept configuration changes**. This security option ensures that only the specific hosts (or all of the host in a specific domain) can gain access to modify your mail system, meaning outside users are unable to get such access even if they know the Postmaster password. This option is set in the System Security Form (Section 4.11).
5. **Set policies for restricting mail relaying**. To stop the problem of abusing relaying (described in Chapter 1) before it starts, you should set up policies for restricting relay before bringing Post.Office online. This allows you to specify the systems and users who should be allowed to relay messages through your mail system. These policies are set in the SMTP Relay Restrictions Form (Section 4.5).
6. **Set your UNIX delivery mail program**. If you are running Post.Office on a UNIX platform, and plan to use the UNIX delivery method, you must specify the mail program that will handle the mail before your users can receive mail this way. The UNIX mail program is specified in the UNIX Delivery Configuration Options Form (Section 4.12).

4.2 System Configuration Menu

To access the Postmaster's web-based system administration interface, log in to the web interface as the Postmaster (refer back to Chapter 3 if you're not sure how to do this). After your login information is confirmed, you will be taken immediately to the Account Administration Menu. Click on the **System Config** menu button at the left of the accounts menu to display the System Configuration menu, which looks like the following illustration:

Figure 4-1 System Configuration menu

No less than 12 forms are invoked from the System Configuration menu. Every last one of these forms in discussed in the following sections, which display a copy of each form and descriptions of every field contained therein.

4.3 Channel Aliases Form

The Channel Aliases Form contains optional rules for special mail routing. These rules are called channel aliases, and are the most efficient way to screen incoming messages and immediately re-route those which need to be forwarded to another host. So be efficient and keep your computer speeding down the information superhighway!

Channel aliases can be used to route mail to an individual who has moved and is receiving mail on another machine. They allow you to request that all mail which arrives for a specific address be resent to another specific address. This involves modifying the message's envelope information to reflect the new destination address, so the change is permanent.

The Channel Aliases Form is invoked from the System Configuration menu by clicking on the **Establish SMTP Channel Aliases** link, and looks like the following illustration:

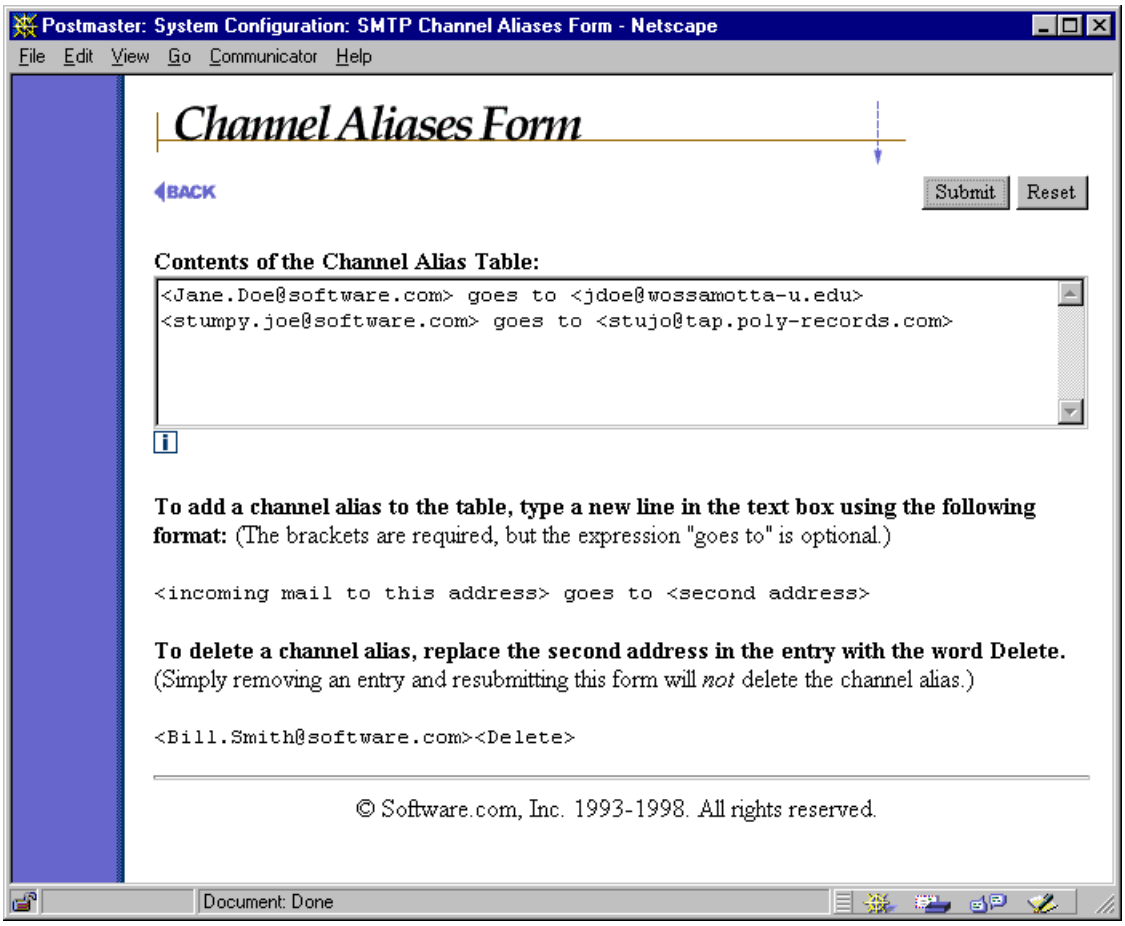

Figure 4-2 Channel Aliases Form

To set up an SMTP channel alias address, enter both the incoming and outgoing addresses in the large text area field. Each line of the table can contain one (and only one) channel alias pair. The format for alias entries is as follows:

```
<Mail-To-This-address> goes to <2nd-address>
```

The <angle brackets> around the addresses are required, but the "`goes to`" statement is not. So both of the following examples are valid channel aliases:

```
<Jane.Doe@domain> goes to <Jane.Doe@host.domain>
<Bill.Smith@host.domain> <Smith@newhost.newdomain>
```

Messages will be rerouted according to a channel alias only if its destination address is identical to an address listed in the Channel Aliases Table.

For example, if Ms. Jane Doe leaves her position in the private sector to serve you in our government in Washington DC, and she has been receiving mail to these addresses:

```
Jane.Doe@Software.com
jane.doe@sparky.software.com
jd@sparky.software.com
```

then we would probably set up a channel alias entry for each address, like the following:

```
<Jane.Doe@Software.com> <jd@whitehouse.gov>
<jane.doe@sparky.software.com> <jd@whitehouse.gov>
<jd@sparky.software.com> <jd@whitehouse.gov>
```

Deleting a Channel Alias

Deleting an alias from the SMTP Channel Aliases table is not done by simply deleting the alias from the list of aliases and submitting the form. Instead, you must replace the outgoing address you wish to delete with the word "delete," and then submit the form. For example, if you created the above example channel aliases and later want to delete two of them, you would submit changes like the following:

```
<Jane.Doe@Software.com> <delete>
<jane.doe@sparky.software.com> <delete>
<jd@sparky.software.com> <jd@whitehouse.gov>
```

Hint: *Because the deletion of a channel alias is often done incorrectly, you should go back to the Channel Aliases Form after deleting a channel alias to confirm that you were successful.*

4.4 Mail Routing Form

The Mail Routing Form covers the configuration of the SMTP mail channel, and it used to establish a variety of options specific to the SMTP configuration. This is one of the first forms that you should use to set up Post.Office after installation.

The Mail Routing Form is invoked from the System Configuration menu by clicking on the **Set Mail Routing Options** link, and looks like the following illustration:

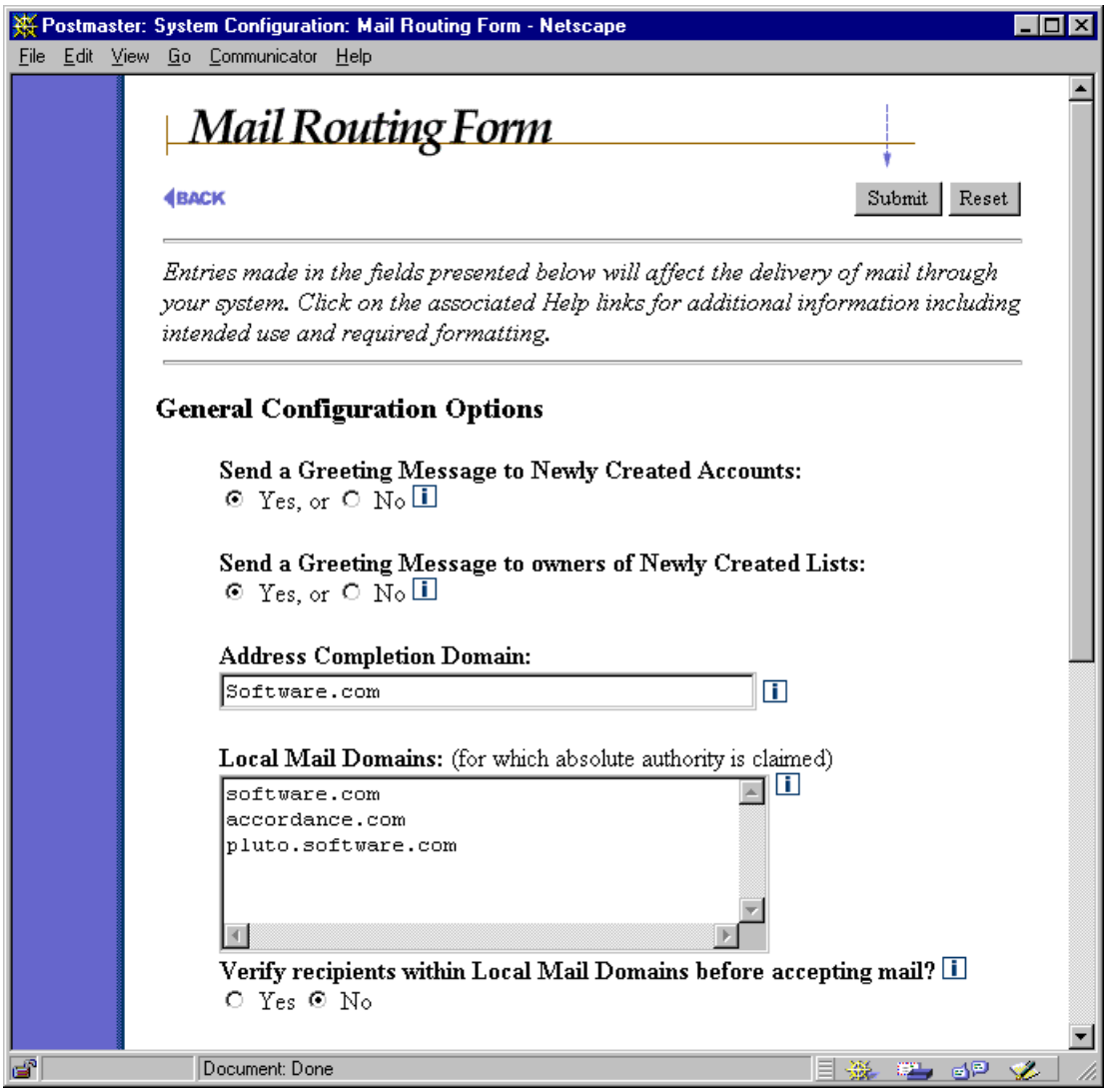

Figure 4-3 Mail Routing Form (part 1 of 2)

4.4.1 General Configuration Options

The fields in this part of the form affect global account-management options.

Send a Greeting Message to Newly Created Accounts

By default, Post.Office sends a greeting message to new users when their accounts are created. The greeting message informs the users of some of their account settings, and explains how they can customize their account. A sample greeting message is shown in Chapter 5. If you prefer that your users not get this greeting message (for example, if you want to keep them in the dark about the account options that are available to them through the web interface), select **No** for this field.

Send a Greeting Message to owners of Newly Created Lists

By default, Post.Office sends a greeting message to list owners when their mailing lists are created. The greeting message informs the list owner of some of the new list's settings, and explains how they can customize the mailing list via the Post.Office web interface. A sample list owner greeting message is shown in Chapter 7. If you prefer that list owners not get this greeting message (for example, if you don't want them to know about the mailing list configuration options available to them through the web interface), select **No** for this field.

Note: *Greeting messages are sent only at time of list creation; if you later add an owner to an existing mailing list, the new owner will not receive a greeting message.*

Address Completion Domain

When mail arrives for a recipient whose address is incomplete by SMTP standards, the domain specified here will be added to their address. If no domain is specified here, then the hostname plus domain of the server system will be assumed.

For example, if the Address Completion Domain is set to software.com, then mail addressed simply as

```
To: joe.schmoe
```

will have the address completion domain added and will therefore be sent to

```
To: joe.schmoe@software.com
```

For more information on address completion, refer to the discussion of mail flow in Chapter 10.

Local Mail Domains

This is a list of all the mail domains and specific individual hosts whose mail is handled exclusively by this machine. If a domain or host is listed here and mail comes in with an address at that domain or at that host, the message is delivered to an account on this machine (unless no account exists, in which case the message is handled according to the

settings of the Unknown Local Account error action). This machine is *not* the primary mail handler for any domain or host that is not listed here (but it can still receive mail for addresses in the unlisted domain if an account is set up with an appropriate address).

Take care in assigning local mail domains. The goal is to claim the maximum authority allowed without overstepping your bounds. If two servers handle your mail, each should claim authority for the appropriate host.domain (for example, `fido.software.com` and `sparky.software.com`), but only one should claim the entire domain (for example, `software.com`).

Verify recipients within Local Mail Domains before accepting mail?

If this option is selected, the system will refuse to accept mail that is addressed to unknown accounts within your local mail domains. Post.Office rejects the message with a standard notice; response to that notice (an error message, storage in a dead letter file, etc.) is controlled by the sender's mail client, so your mileage may vary. See Chapter 10 for a discussion of the effect that this option on mail routing.

Note: *Because this feature prevents acceptance of mail addressed to non-existent accounts, selecting* ***Yes*** *effectively overrides whatever options you chose in the Error Response Parameters Form for the handling of mail to Unknown Users.*

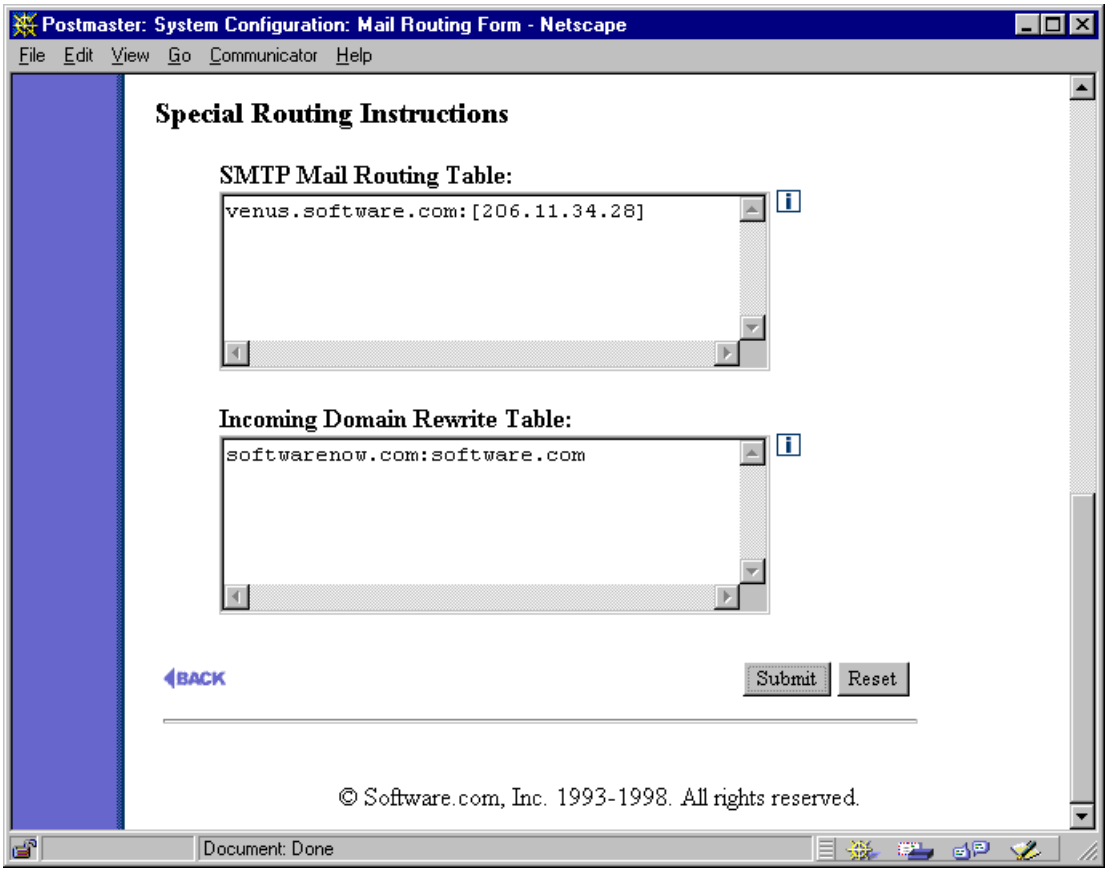

Figure 4-4 Mail Routing Form (part 2 of 2)

4.4.2 Special Routing Instructions

These options allow you to set options for controlling the routing of mail into or out of Post.Office.

SMTP Mail Routing Table

Entries made in this table allow you to route mail to a host other than the destination specified in the original address (that is, to a different computer). Normally this is used only in a situation where a firewall prevents direct access to the destination mail server, or when mail needs to be sent through a gateway to another network (such as a UUCP network). The format for entries in the table is:

```
original_domain:destination_host.domain
```

For example, to configure Post.Office to route mail to a UUCP gateway you might use the following entry:[24]

```
uucp_domain:uucp.gateway.host
```

If you want to specify an IP address for the destination host (as opposed to a host.domain name) you must enclose the IP address in a set of square brackets, as in the following examples:

```
domain:[206.11.34.28]
```

The asterisk (*) character is used as a wildcard which will match any string of characters. A default route can be set up using this wildcard so that all mail goes to a single machine as in this example:

```
*:mail.hub.machine
```

A default route should only be used if it is absolutely necessary (as in a gateway or firewall situation) since it puts an additional burden on the mail hub machine and can slow it down.

Add as many entries to the table as you need, but remember that the order is important, since it indicates the routing sequence. Keep that in mind when entering values that include a wildcard. For example, a normal set of entries to properly route internal mail from a host within the `software.com` domain to another internal mail server (or to a gateway machine prior to going to the firewall) would appear as follows:

```
software.com:mailserver2.software.com
*.software.com:mailserver2.software.com
msmail.com:msmail_gateway.software.com
*.msmail.com:msmail_gateway.software.com
*:fido.software.com
```

24 You should of course substitute the actual address for your UUCP gateway in the place of “uucp.gateway.host”.

These entries would cause all internal mail addressed to `@software.com` or `@anyhost.software.com` to be sent directly to the other internal mail server, while internal mail addressed to `@msmail.com` or `@anyhost.msmail.com` would be sent to the internal gateway machine; however, all mail sent to other domains on the Internet would go to the firewall machine, `fido`, where it would be sent on to its destination.

If you are using a firewall and your external DNS records do not include identification of internal mail servers or internal gateway machines, you need to add entries to the Mail Routing Table to route mail to those servers prior to routing to the firewall. This can be accomplished by using a "*" as the second host, or by specifying its hostname and domain.

The mail routing entries below route outgoing mail from a host within the `software.com` domain, which is protected by a firewall installed on the host `fido.software.com` to the mail server identified in the internal DNS records (which may differ from the server identified in external records):

```
software.com:*
*.software.com:*
*:fido.software.com
```

The above entries cause all internal mail (that is, all mail sent to `@software.com` or `@anyhost.software.com` addresses) to be sent directly to the host indicated by the internal DNS; however, all mail to other domains on the Internet would go to the firewall machine, fido, where it would be sent on to its destination. Such routing is useful in a firewall situation (or for a domain with intermittent SL/IP or PPP access to the Internet) where outgoing mail cannot be sent directly to some destinations.

The Mail Routing Table may also be used to route *outgoing* mail to a port other than port 25 (the standard SMTP port). This feature provides added flexibility and is particularly useful in gateway configurations where the gateway is capable of listening to a port other than Port 25.

To do so, append a "#" character and the desired port number to the end of the mail routing entry (as illustrated below).

```
*.domain:host.otherdomain#26
```

Note: *No rewriting of the destination address is performed on messages redirected according to entries in the Mail Routing Table. This means that the destination server must understand the original address on the message and handle it appropriately.*

Incoming Domain Rewrite Table

This option is used to rewrite domain names in the destination address of incoming messages. Domain rewriting allows the accounts at your site to receive mail at multiple domains, without needing to create alias addresses for each account.

For example, if the domains `accordance.com` and `rex.software.com` are rewritten to `software.com` in incoming messages, an account with the address `john.doe@software.com` can receive mail sent to any of the following addresses:

```
john.doe@software.com
john.doe@accordance.com
john.doe@rex.software.com
```

Note: *Domain rewriting applies only to the envelope of a message – the To: header of incoming messages is NOT rewritten.*

Each entry in the Incoming Domain Rewrite Table must consist of a domain name followed by a colon (:) and a second domain name. Both domains must be valid for the Internet, containing one or more words separated by periods, with each word containing only letters, digits, or hyphens. Both domains may include hostnames, but can not include wildcard (*) characters.

For example:

```
host1.domain.com:host2.some-other-domain.com
old-domain.net:new-domain.com
```

4.5 SMTP Relay Restrictions Form

The SMTP Relay Restrictions Form is used to prevent users and/or systems from relaying mail through Post.Office. Preventing mail relay can be a very important issue if your mail server is not behind a firewall or is otherwise left exposed to the great wide Internet. Refer to Chapter 1 for a refresher on the concepts of mail relaying and the havoc it can wreak on your mail server if left unstopped.

Warning! Preventing mail relaying is a complicated operation, and it is highly recommended that you use Post.Office's relay-prevention features only if you're experienced enough to know what you're doing. Incorrectly setting relay restrictions can cause you to prevent all mail – even the legitimate messages that you want your users to receive – from being accepted by Post.Office. Please proceed with caution.

The SMTP Relay Restrictions Form is invoked from the System Configuration menu by clicking on the **Restrict Mail Relaying** link. The form is divided into two sections: the first section is used to define the sources (specific computers and/or users) of relay mail that you want to prevent, while the second section is used to define what mail destinations (if any) should receive relay mail that you restricted in section one. Together these sections allow you to define a rule for the handling of relay mail, and then define exceptions to that rule.

The following illustration shows the first section of the SMTP Relay Restrictions Form:

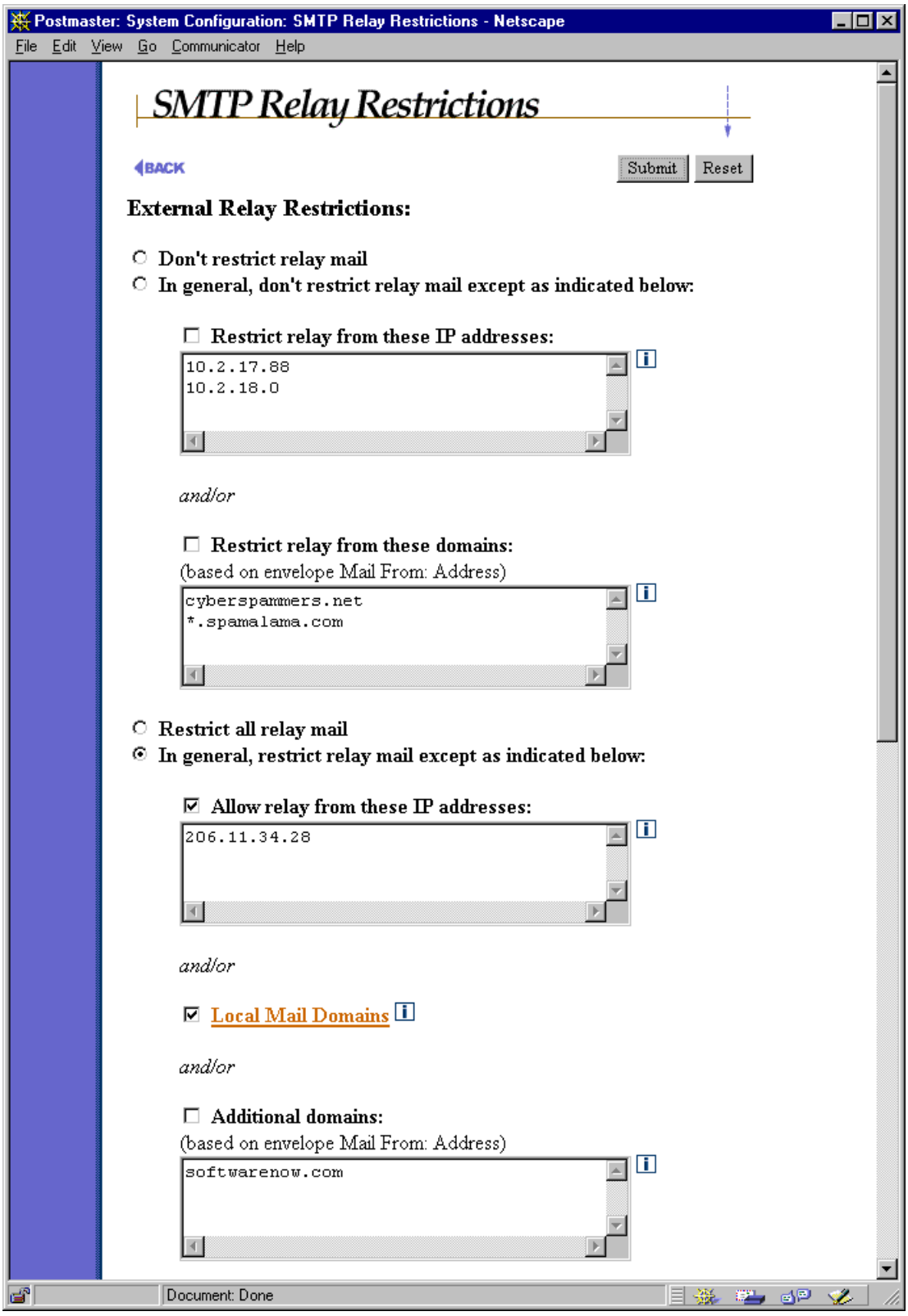

Figure 4-5 SMTP Relay Restrictions Form (part 1 of 2)

4.5.1 External Relay Restrictions

This radio button field allows you to specify rules for the restricting of relay mail. The four available selections are:

- **Don't restrict relay mail**. This option allows all users to relay mail through your mail server without restriction.

- **Don't restrict relay mail except as indicated below**. This option, which includes fields for specifying the systems and/or domains that are restricted from relaying, allows you to maintain a mostly open system which allows relaying except in special cases. When using this option, you would typically specify the IP address of a system that has been abusively relaying mail through your server, or the domain in the return address of these relayed messages.

- **Restrict all relay mail**. This option restricts all relay mail, including messages sent by your own local users (recall from Chapter 1 that a local user causes mail to be relayed whenever he or she sends a message which is addressed to a mail host other than his/her SMTP server). This option provides the ultimate in security, but in most cases is too restrictive.

Note: *If you select this option and then forget to define any exceptions to this rule in the form's second section, your mail server will never accept any mail! Unless that is your goal, be sure to use the fields at the bottom of this form to specify which domains are allowed to receive relay mail from your server.*

- **Restrict relay mail except as indicated below**. This option, which includes fields for specifying the systems and/or domains that are free to relay, allows you to maintain a mostly restricted system which allows relaying only in special cases. This is the typical selection for sites which want to restrict mail relaying. When using this option, you would typically enable the **Local Mail Domains** option to allow local users to relay mail, and also specify the IP address of systems which use your mail server as an SMTP hub.

When specifying the IP addresses of systems which are or are not restricted from relaying, you can enter an IP address that uses 0 (zero) as a wildcard to specify an entire network. For example, restricting relay from the IP address

```
222.33.44.0
```

restricts all relay mail from any machine with an IP address in the class-C network 222.33.44.

Note: *When allowing relay by IP address, you should always include the IP address 127.0.0.1, which refers to the host on which Post.Office is running. This allows the server system to "relay" mail to Post.Office, which is required for some legacy mail clients, such as elm.*

When specifying the domains which are or are not restricted from relaying, you can enter a domain name that includes the * character as a wildcard to specify all of the hosts in a domain. For example, restricting relay from the domain

```
*.promos.com
```

restricts all relay messages whose return address includes these domains, such as:

```
free.stuff@promos.com
incredible.credit.card@credit.promos.com
phone.service@phone.promos.com
```

When relay restrictions are set using domain names, Post.Office checks the return address on the envelope of every message in the system against the list of allowed or restricted domains. Because a user can easily alter his/her return address to include any domain, using domains to restrict or allow relaying is not as secure as restricting by IP addresses.

4.5.2 Allowing Delivery of Restricted Relay Mail

Remember again that the fields described above in the External Relay Restrictions portion of the form do not *prevent* relay mail, they *restrict* it. Mail that is restricted may still be allowed to be relayed by Post.Office, depending on the delivery rules that you set at the bottom of the SMTP Relay Restrictions Form. *If you don't stop the delivery of restricted relay mail, you are not preventing mail relay!*

To permit or deny the delivery of specific relay mail, use the fields in the second section of this form, which looks like the following:

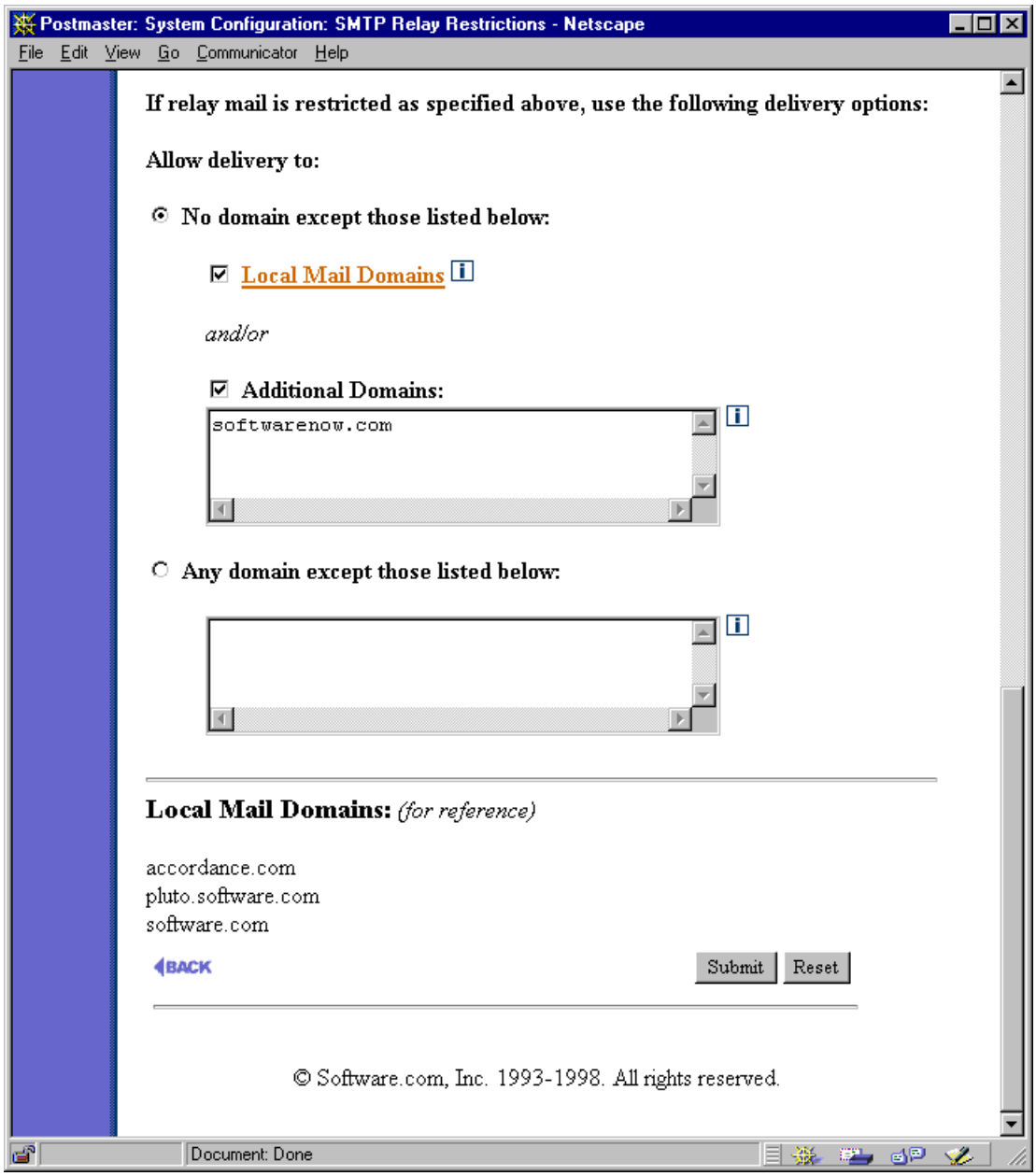

Figure 4-6 SMTP Relay Restrictions Form (part 2 of 2)

Allow delivery to:

This radio button field is used to define rules for the delivery of relay mail that is restricted according to the rules that you set at the top of this form. The available selections are:

- **No domain except those listed below**. This option denies the delivery of all restricted relay mail except when addressed to one or more specific domains. This is the recommended relay configuration. Included with this option are fields for allowing delivery of relay mail to Local Mail Domains and/or other domains. The Local Mail Domains option should always be enabled when using this delivery option,[25] while the Additional Domains field should include domains for which your mail server is an MX backup.
- **Any domain except those listed below**. This option allows the delivery of *all* restricted relay mail except when addressed to one or more specific domains. Remember that selecting this options means that you are *not* preventing the type of relay that you decided to restrict in the first section of the form.

Note: *The default delivery option specifies that delivery should occur only to local users. This means that if you set a bunch of relay restrictions in the first part of this form, and then ignore the delivery portion of the form, all of the mail you restricted in part one of the form will be rejected.*

As in the relay restriction fields, you can enter a domain name in the delivery fields that includes a * character as a wildcard to specify all hosts in that domain. For example, denying delivery to the domain

```
*.remote-net.com
```

prevents the delivery of restricted relay mail addressed to this domain, or to any host in this domain.

If delivery of a relayed message is prevented, Post.Office returns an error to the sender indicating that the attempt to relay mail to the specified domain was denied. The event is also entered in the Post.Office logs (if SMTP-RelayDenied logging is enabled). If a relayed message is addressed to multiple users, some of which are located in a denied domain, normal delivery will take place for the users whose domains are not denied.

25 Although relay mail is – by definition – mail that enters your system which is *not* addressed to users in your local mail domains, the fields in the External Relay Restrictions portion of this form make it possible for you to restrict mail addressed to your local mail domains. That's why you should always select this option if you're allowing delivery of relay mail to only a few domains – it prevents you from blocking messages which are legitimately addressed to your users.

4.5.3 Relay Prevention Examples

The following scenarios demonstrate the uses of the Post.Office anti-relay features, and give directions for dealing with specific situations of concern to mail administrators.

Scenario #1: *"A particular system is using my server to distribute junk e-mail. How do I stop this?"*

By reviewing the Post.Office logs, you should be able to get the IP address of the offending system (this information is given with the SMTP-Accept:Receive log entries that record information on incoming messages). To prevent relaying from this host, but otherwise leave your mail server available for relaying, execute the following steps in the SMTP Relay Restrictions Form:

1. Select the External Relay Restrictions radio button labeled **Don't restrict relay except as indicated below**.
2. Enable the check box above the text field labeled **Restrict relay from these IP addresses**, and enter the IP address of the offending system in this field.
3. In the delivery section at the bottom of the form, select the radio button labeled **No domain except those listed below**. This allows you to deny delivery of the relay messages from the system whose IP address you specified in step 2.
4. Enable the check box field for the **Local Mail Domains** delivery option. This allows the restricted system to continue to send messages to users within your local mail domain while still preventing this system from simply relaying messages through your mail server.
5. Enable the check box field for the **Additional Domains** option. In the text field below it, enter any additional domains that should be allowed to receive mail relayed through your server, such as sites for whom you are a backup MX site.

Note: *If you want to disallow all incoming mail from a particular system – even messages addressed to users in your local mail domains – use the mail blocking features described in Section 4.6.*

Scenario #2: *"Someone distributed the name of my mail server to people who relay junk e-mail, and now several users are relaying mail through my system, which is killing my server's performance."*

If you can't eliminate the problem of mail relaying on your server by restricting a few specific systems, change your relay settings to be more restrictive. Use the following settings in the SMTP Relay Restrictions Form:

1. Select the External Relay Restrictions radio button labeled **Restrict relay mail except as indicated below**.
2. Enable the check box field labeled **Local Mail Domains** in the relay restrictions portion of the form. This allows users whose return addresses match one of your local mail domains to continue sending mail through your installation of Post.Office.

3. Enable the **Additional Domains** option directly below the **Local Mail Domains** field that you enabled in step 2. In the text field below this check box, enter the other hosts and/or domains of users who should have access to send mail through your system. Remember that relay mail is restricted based on the sender's return address, so be sure to include the hostnames and domains in the return addresses of all users who should be using your installation of Post.Office as their SMTP server.
4. In the delivery section at the bottom of the form, select the radio button labeled **No domain except those listed below**.
5. Enable the check box field for the **Local Mail Domains** delivery option. This allows restricted system to continue to send messages to users within your local mail domain while still preventing it from simply relaying messages through your mail server.
6. Enable the check box field for the **Additional Domains** option. In the text field below it, enter any additional domains that should be allowed to receive mail relayed through your server, such as sites for whom you are a backup MX site.

Note: *The configuration in the above example is not incredibly secure, because a user from outside of your network can easily alter his/her return address to include one of your local mail domains; Post.Office will assume that this user is one of your own because of the return address, and will therefore allow them to relay free of restriction. Restricting relay by IP address, as shown in the next example, is much more secure than restricting by domain.*

Scenario #3: *"I want to restrict access to my mail server so that it allows only following: a.) all systems within my specific range of IP addresses can send mail to anyone; b.) all users in my local mail domains can receive mail from anywhere. How do I do this?"*

This configuration is the most secure (short of disallowing all relay), because it allows relaying only by systems within a network, as defined by a range of IP addresses. Meanwhile, users with e-mail accounts on this server will not be prevented from receiving legitimate message from any e-mail sender.

To set these relay and delivery rules, set the following in the SMTP Relay Restrictions Form:

1. Select the External Relay Restrictions radio button labeled **Restrict relay mail except as indicated below**.
2. Enable the check box field labeled **Allow relay from these IP addresses** in the relay restrictions portion of the form. Enter the IP address(es) that reflects the IP addresses of your network, using a 0 (zero) as a wildcard where appropriate. For example, entering the IP addresses

   ```
   222.33.44.0
   127.0.0.1
   ```

   will allow relay mail from any machine with an IP address in the class-C network 222.33.44, as well as from the server system itself (localhost).

Note: *Do not enable the* **Local Mail Domains** *option in the External Relay Restrictions portion of the form. Enabling this option allows messages to be relayed according to the return address on the message's envelope. Because a user can easily modify their return address to include one of your local mail domains, restricting relay by domain is not as secure as restricting by IP address.*

3. In the delivery section at the bottom of the form, select the radio button labeled **No domain except those listed below**.
4. Enable the check box field for the **Local Mail Domains** delivery option.
5. Enable the check box field for the **Additional Domains** option. In the text field below it, enter any additional domains that should be allowed to receive mail relayed through your server, such as sites for whom you are a backup MX site.

The effect of the above settings is that a message will *never* be handled by Post.Office unless it is either a.) sent from a system whose IP address is within your network; or b.) addressed to a user in your local mail domains. Again, this is the most secure configuration for preventing mail relay.

Scenario #4: *"The administrator of another domain is complaining that my mail server is being used to relay unsolicited mail to his users. How do I prevent outsiders from relaying mail to his server, while still allowing my own users to send mail there?"*

Although you'll probably want to deny outsiders from relaying mail through your system for security and performance reasons (as described in scenarios 1-3), you may decide to allow relay unless the people who receive this mail complain about it. In this case, you can prevent relayed mail from going to the complaining domain by using the following settings in the SMTP Relay Restrictions Form:

1. Select the External Relay Restrictions radio button labeled **Restrict relay except as indicated below**.
2. Enable the check box field labeled **Allow relay from these IP addresses** in the relay restrictions portion of the form. Enter the IP address(es) that reflects the IP addresses of your network, using a 0 (zero) as a wildcard where appropriate. For example, entering the IP addresses

   ```
   222.33.44.0
   127.0.0.1
   ```

   will allow relay mail from any machine with an IP address in the class-C network 222.33.44, as well as from the server system itself (localhost).

Note: *You can also enable the* **Local Mail Domains** *option in the External Relay Restrictions portion of the form if you want your users to be able to send mail to the domain in question. However, because a user can easily modify their return address to include one of your local mail domains, this method of restricting relay is not as secure as restricting by IP address.*

3. In the delivery section at the bottom of the form, select the radio button labeled **Any domain except those listed below**. In the text area field below this radio button, enter the domain that you don't want to receive relayed mail. This means that restricted relay mail (that is, all relay mail from users outside of your network) will be delivered to all domains except the one you entered here.

4.6 Mail Blocking Form

The Mail Blocking Form is used to prevent specific users and/or systems from sending mail – *any* mail – to your mail server. Like relay-prevention, Post.Office's mail blocking features are useful when dealing with distributors of "junk" e-mail who barrage your users with unsolicited mail.[26] However, because mail blocking is an all-or-nothing proposition that may prevent your users from getting mail that they really want, blocking is recommended only in extreme cases.

Warning! Mail blocking provides an even greater opportunity to shut off all mail delivery to your system than restricting relay mail. Proceed here with twice the caution that you did when setting relay restrictions.

26 As noted in Chapter 1, this type of junk e-mail is popularly known as unsolicited commercial e-mail (UCE), or "spam."

The Mail Blocking Form is invoked from the System Configuration menu by clicking on the **Set Mail Blocking Options** link, and looks like the following illustrations:

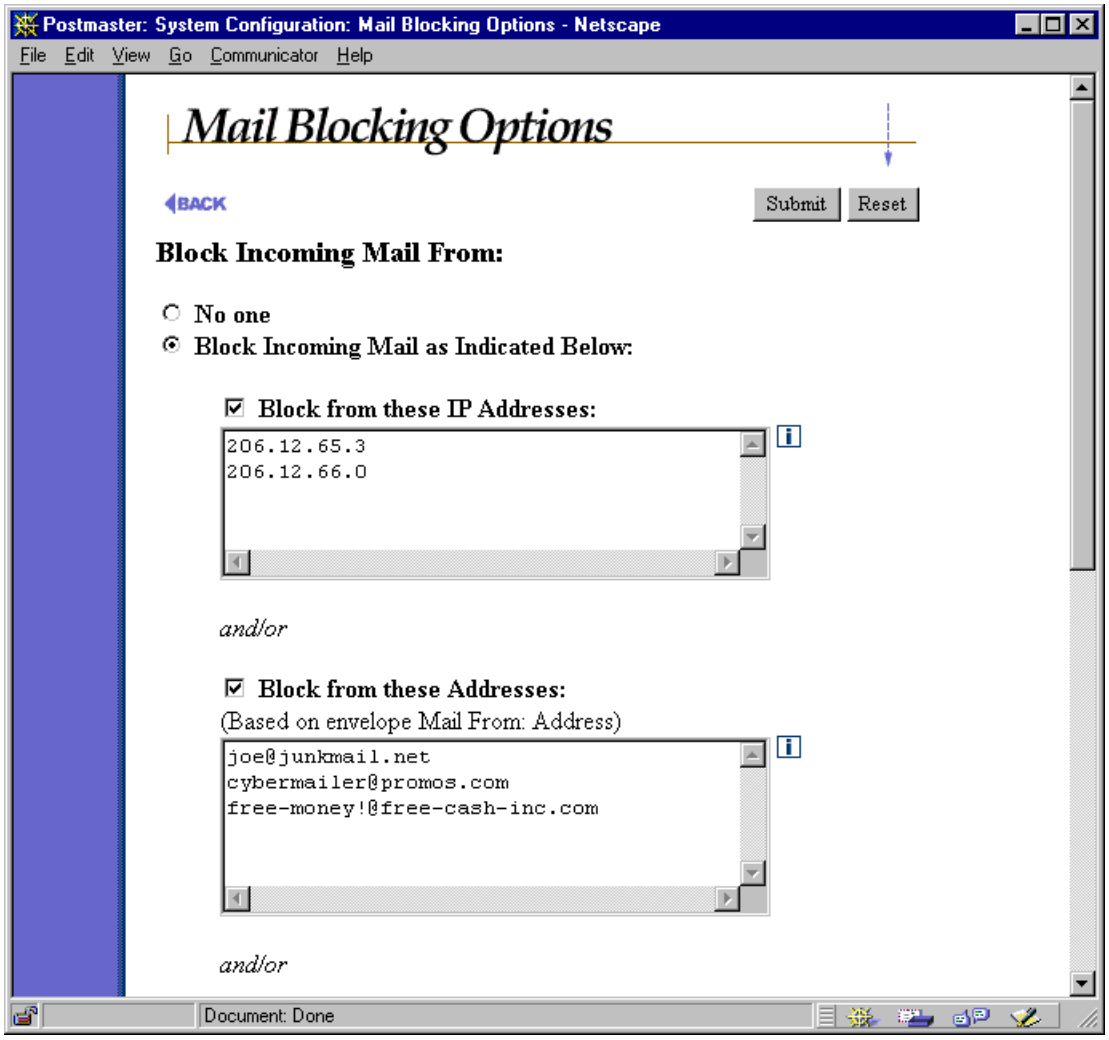

Figure 4-7 Mail Blocking Form (part 1 of 2)

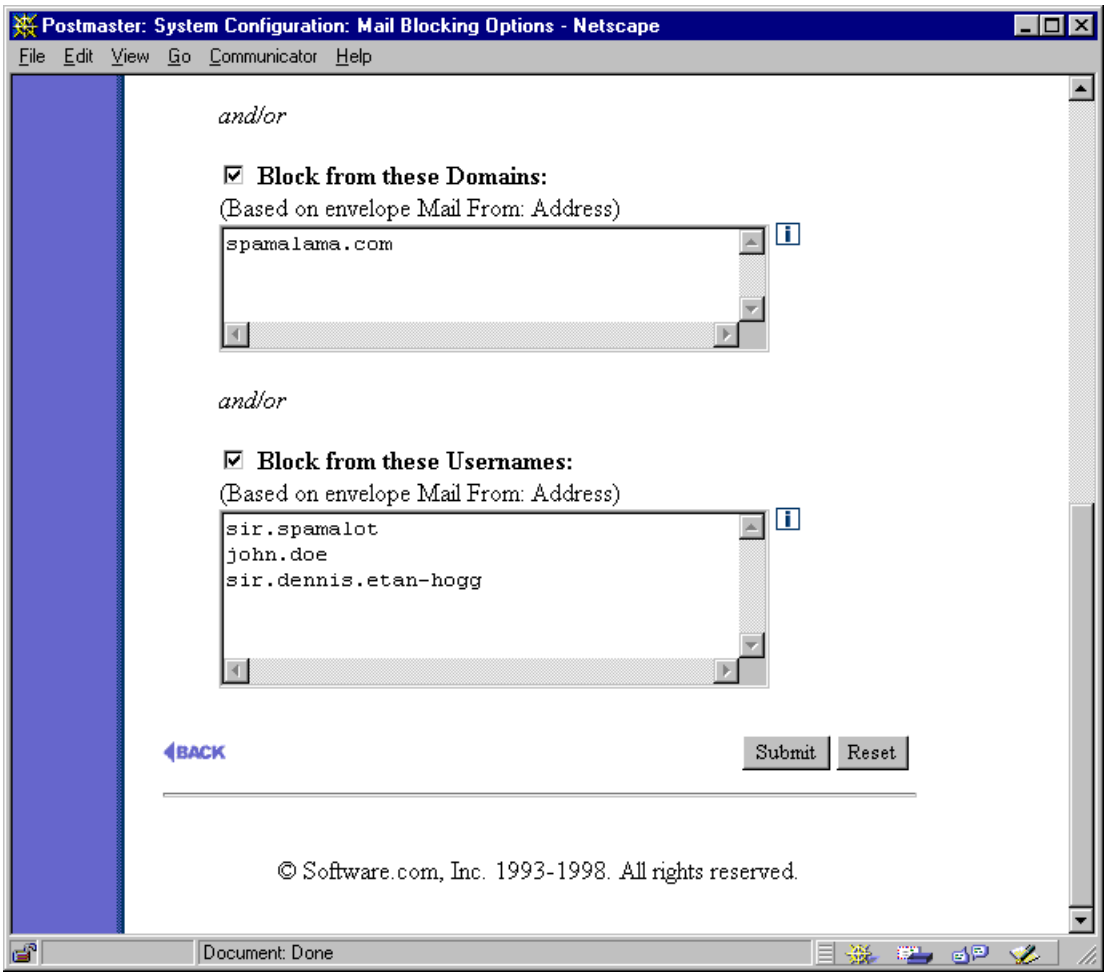

Figure 4-8 Mail Blocking Form (part 2 of 2)

By default, the selected radio button option for the **Block Incoming Mail From** field is **No one**, which disables all mail blocking. To enable blocking, change this option to **Block Incoming Mail as Indicated Below** and enable the check box fields for the types of blocking that you want to use.

There are four different criteria that you can use to block incoming mail: by IP address, and by the address, domain, and username of the sender's return address.

Blocking by IP address

This option allows you to block all SMTP network connections to Post.Office from specific computers or networks, as defined by IP address. When accepting network connections, Post.Office checks the IP address of the connecting system against the list of IP addresses specified in this field; if the IP address of the connecting system is listed here, Post.Office refuses the connection.

To block connections from an entire network, enter an IP address that uses 0 (zero) as a wildcard. For example, specifying the IP addresses

```
123.45.6.78
222.33.44.0
```

blocks all SMTP connections from the machine with IP address 123.45.6.78, or from any machine with an IP address in the class-C network 222.33.44.

Note: *If you use backup mail servers for your domain, you should likewise configure these mail servers to refuse connections from the IP addresses that you enter in this field.*

When Post.Office refuses a connection from a blacklisted system, an SMTP-Accept:ConnectionRefused event is entered in the Post.Office log (if you are logging this event). For example:

```
19970425164342-0700:SMTP-Accept:ConnectionRefused:[123.45.6.78]
```

This log entry indicates that a system with a blocked IP address attempted to connect to Post.Office. The IP address of the blocked system is given at the end of the log entry.

Blocking by E-mail Address

This option allows you to block incoming mail based on its envelope return address. When accepting messages, Post.Office checks the return address on the envelope of each message against the list of e-mail addresses specified in this field; if the return address is listed here, Post.Office rejects the message by reporting to the sending system that the destination address(es) of the message do not exist.

For example, specifying the e-mail addresses

```
joe@junkmail.net
cybermailer@promos.com
free-money!@freecash.com
```

blocks all messages which have a return address that is identical to one of these. When a message is blocked because of its return address, an SMTP-Accept:SenderBlocked event is entered in the Post.Office log (if you are logging this event). For example:

```
19970425164317-0700:SMTP-Accept:SenderBlocked:
[10.3.91.11]:<joe@junkmail.net>:3
```

In this example, the address `joe@junkmail.net` was in the list of blocked addresses, and was prevented from submitting mail to Post.Office. The number `3` at the end of the log entry indicates the number of intended recipients of the blocked message.

Blocking by Domain

This option allows you to block incoming mail based on the domain of its envelope return address. When accepting messages, Post.Office checks the domain of the return address of each message against the list of domains specified in this field; if the domain of the return address is listed here, Post.Office rejects the message and notifies the sender that he/she is not allowed to send mail to the destination address(es).

For example, entering the domains

```
promos.com
freecash.com
host1.someisp.net
```

blocks all messages whose return address includes one of these domains, such as:

```
incredible.credit.card@promos.com
free-money!@freecash.com
susie.queue@host1.someisp.net
```

However, messages from

```
jack.flash@someisp.net
more-free-money!@more.freecash.com
```

will *not* be blocked, because the domains of these addresses are not specified in the above example. Note that in this field, unlike domain fields on the SMTP Relay Restrictions Form, wildcards can not be used with domain names to specify all hosts in a domain.

When Post.Office refuses a message a blocked domain, an SMTP-Accept:SenderBlocked event is entered in the Post.Office log (if you are logging this event). For example:

```
19970425164317-0700:SMTP-Accept:SenderBlocked:
[10.3.91.11]:<offer@promos.com>:5000
```

In this example, the domain `promos.com` was in the list of blocked domains, and was prevented from submitting mail to Post.Office. The number `5000` at the end of the log entry indicates the number of intended recipients of the blocked message.

Blocking by username

This option allows you to block incoming mail from e-mail addresses that include one or more specific usernames (the username is the portion of an e-mail address to the left of the "@" symbol). This feature is useful if you want to block mail from distributors of "junk" e-mail who send messages from multiple domains but with the same username. When accepting messages, Post.Office checks the return address username on the envelope of each message against the list of usernames specified in this field; if the username of the return address is listed here, Post.Office rejects the message and notifies the sender that he/she is not allowed to send mail to the destination address(es).

To block all mail from a specific username, enter the username in this field. For example, specifying the usernames

```
incredible-offer
john.doe
```

blocks all messages whose return address includes these usernames, such as:

```
incredible-offer@junkmailer.com
incredible-offer@megamailer.com
incredible-offer@supermailer.com
john.doe@megapromo.com
john.doe@friendlyisp.net
```

When Post.Office refuses a message from a blocked username, an SMTP-Accept:SenderBlocked event is entered in the Post.Office log (if you are logging this event). For example:

```
19970425164317-0700:SMTP-Accept:SenderBlocked:
[10.3.91.11]:<incredible-offer@junkmailer.com>:30000
```

In this example, the username `incredible-offer` was in the list of blocked usernames, so the sender was prevented from submitting this message to Post.Office. The number `30000` at the end of the log entry indicates the number of intended recipients of the blocked message.

4.7 System Performance Parameters Form

The System Performance Parameters Form is used for setting a variety of options relating to server storage requirements and performance. Although Post.Office comes with a set of predefined defaults for these options, they will likely be among the first things that you modify after installation.

This form is invoked from the System Configuration menu by clicking on the **Establish System Performance Parameters** link, and looks like the following illustration:

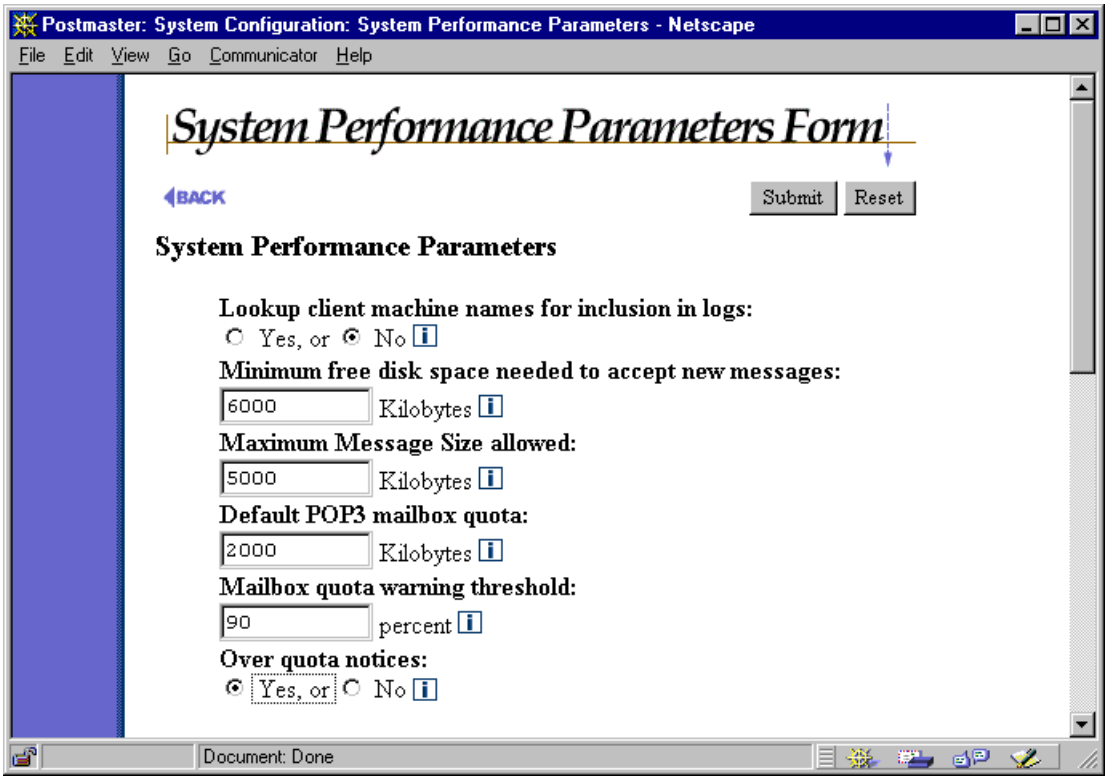

Figure 4-9 System Performance Parameters Form (part 1 of 2)

System Performance Parameters

Lookup Client Machine Names: Enable this option by selecting **Yes** if you would like to have Post.Office perform a name lookup (via the DNS) on all connecting client machines. If enabled, machines will be referred to by their domain names; otherwise they will be referred to by their IP addresses. Places where these names show up include the process table, the log file, and “Received” lines in message headers. If you handle a large volume of messages, be aware that selecting this option will slow down Post.Office slightly.

Minimum Free Disk Space: Post.Office tries to use all available disk space in its spool area for the processing of messages – new mail will be accepted as long as it fits on the disk. If you would like Post.Office to leave some of the disk empty, enter the desired space to be reserved (in kilobytes) in this field. If this field is left blank (the default option), no free disk space is reserved. Similarly, if the contents of this field are deleted, the reservation is set aside and Post.Office will continue receiving mail as long as it fits on the disk.

When Minimum Free Disk Space is specified, any incoming messages that – if received – would cause free disk space to drop below the reserved minimum are refused by Post.Office until enough space becomes available. Messages refused in this manner are not returned to sender, but are simply queued by the sending system for a later attempt.

If the Post.Office spooling directory and the mailboxes do not reside on the same disk, the check for minimum free disk space may be performed more than once for each message. First, the SMTP-Accept process will check the disk where the Post.Office directory resides; if this check fails, the message will be queued by the sending system for a later attempt. If the initial check succeeds, the message is accepted by Post.Office. A second check is run on the disk where the mailboxes reside if Mailbox-Deliver is called; if it succeeds, the mail is delivered to the appropriate mailbox. If the check fails, the mail is queued internally for a subsequent attempt at mailbox delivery.

Maximum Message Size: The entry made in this field indicates the maximum message size (in kilobytes) that will be accepted by your mail server. Acceptable values are from 64 to 1,000,000 kilobytes. Incoming messages that exceed the established limit are not accepted; they are returned to their sender with notification that the original message was too large, as in the following sample notification:

```
This Message was undeliverable due to the following reason:
Your message is larger than the destination computer is willing
to accept, so it was returned. The error message below indicates the
size of your message and the maximum size allowed by the receiving
e-mail system. You may be able to split your message into several
smaller pieces and have them delivered separately.

Size of this message: 116683 bytes

     Server maximum size:  65536 bytes
```

If the Maximum Message Size field is blank (the default value), no limit will be enforced and messages of any size will be accepted.

Note: *The limit specified here takes precedence over the maximum message size limit placed on each mailing list if this system-wide limit is lower.*

It's important to understand that the message size limit applies to the *total* message, including any attachments. This condition is further complicated by the fact that the conversion operation associated with attachments results in four characters being transmitted for every three in the original text, so a file that was originally 300k in size will add 400k when attached.

Warning! Although not required, you really should set some type of limit for this field – otherwise, your mail server may someday have to contend with mega-messages of dozens of megabytes in size. Although Post.Office can certainly handle such messages, depending on your system resources it may be doing nothing but processing the mega-message for some time.

Your particular organization may require some large message transfers, but if not, set a reasonable limit – say, 3 Mb – for this field. If your users frequently transmit messages with attachments, you should probably consider surveying your users to find average messages sizes before establishing this limit.

Default POP3 mailbox quota: The entry in this field is used as the maximum POP3 mailbox size for any user that does not have a size limit explicitly set in their account. Acceptable values are from 100 to 1,000,000 kilobytes. If this field is left blank (the default option), then no limit will be enforced and mailboxes can grow to any size. Similarly, if the contents of this field are deleted the system is reset to allow mailboxes of limitless capacity.

Mailbox size limits apply to only those accounts with POP3 delivery, which allows you to control system resources and prevent users from acquiring more than their fair share of disk space. Messages that would increase the size of a user's POP3 mailbox beyond the established limit are not accepted; instead, they are returned to their sender with a message indicating that the intended recipient's mailbox is full. The Postmaster is also notified.

Hint: *For ease of maintenance it is recommended that maximum POP3 mailbox size be established at the global level and passed to all individual accounts by default. Only exceptions to the global default should be entered at the level of an individual account.*

Warning! To avoid unintentional rejection of incoming mail, you should check current mailbox sizes before establishing the Default POP3 mailbox quota. Current mailbox size can be displayed on the List of Accounts menu (described in Chapter 5) for each account that uses POP3 delivery.

Mailbox quota warning threshold: To prevent users from reaching their mailbox size limits, it's a good idea to alert them when their accounts are approaching their quota. This field allows you to set a warning threshold that will trigger such a notification. Enter an integer value from 1 to 100 in this field to indicate the percentage of the quota that that triggers the warning. For example, if an account has a 2 MB mailbox quota, a quota warning threshold of 90 will cause the account to receive a warning message when the amount of mail in its mailbox exceeds 1.8 MB (90% of 2 MB).

Over quota notices: This option controls whether or not a warning message will be sent to an account when its mailbox has exceeded its quota warning threshold (defined in the field above). This notification informs users of their account's quota and their current mailbox usage, and provides information on deleting mail form the mailbox. Because users should be aware of their mailbox usage, we highly recommend that you use this feature.

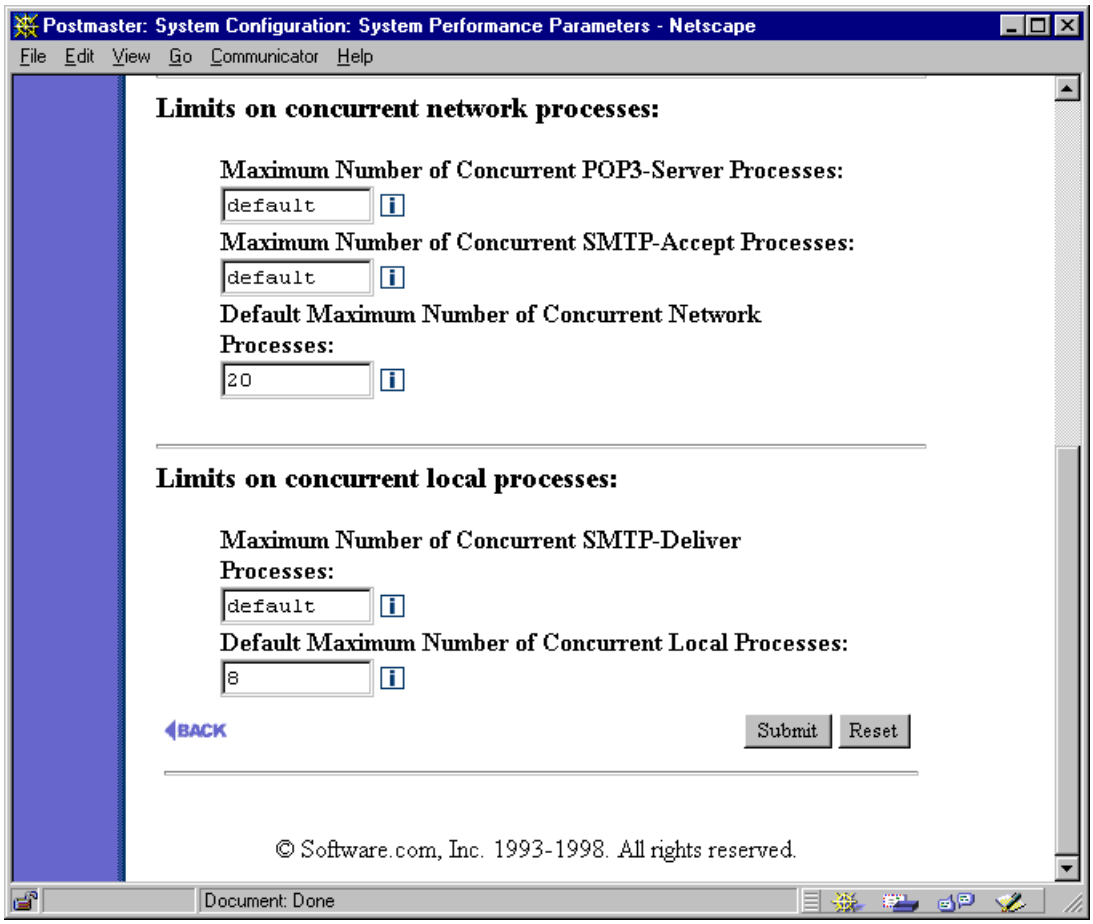

Figure 4-10 System Performance Parameters Form (part 2 of 2)

Limits on concurrent network processes

Maximum Number of Concurrent POP3 Client Connections. Entries in this field override system defaults and set an independent limit for the maximum number of concurrent POP3 client connections. If the field is blank (or contains the word "default") the default value will be applied to this network process.

Maximum Number of Concurrent Incoming SMTP Connections. Entries in this field override system defaults and set an independent limit for the maximum number of concurrent incoming SMTP connections. If the field is blank (or contains the word "default") the default value will be applied to this network process.

Default Max Concurrent Network Servers. The default value established in this field applies to all network processes (Finger-Server, the Password-Server, the POP3-Server, SMTP-Accept, and the WWW-Server) unless overridden by specific entries in the fields labeled Maximum Number of Concurrent POP3 Client Connections, and Maximum Number of Concurrent Incoming SMTP Connections.

It's important to understand that the default limit on concurrent processes applies to each process independently. If the default limit is eight, then eight instances of *each* network process can be run at the same time (not a total of 8 network processes).

This feature provides Postmasters with the flexibility required to fine tune their system for improved performance. These parameters control the mail system's use of processor time to either limit the mail system's impact on systems heavily loaded with other concurrent application programs or to ensure sufficient processor allocation to the mail system as necessitated by the number of users served by this Post.Office installation.

With regard to limits on concurrent processes, it is generally recommended that default values be set low and individual limits be increased as required.

Limits on concurrent local processes

Maximum Number of concurrent Outgoing SMTP Connections. The entry in this field overrides the system default and sets an independent limit for the maximum number of concurrent outgoing SMTP connections. If the field is blank (or contains the word "default") the default value will be applied to this local process.

Default Max Concurrent Local Processes. The value entered in this field applies to all local processes (Account-Manager, AutoReply-Handler, Configuration-Manager, Error-Handler, Mailbox-Deliver, SMTP-Deliver, List-Manager, List-Scheduler, List-Exploder) by default.

The default value applies to each process independently. It is generally recommended that defaults be set low and individual limits increased as required. The recommended default value for local processes is 5; it is highly recommended that you do not change this default, especially if you're using mailing lists. If you need to assess the local processes and what they do, you can go clobber yourself with the architecture nitty gritty in Appendix A.

Again, remember that the default limit on concurrent processes applies to each process independently. If the default limit is eight, then eight instances of *each* local process can be run at the same time (not a total of 8 local processes).

4.8 End User's Account Options Form

The End User's Account Options Form allows you, the Postmaster, to restrict the operations that are available to end users in the Post.Office web interface. By default, all user with mail accounts in Post.Office can perform the follow account-related tasks:

- change their mail account/POP3 password
- choose delivery methods, such as mail forwarding and POP3 delivery
- set their vacation message
- edit their directory information
- edit their finger information
- view the list of e-mail addresses for their account
- view the access restrictions on their account
- view the Mail Account Directory

However, you may decide that some of these options are not compatible with your organization's policies. For instance, you may decide that users should not be allowed to change their mail delivery method or view the access restrictions that you've put on their account. Because of this, the End User's Account Options Form allows you to "hide" one or more of these operations from users in the web interface, which prevents users from accessing (or even being aware of the existence of) these hidden options.

The End User's Account Options Form is invoked from the System Configuration menu by clicking on the **Define End User's Account Editing Options** link, and looks like the following illustration:

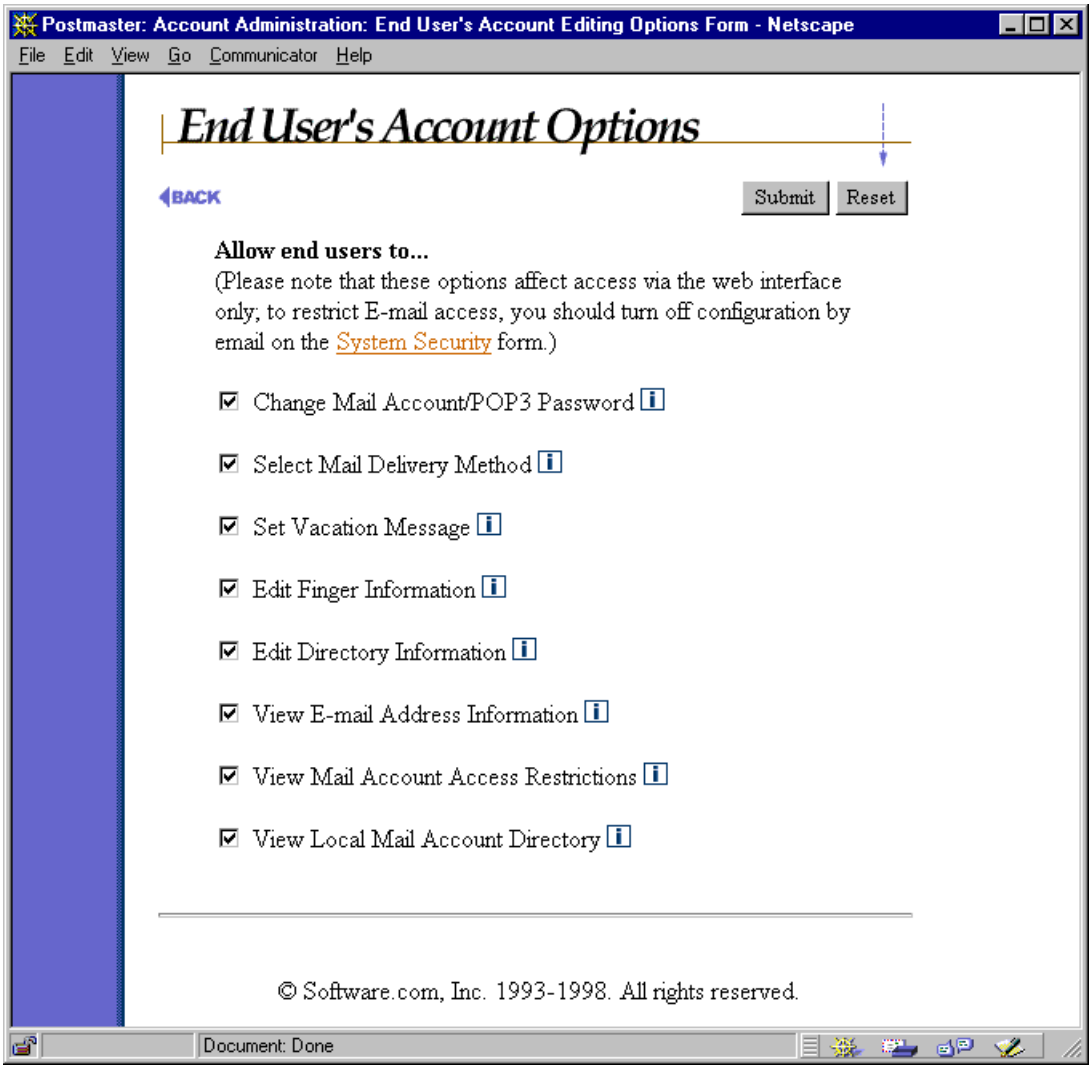

Figure 4-11 End User Account Options Form

Each of the toggle buttons on this form corresponds to a menu entry on the end user's Account Management Menu, and a corresponding form in the end user web interface. When an end user option is disabled, the end user's menu simply doesn't show the link to the appropriate form, so access to that form – and the functionality it offers – is unavailable.

Note: *The options in this form are set globally, so all of your users will be restricted from whichever account options that you disable here. This includes you also – if you log into your personal Post.Office account, you will also be restricted from the hidden account options.*

The items available in the End User's Account Options Form are:

- **Change Mail Account/POP3 Password**. Controls access to the Mail Account Password Form
- **Select Mail Delivery Method**. Controls access to the Mail Delivery Method Form
- **Set Vacation Message**. Controls access to the Vacation Message Form
- **Edit Directory Information**. Controls access to the Directory Info Form
- **Edit Finger Information**. Controls access to the Finger Information Form
- **View E-mail Address Information**. Controls access to the E-mail Address Information Form
- **View Mail Account Access Restrictions**. Controls access to the Mail Account Access Form
- **View Mail Account Directory**. Controls access to the Mail Account Directory

The following is an illustration of the complete, unrestricted Account Management menu for end users:

Figure 4-12 End user's Account Management Menu ("before")

And now here's the same menu after the options for changing the password, setting mail delivery, and viewing the Mail Account Directory have been disabled by the Postmaster:

Figure 4-13 End user's Account Management Menu ("after")

Remember, setting end user account options is a global operation, so all users of the installation of Post.Office pictured above will be confined to only these three account management items.

4.9 Logging Options Form

The Logging Options Form allows you to request the activities within Post.Office that you want to be recorded in the daily system log file. Log files – as described in Chapter 8 – are useful for troubleshooting odd or unintended mail system behavior.

This form is invoked from the System Configuration menu by clicking on the **Set Logging Options** link, and looks like the following illustrations:

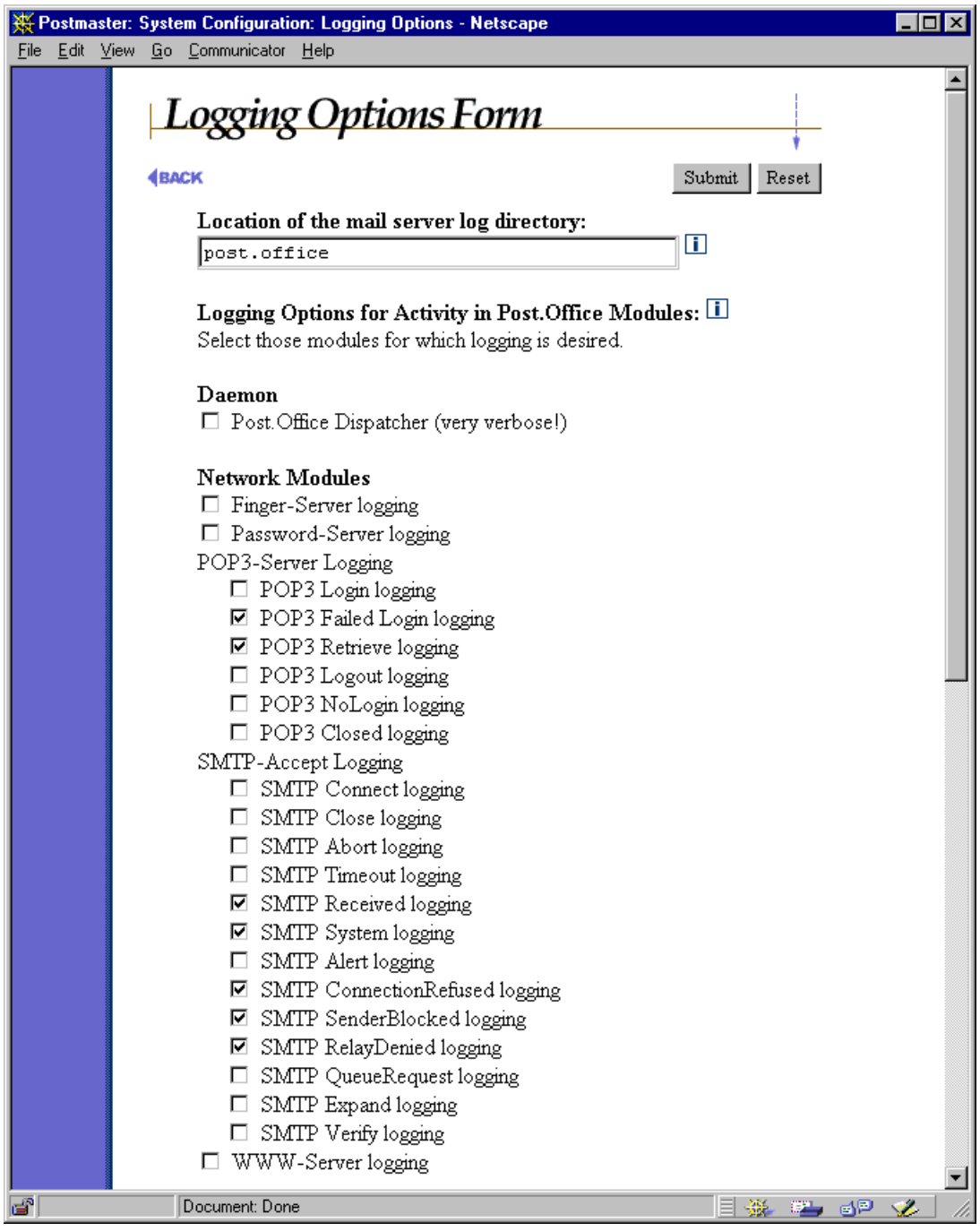

Figure 4-14 Logging Options Form (part 1 of 2)

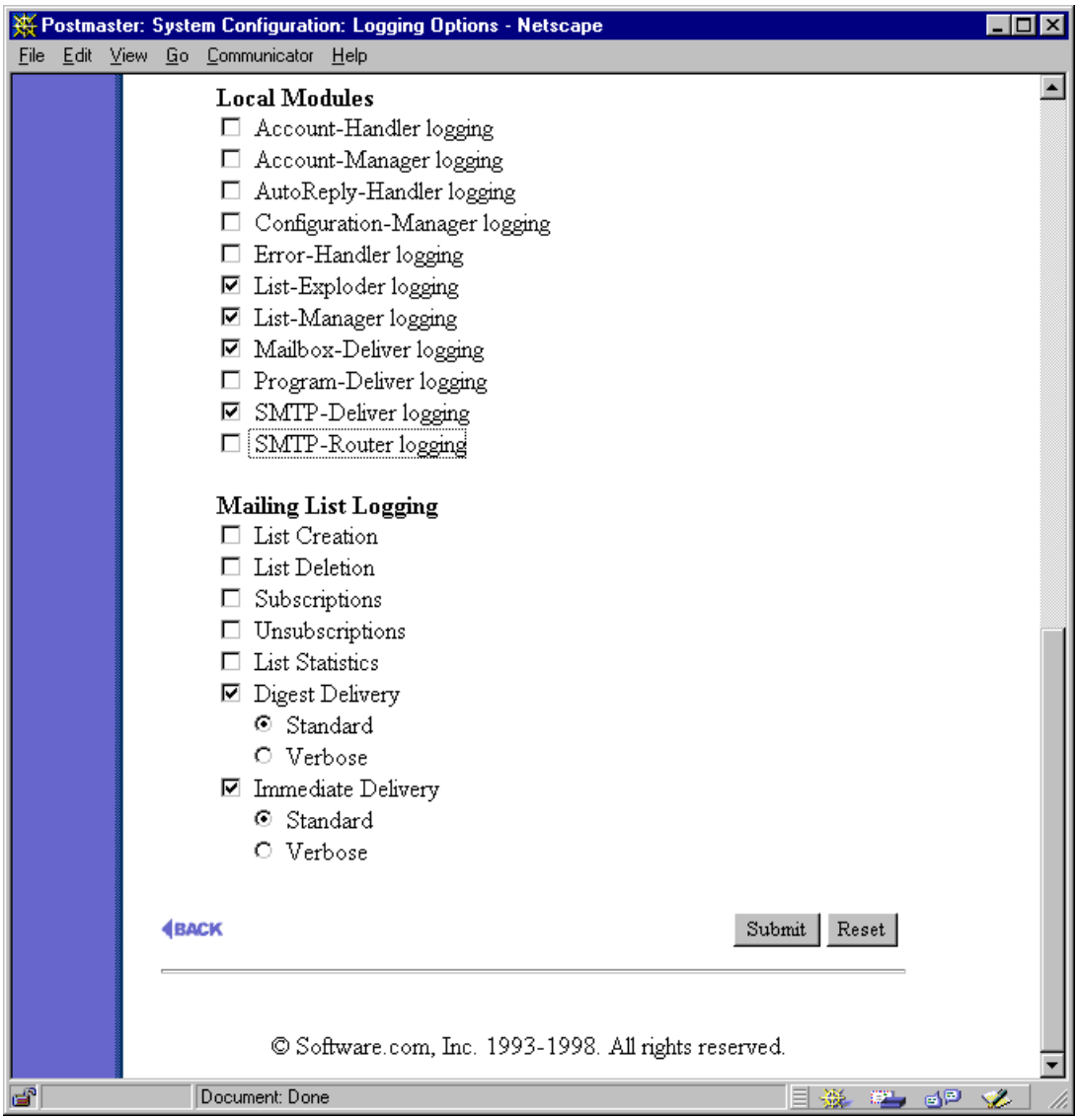

Figure 4-15 Logging Options Form (part 2 of 2)

Location of the mail server log directory

By default, all log files are kept in the `post.office/log` directory, but you can override this by specifying the full pathname of the directory where the logs should be stored. You do not need to specify the full path name to use the default – use the special keyword “post.office” to request the default directory location. Our recommendation is to leave this field as-is.

Warning! Be sure that the permissions of the log directory you choose allow Post.Office access to the log files. Since Post.Office runs as a non-privileged user (for enhanced system security), it may be unable to create log files that do not already exist.

Logging Options

All Post.Office log information for a specific day is kept in a single file, named in the format `post.Office-####.log` (where #### represents the month and day). The log files are automatically rotated on a daily basis. The location of the log file is under Postmaster control via the above directory location field.

Each module that writes an entry in the log file uses a format that is machine readable for automatic processing. Each entry consists of the current date and time, the module that recorded the information, and the module specific information. The date and time are relative to the local time zone in the format YYYYMMDDhhmmss (year, month, day, hour (00-23), minute, and second). The module specific information varies on a per module basis.

See Chapter 8 for information on reading Post.Office log files.

4.10 Error Response Parameters Form

The Error Response Parameters Form defines rules for the handling of undeliverable messages. You can choose to set the system on auto-pilot and just have it bounce (that is, return to sender) all undeliverable or otherwise problematic messages, or you can choose to send all such message to the Postmaster (you) for manual processing.

This form is invoked from the System Configuration menu by clicking on the **Establish Error Response Parameters** link. A similar link on the Status of Deferred Mail menu invokes this same form.

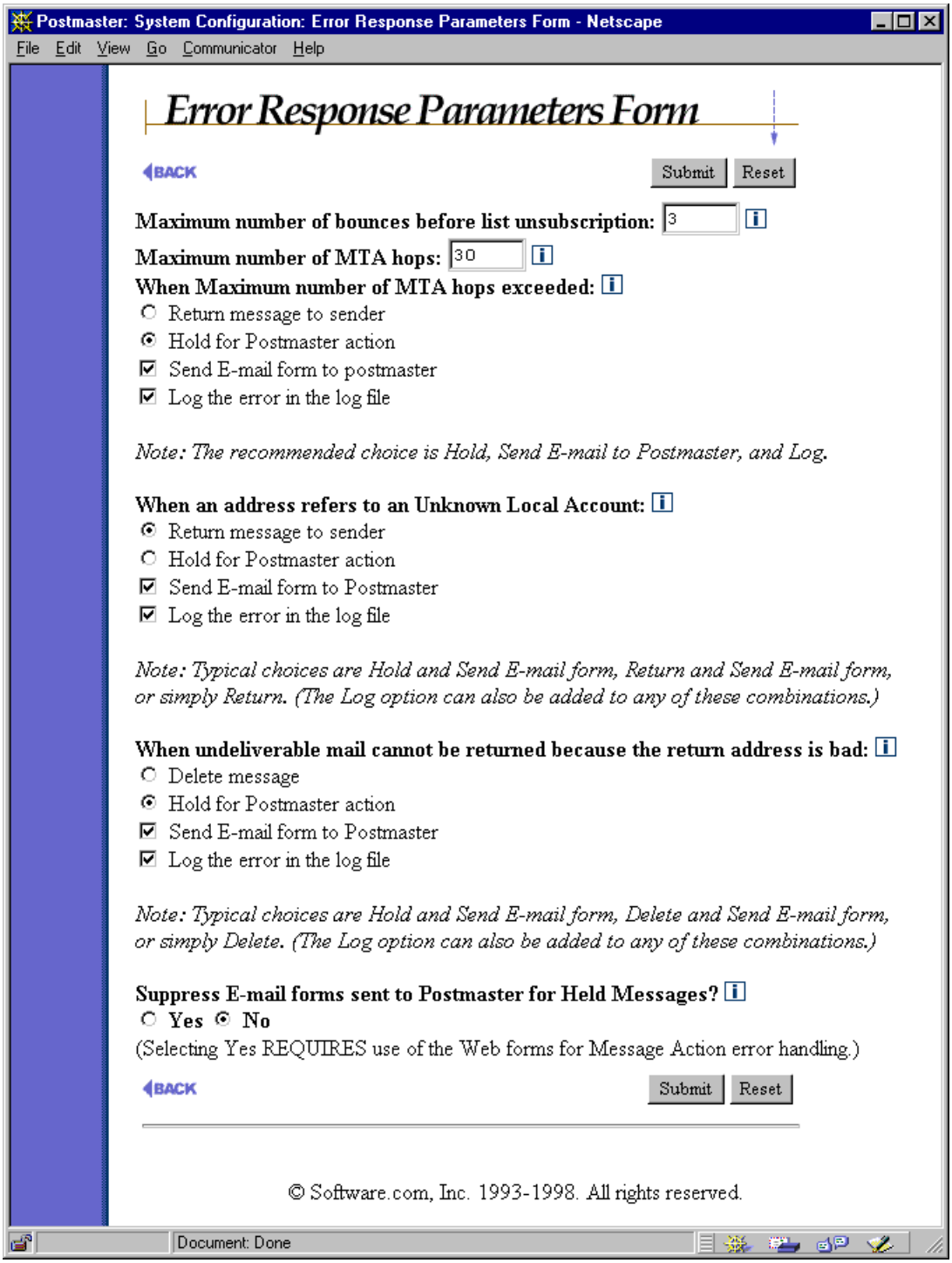

Figure 4-16 Error Response Parameters Form

Maximum number of bounces before list unsubscription This option is used to keep the membership of Post.Office mailing lists up-to-date. Because the computer users of the world tend to change their e-mail addresses quite frequently, subscriber lists often include addresses that are no longer valid. Having obsolete e-mail addresses in a subscriber list is undesirable, because they cause Post.Office to waste processing time sending list postings to the non-existent subscribers, and handle resulting bounce messages from other mail servers.

When a mailing list posting causes a bounce message from a remote mail server, Post.Office counts this bounce against the unreachable subscriber. When the number of bounces recorded against a subscriber reaches the number specified here, the subscriber will be dropped from all mailing lists on your system.

Note: *This field specifies the number of* cumulative *bounces that cause an address to be dropped from mailing lists, not a* consecutive *count. The bounce count against a subscriber is not reset to zero if the account is reachable during subsequent mailing list distributions.*

Maximum MTA Hops. This parameter is used to prevent mail loops. Usually a mail loop is caused by having two e-mail accounts on different machines that forward mail to each other. Any mail sent to either of these accounts could bounce back and forth between the machines forever. Fortunately, these loops can be detected and stopped since each mail server stamps all incoming messages as Received. By counting the number of Received lines in the message header, Post.Office knows how many hops the message took to get here. The recommended value for this parameter is 30. If the number of hops of an incoming message exceeds the maximum specified in this parameter, an error results and will be handled according to the **Max Hops Exceeded** field below.

When Maximum number of MTA hops exceeded. When a message arrives whose number of MTA hops exceeds the limit defined in the above field, the error will be handled as you request here. Accounts that cause mail loops need to be corrected as soon as possible; hence, the strongly recommended combination of actions for this error is to hold the message and notify the Postmaster so that the responsible account can be corrected.

Warning! Because a message which exceeds the maximum number of MTA hops is probably caught in a mail loop, it is highly recommended that you do not select the **Return message to sender** option, since this will almost certainly continue the mail loop.

When an address refers to an Unknown Local Account. This error means that a message addressed to one of the local domains was received with an address that did not correspond to an account, mailing list, or channel alias on the system. Often this is caused by the sender mis-typing the address. It can also mean that an account or alias should be added to the system.

The recommended response to this condition is to hold the message (**Hold for Postmaster action**) and notify the Postmaster (**Send e-mail form to postmaster**). This combination is especially important when upgrading from a different mail system or redistributing users among various hosts. The Postmaster is notified via e-mail of each error by the arrival of a Message Action Form, and can submit this e-mail form to process the undeliverable message.

Because held messages can also be handled via the web interface, you may choose to hold the message, but not notify the Postmaster via e-mail. You should select this combination only if you intend to handle all Unknown User errors exclusively through the web interface.

If you do not wish to handle mail addressed to Unknown Users you should select the Return option. Once the mail system is up and running smoothly, it is common to then return unknown user mail (with or without Postmaster notification).

When undeliverable mail cannot be returned because the return address is bad. This error means that Post.Office attempted to return a message to its sender (for example, in the case of a message addressed to an unknown local user), but could not because the return address of the original message does not exist. Often this is because the sender has configured their e-mail client incorrectly; in other cases -- such as with distributors of "junk" e-mail – it is because the sender does not want to receive direct responses to their message.

Because this error condition involves a message that could be neither delivered nor returned, the recommended response is to **Delete** the message, **Send** a notification to the Postmaster, and **Log** the error.

Suppress e-mail forms sent to Postmaster for Held Messages? By default, Postmasters receive notification via e-mail when errors occur that require Postmaster action. Setting this field to **Yes** will suppress delivery of those messages and require all error handling to be accomplished via web forms. If you wish to maintain the option of handling errors via e-mail or via the web you should select **No**.

4.11 System Security Form

The System Security Form allows you to set certain system wide security options. Among these security items are options for restricting access to Postmaster operations in both the web and e-mail interfaces.

This form is invoked from the System Configuration menu by clicking on the **Establish System Security** link, and looks like the following illustration:

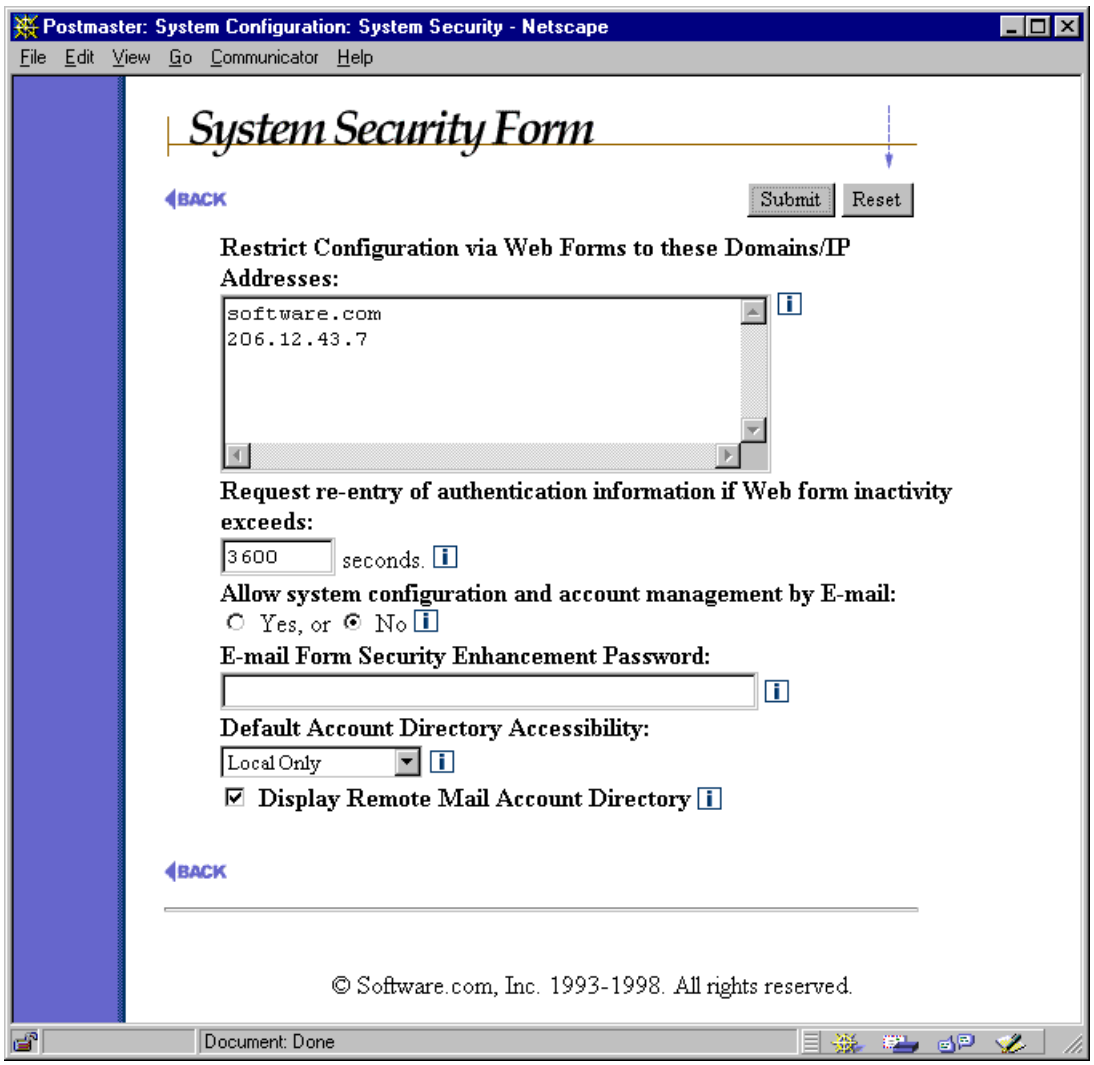

Figure 4-17 System Security Form

Restrict Configuration via Web Forms to these Domains/IP Addresses. This option allows you to set access restrictions for all web configuration of Post.Office on this host. These restrictions are specified as domain names and/or IP addresses, one per line in this field. You probably want to ensure that users who have no business poking around your e-mail can't access your mail server, so be sure to enter hostnames, domain names, and/or IP addresses that provide access to only a few computers.

The following algorithm used to determine if a connecting client has permission to access the Post.Office web interface based on the information entered in this field:

1. If the list is empty, access is allowed.
2. If the keyword "none" appears in the list, access is denied.
3. If the client's machine name is within one of the named domains, access is allowed.
4. If the client's IP address is within one of the listed networks, access is allowed.
5. In all other cases, access is denied.

For example, suppose configuration is limited to the following:

```
software.com
```

Postmasters and users alike can only make configuration changes from a machine in the `software.com` domain, such as `sparky.software.com`, even if the Access Restrictions for their accounts are less restrictive and they know the correct password.

Request re-entry of authentication information if Web form inactivity exceeds ___ Seconds. After a certain period of inactivity, the web server automatically closes the session as a safeguard. This protects you from absent-mindedly logging in to Post.Office and then leaving work for the day with your web browser still running, which would otherwise allow anyone to sit down at your computer and begin impersonating your Postmaster persona. This field allows you top specify your preference for the length of this timeout period. A good setting is 600 seconds (10 minutes).

Allow system configuration and account management by e-mail. If you do all your configuration through the web interface, you might as well turn this off by selecting **No**. Otherwise, set this field to **Yes** to allow yourself the option of using the e-mail interface. Turning off e-mail configuration represents another level of security.

E-mail Form Security Enhancement Password. Use this field to set the encryption key used to make the Form Identifier (at the bottom of all e-mail forms). This is especially useful if you need the ability to transfer mail accounts between different machines. Two machines with the same E-mail Form Security Password will be able to submit forms from one machine to the other most easily.

Default Account Directory Accessibility. This field sets a global default for the visibility of accounts in the Mail Account Directory. This directory displays the Real Name and Primary E-mail Address of each listed account, with optional links to user home pages. Directory accessibility can be controlled globally with this option, or can be set on an account-by-account basis. When the Directory Accessibility of an account is Default, the account will have whatever listing status that you define here.

There are three selections for this field:

- **Local Only** specifies that accounts are visible only in the Mail Account Directory accessible to local users (that is, users who have Post.Office accounts on your system).
- **Local and Remote** specifies that accounts are visible in the Mail Account Directory for both local and remote users. This means that users from outside of your system will be able to view the Real Name and Primary E-mail Address of the accounts in the public Mail Account Directory.
- **Unlisted** specifies that accounts should not be listed in the Mail Account Directory.

Display Remote Mail Account Directory. This option controls whether or not a Mail Account Directory will be available to remote users. When this option is enabled, a **Mail Account Directory** menu button will appear on the Authentication Information form; by clicking on this button, users can view a listing of accounts that have been specified as visible here (see the available account listing types above shown above). Disabling this option removes this menu button from the Authentication Information form, thereby disabling the public Mail Account Directory.

4.12 UNIX Delivery Configuration Options Form

On UNIX platforms, one of the available configuration forms is the UNIX Delivery Configuration Form, which allows you to set system-wide options for UNIX mail delivery methods. If any of your users employ the UNIX delivery feature for their accounts, you will need to set the required options in this form before they can receive their mail via UNIX delivery.

The UNIX Delivery Configuration Form is displayed from the System Configuration menu by clicking on the **Set Special Delivery Configuration for UNIX** link, and looks like this:

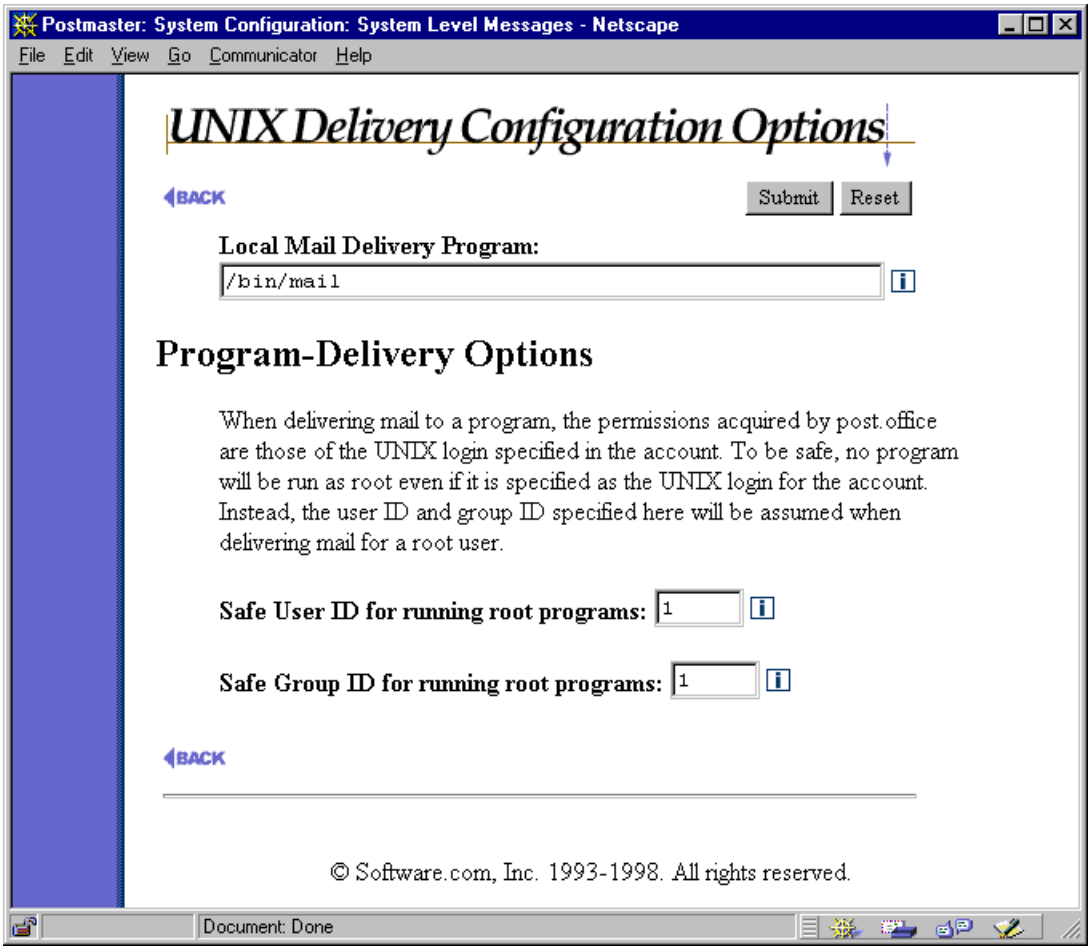

Figure 4-18 UNIX Delivery Configuration Options Form

Local Mail Delivery Program. For accounts that have UNIX Delivery enabled, an external program is run by Post.Office to perform the final delivery of messages that are received for these accounts. The final delivery program typically places the incoming messages in a file (commonly called a maildrop file) such as `/var/mail` or `/var/spool/mail`.

The delivery program is run with command-line arguments that indicate who the sender of the message is (the `-f` argument), and the UNIX Login Name of the recipient (the `-d` argument), as in the example:

```
/usr/bin/mail -f george@xyz.org -d root
```

Program Delivery Options

The fields in this section of the form pertain to the Program Delivery method for delivering messages, which is described in Chapter 6. When invoking programs for Program Delivery, Post.Office runs the program with the permissions of the user specified in the UNIX Login Name field of the account. However, due to security concerns, Post.Office will never run any program as `root`; so if this is the specified UNIX login name, Post.Office will instead use the **Safe User ID** and **Safe Group ID** specified here.

4.13 System-Level Default Messages Form

The System-Level Default Messages Form provides various default finger and auto-reply messages for Post.Office to use in a pinch. It provides information used to answer finger queries which include the name of the host system but no username, and provides auto-reply messages for accounts that are using this feature but which have no auto-reply message.

Admittedly, providing a default auto-reply message isn't the most useful thing in the world, since the point of such messages is that they provide some type of account-specific information. However, this option is here for you if you decide that having defaults for such values makes sense for your organization.

The System Level Default Messages Form is displayed from the System Configuration menu by clicking on the **Edit System-Level Messages** link, and looks like this:

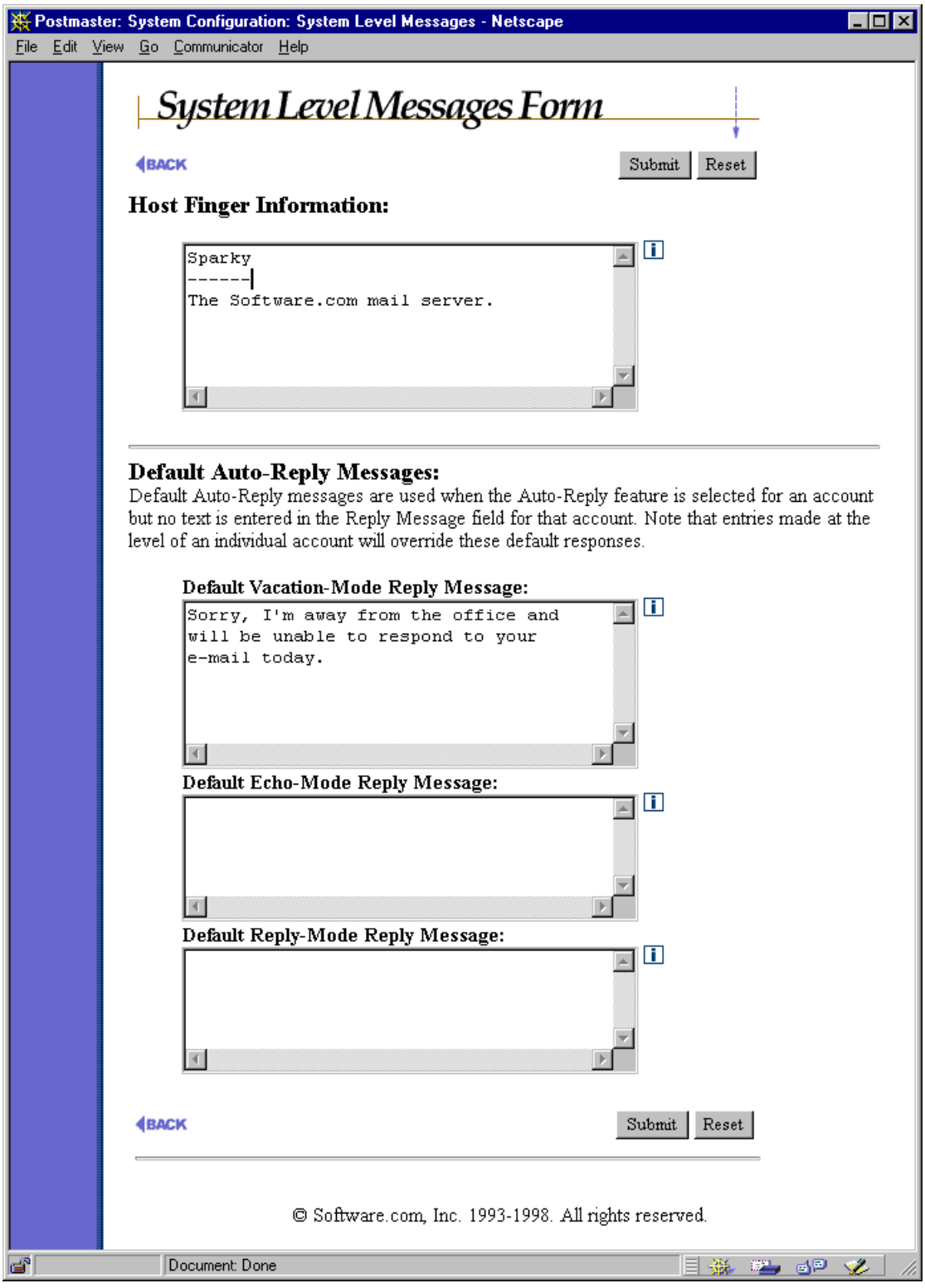

Figure 4-19 System-Level Messages Form

Host Finger Info: This is the response provided to queries that do not include a user name and are therefore addressed more generally to your company or organization. This is a good place to put miscellaneous information about your company and maybe a couple of contact names and e-mail addresses. However, this information probably won't get seen by many users.

For example, if you finger "software.com" without specifying a host-name or someone's address you'll get a couple e-mail addresses that you can use to contact us as well as our postal address and phone number.

Default Vacation Message. Hopefully your users will remember to make their own vacation messages before they take off for Las Vegas or Graceland. However, users may enable the vacation message feature for their account without having actually written one; in this case, the vacation message specified here is provided for them. This is a good place to leave a fairly generalized message saying something along the lines of "The person you tried to contact is on vacation..."

Default Echo Message. Like the default vacation message described above, this message is sent on behalf of accounts which have the auto-reply feature enabled in **Echo** mode, but which have no auto-reply message.

Default Reply Message. Like the default vacation message, this message is sent on behalf of accounts which have the auto-reply feature enabled in **Reply** mode, but which have no auto-reply message.

Hint: *We recommend that you just leave the three default message fields blank. That way, if an account is using the auto-reply feature but has no auto-reply message, the Postmaster will be notified so that he/she and correct the problem.*

4.14 Licensing/Configuration Information Form

The Licensing/Configuration Information Form is a read-only form that provides a variety of information about your particular installation of Post.Office. This information can be very useful for troubleshooting and system administration.

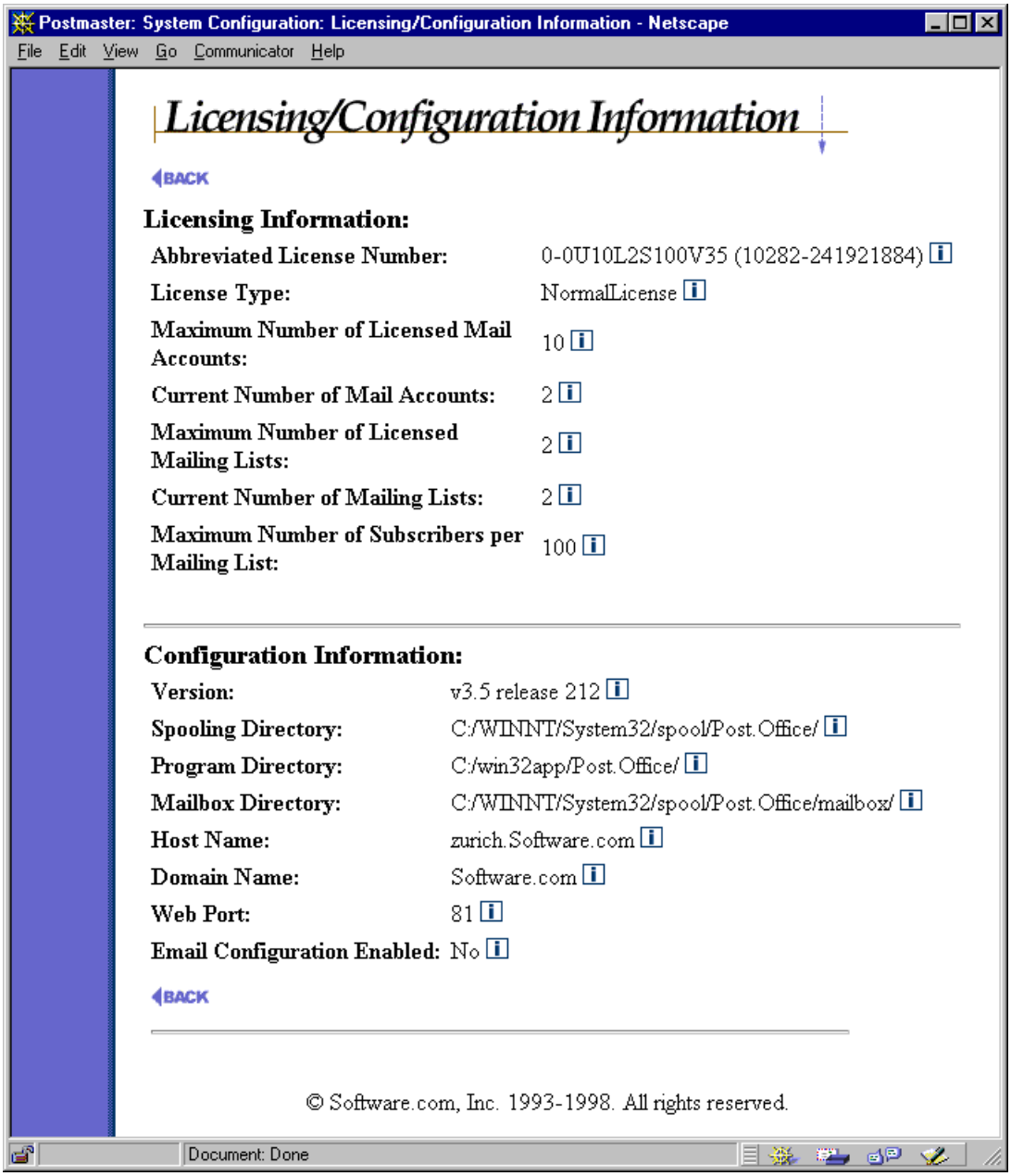

Figure 4-20 Licensing/Configuration Information Form

Licensing Information

This part of the Licensing/Configuration Form contains information regarding the licensing of your particular Post.Office installation.

- **Abbreviated License Number**. This information provides easy identification of your Post.Office package. The complete license number (which should have been received upon product purchase) is only required when installing or upgrading the Post.Office software.
- **License Type**. This value indicates the type of license associated with your Post.Office installation (Standard, Trial, etc.) and may change upon installation of a new license number.
- **Maximum Number of Licensed Mail Accounts**. This value indicates the maximum number of mail accounts licensed for use on this server. Please contact your Post.Office vendor if you wish to purchase a license for additional mail accounts
- **Current Number of Mail Accounts**. As the name suggests, the current number of mail accounts reflects the total number of accounts currently established on this mail server (excluding those reserved accounts created at time of installation which are required for the operation of Post.Office). Licensing restrictions dictate that the number of mail accounts may not exceed the maximum number of Licensed Mail Accounts. Please contact your Post.Office vendor if you wish to purchase a license to accommodate additional mail accounts.
- **Maximum Number of Licensed Mailing Lists**. This value indicates the maximum number of mailing lists licensed for use on this server. Please contact your Post.Office vendor if you wish to purchase a license for additional mailing lists.
- **Current Number of Mailing Lists**. As the name suggests, the current number of mailing lists reflects the total number of mailing lists currently established on this mail server. Licensing restrictions dictate that the number of mailing lists may not exceed the Maximum Number of Licensed Mailing Lists. Please contact your Post.Office vendor if you wish to purchase a license to accommodate additional mailing lists.

Hint: *We recommend that you periodically look at the above limits so that you know when your mail system is approaching them.*

- **Maximum Number of Subscribers per Mailing List**. This value indicates the maximum number of subscribers allowed for any mailing list on this mail server. The limit is controlled via license. Please contact your Post.Office vendor if you wish to purchase a license to accommodate a larger limit. The Maximum Number of Subscribers per Mailing List is a system level default. More restrictive limits may be set on a list by list basis via the Maximum Number of Subscribers field found in the Mailing List Data Form.

Configuration Information

This part of the Licensing/Configuration Form contains information that is useful during troubleshooting or when you need more information about the server system where Post.Office is running.

- **Version**. This value indicates the version and build of the Post.Office software currently installed on your mail server. Please contact your Post.Office vendor to purchase an updated version of the Post.Office software or to inquire about the latest version number.
- **Spooling Directory**. This value indicates a path to the Post.Office Spooling Directory. Most discussions in the Post.Office manual assume that the Spooling Directory was installed in the default location. If the entry here indicates an alternate location you will need to adjust those examples to suit your particular configuration.
- **Program Directory**. This value indicates the directory on the server file system where the Post.Office executables are stored.
- **Mailbox Directory**. This value indicates a path to the directory where the Post.Office mailboxes are stored.

Hint: *The locations of the three above directories are important for you to know when backing up your mail system.*

- **Host Name**. This value indicates the host name specified during installation of the Post.Office software.
- **Domain Name**. This value indicates the domain name specified during installation of the Post.Office software.
- **Web Port**. This value identifies the server port currently used by the Post.Office WWW server for handling of the Post.Office web forms. This port number is initially specified during installation of the Post.Office software.
- **E-mail Configuration Enabled**. This value indicates whether or not your system is currently set to allow configuration changes of Post.Office using the e-mail forms. If e-mail configuration is disabled, mail sent to the Configuration Manager account (`configuration@host.domain`) or Account Manager account (`accounts@host.domain`) will be returned to the sender with an appropriate error message. This option can be enabled or disabled in the System Configuration Form.

5

Account Management

This chapter discusses e-mail accounts controlled by Post.Office, and includes the following topics:

- An introduction to account attributes and the different kinds of Post.Office accounts
- An overview of the account management menus
- Instructions for creating, modifying, and deleting accounts via the web interface
- Instructions for managing the special Postmaster administrative account
- Information on the Mail Account Directory

5.1 What Is an Account?

Information about the users who receive e-mail on your system is organized by Post.Office into accounts. An e-mail account is the electronic equivalent of a P.O. box. The information included in an e-mail account identifies the recipient by name and address, determines how messages are accepted and delivered for that account, and defines the directory information that is provided.

Accounts contain, among other information, the name of the user, their e-mail address or addresses, how and where they receive their e-mail, what their password is, and what their directory information is. The sum of the information in all accounts is held in an internal Post.Office account database. There are eight principle kinds of information for each account in the database:

- **General Account Information** includes the name of the account and the associated password (the password is stored in an encrypted format).
- **E-mail Addressing Information** is specific to the SMTP channel (used for Internet mail), and includes the e-mail address or addresses for that account. This identifies the addresses for which mail will be received.
- **Mailing List Subscription Information** defines the mailing lists to which the account is subscribed. The account will receive mail from these mailing lists.
- **Local Delivery Information** specifies how messages for an account are delivered to its recipient. They can be delivered via POP3, placed in a UNIX maildrop file (UNIX platforms only), forwarded to another address, or delivered to a program.
- **Mailbox Size Information** is provided for users with POP3 mailboxes. Both current mailbox size and the maximum limit (if any) are noted.

- **Account Security Parameters** determine the access restrictions for the account.
- **Automatic Reply Information** is optional, and if provided, enables an automatic response to mail sent to the account.
- **Directory Information** is optional. If completed, it provides information displayed in the Mail Account Directory.
- **Finger Information** is optional. If completed, it provides finger directory information.

While most of the account information is controlled solely by the Postmaster, certain items which apply only to the account – namely the password, delivery information, vacation message, and finger information – can be changed by the account's owner. This allows individual users to set a few account options without bothering the Postmaster, but still prevents them from making any changes that could potentially compromise system security.

5.1.1 Types of Accounts

Although all accounts in Post.Office include the attributes described above, there are three different main types of Post.Office accounts: Administrative accounts, general accounts, and reserved accounts. Each of these groups have sub-groups that further divide accounts into special categories.

Administrative Accounts

Two types of accounts fall into the category of administrative accounts: the Postmaster account itself, and the general accounts of all users who have been given Postmaster privileges (i.e., you). These accounts are displayed apart from the general user accounts to signify their importance.

Postmaster Account. The Postmaster account represents a function, not a person, but the account itself can receive e-mail just like any other account. All messages sent to the Postmaster account are forwarded all of the users who have been given Postmaster privileges. In this way, the Postmaster account is technically a group account (described later in this section).

The Postmaster password – that is, the password defined for the Postmaster account – is required for carrying out all configuration and account management. It is something you don't want to share, since it gives you access to every detail of the Post.Office system.

The Postmaster account is created on installation and cannot be deleted from the accounts database. The general account of the initial Postmaster is also created during installation, but this account can be later deleted or reduced to the role of non-Postmaster (provided at least one other general account has been given Postmaster privileges).

Designated Postmasters. These are the individual users who have been granted Postmaster privileges (that is, people like you). These are the people who are sent (and can respond to) error notifications and other system maintenance tasks. A user need not have a mail account in Post.Office to be a designated Postmaster, so just about anybody with an e-mail address can be your mail administrator.[27] You can also have any number of designated Postmasters if you decide to share this solemn duty with other users.

Reserved

In addition to the Postmaster account, several other reserved accounts exist in Post.Office. These accounts have special features necessary for the operation of the program, and are used frequently, but not necessarily directly. Most of these accounts are used only if you decide to operate the system via the e-mail interface (an old Post.Office feature which remains supported for backward compatibility), but you should still be familiar with these accounts.

Reserved accounts include the following:

- **Default Account**. This is the account that holds default information used in the creation of all new accounts. This isn't technically an account, since mail cannot be addressed to it, but it exists in the account database and contains all of the attributes of an account. All of the information included in this account is inserted into the New Account Data Form whenever you create a new account.
- **Account Manager**. This account sends and receives e-mail forms for transactions involving account management, and has the default address `accounts@[IP.address]`.
- **Configuration Manager**. This account sends and receives e-mail forms for transactions involving Post.Office system settings, and has the default address `configuration@[IP.address]`.
- **Error Handler**. This account sends and receives e-mail forms regarding the handling of undeliverable or unreturnable mail, and has the default address `error-handler@[IP.address]`. The e-mail interface for error-handling is described in Chapter 8.
- **List Manager**. This account sends and receives messages related to mailing lists, and has the default address `list-manager@host.domain`. The e-mail interface to mailing list functions is described in Chapter 7.
- **All Mailboxes**. This reserved account is actually a reserved mailing list, which can be used to broadcast messages to all local mail accounts that use the POP3 method of delivery. The default address for this mailing list is `all-mailboxes@host.domain`.

---

27 This is why it's very important to set the security parameter which limits the hosts and/or domains from which Postmaster modification can be carried out. When this access is set accordingly, unauthorized users will be unable to carry our Postmaster tasks, even if they know the Postmaster password.

You can change the e-mail address of these accounts in the event it interferes with your system. Otherwise it is a good idea to leave them the way they are. For example, "list.manager" is the primary address for the List-Manager, but you may already be using the address "list.manager" for something else. Or maybe you want to add an extra address to these accounts so that they are easier for you to remember (or even just quicker to type). When customizing your system this way, it is strongly recommended that you insert additional addresses rather than to remove the primary default values.

General

The majority of Post.Office accounts – that is, all accounts that are not administrative or reserved accounts – fall into the category of general accounts. Unlike reserved accounts, general accounts are not created automatically; these are the e-mail accounts that you create, typically for users who will receive their e-mail through Post.Office. General accounts are the only type of account that can be created or deleted.

Within the category of general accounts, there are four basic conceptual types:

- **Individual**. Individual general accounts are by far the most common type of Post.Office account, and typically correspond to an individual computer user (for example, `john.doe@software.com`)who receives, forwards, or stores e-mail in Post.Office.
- **Group**. A group account is an account which forwards incoming messages (addressed to the group account) to other accounts. As opposed to individual accounts that typically forward mail in addition to storing messages for user collection, group accounts simply pass messages on a group of users. Group accounts are simple mailing lists, and typically correspond to a group of computer users. For example, you might create a group account such as `social.committee@software.com` which forwards all messages to the members of this committee.

Note: *Because mailing lists offer the same functionality as group accounts, but with far more flexibility and features, you probably won't use group accounts too much. Group accounts have been used as mailing lists in previous versions of Post.Office that did not include the mailing list manager. Still, group accounts have their place and can be quite useful. See Chapter 7 for a discussion on group accounts vs. mailing lists.*

- **Auto-Reply**. All accounts can include an auto-reply message, which is sent to all users who address messages to that account. However, only the Postmaster can create an auto-reply account, which uses the auto-reply facility to respond to all messages but which has no associated delivery method (incoming mail is simply deleted). This type of account is useful for distributing information that does not require a personal response, such as a price list, sales brochure, order form, directions to your office, etc.

- **Wildcard**. Among the many mail routing options provided by Post.Office is the ability to deliver to a single account any message that is addressed to a particular local mail domain. This allows mail to be delivered to your site, even if it's addressed to an unknown address in the domain. The account that receives all mail for unknown users in a particular domain is known as a wildcard account. (See Section 5.3.2 for information on setting wildcard delivery for an account).

5.1.2 How Account Information Is Used

The illustrations below demonstrate the ways in which the information in an account ends up in actual e-mail (Figure 5-1) and in a reply to a finger query (Figure 5-2):

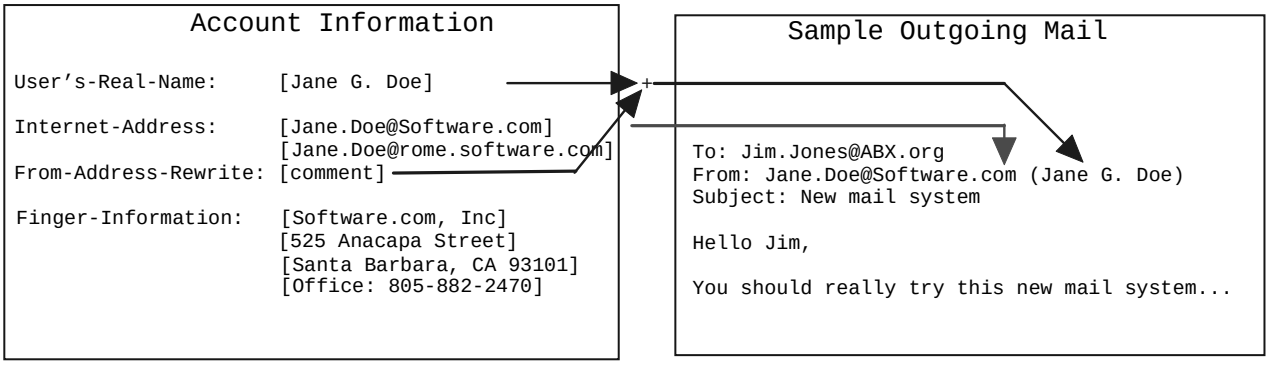

Figure 5-1 How account information is used in messages.

Exact usage may vary based on account options selected. We'll get into all of that later on in this chapter.

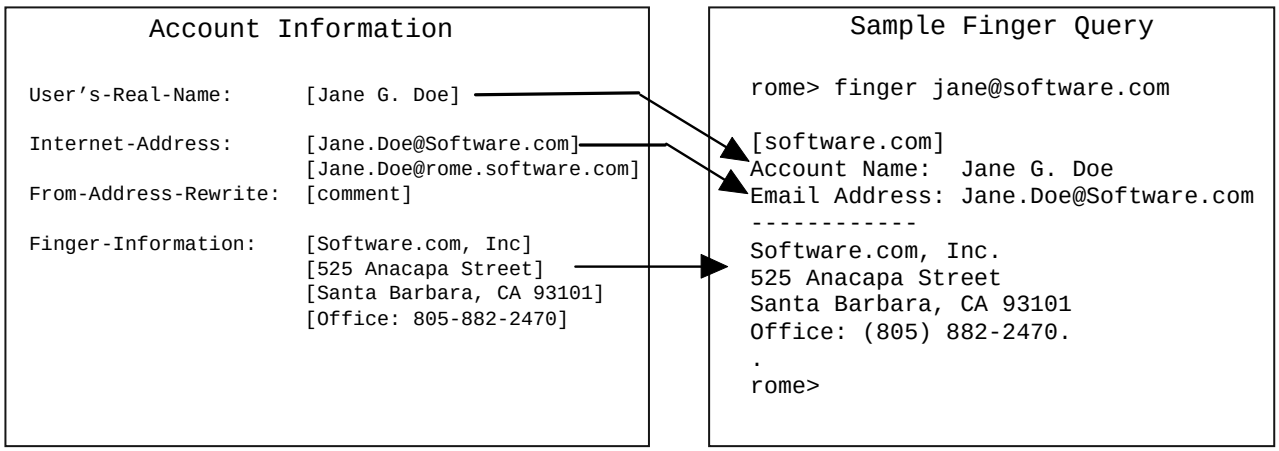

Figure 5-2 How account information is used in finger queries.

Finger queries elicit the user's name and official e-mail address, as well as their custom finger information. In this case, Jane Doe uses her finger information to make her mailing address and telephone number public.

5.1.3 Security Features of Accounts

Security of Post.Office accounts is enforced by careful use of passwords and access restrictions. These concepts are discussed briefly in the following sections.

Passwords

Passwords, when kept private and changed from time to time, act as electronic keys:

- Users need their password in order to retrieve messages when they are using POP3 delivery. As long as the password is secret, nobody else can retrieve their e-mail.
- Users must supply their password when logging in to the web interface.
- The Postmaster's password is required for all system configuration changes or to submit e-mail forms.

Post.Office passwords are case-senSiTive and must be at least six characters in length.

Note: *Be cautious about sending an unencrypted password across a public network (such as outside of your organization and across the Internet). If passwords are sent "in the clear" over such a network you may want to use the general access restriction feature for added security (see below).*

Account Security Parameters

Account Security Parameters limit the locations from which a user can obtain access to their account. These access restrictions provide an additional layer of security for all accounts, so even with the correct password, an account is inaccessible except to the domains or hosts that have specifically been granted access. You can use this option to prevent all users from outside of your network from accessing your mail accounts.

For example, when a user attempts to retrieve his e-mail with a mail client, the host name or IP address of the computer he's using is checked against the valid host names/IP addresses for his account; if the computer doesn't meet the access restrictions for his account, mail delivery will be denied. Similarly, when the user attempts to access his account via the web interface, the computer on which his web browser is running is checked against these host names and IP addresses, and will be denied if it fails to meet this criteria.

Access can be limited to a single computer or a set of computers in an addressing hierarchy. A single computer can be specified either by giving its fully-qualified domain name (for example, `sparky.sales.software.com`) or its IP address (for example, 10.2.111.30). Likewise, a set of computers can be specified by using an incomplete DNS address or IP address. An incomplete DNS address is one which does not specify a host (for example, `software.com`), while an incomplete IP address is one which contains a "0" (zero) in any of the four segments (the zero acts as a wildcard). The general access

restriction feature can be left blank to allow access from anywhere, or contain the keyword “none” to prevent any access at all to an account (except by the Postmaster).

The following algorithm used to determine if a connecting client has permission to access their account based on the information entered in this field:

1. If the list is empty, access is allowed.
2. If the keyword “none” appears in the list, access is denied.
3. If the client’s machine name is within one of the named domains, access is allowed.
4. If the client’s IP address is within one of the listed networks, access is allowed.
5. In all other cases, access is denied.

Use of domain names or IP addresses is a trade off between flexibility and security. Using a host or domain name is easily understandable and immune to network topology changes, while an IP address (or range of addresses) may not be. Generally speaking, IP addresses are safer than domain names for access restrictions, because they are more specific.

For maximum security, you can configure your access restriction to be the IP address of a single computer in your office. With this precaution in place, keeping the door to your office locked or otherwise restricting access to your computer will ensure that nobody can access your e-mail, even if they obtain your password. The use of IP addresses, however, does not require the presence of reverse-lookup records in the DNS.

For example, an access restriction might be set up as follows:

```
sparky.software.com
math.ucsb.edu
128.123.45.0
```

The above restriction entries would allow you to access the account from any of the following computers:

```
sparky.software.com
complex.math.ucsb.edu
128.123.45.22
128.123.45.67
```

However, you would *not* be able to access the account from the following:

```
fido.software.com
laser.ece.ucsb.edu
128.123.46.22
128.124.45.67
```


Finger-Access Restrictions

The finger access restriction feature limits the domains that have access to an account's finger information. If access is not allowed, no information is returned for the request.

For example, you may want to restrict your finger access to your company's domain. This way, only people within your organization would have access to the directory information in your account. You could then record sensitive information (like home phone numbers) in the finger information for each account, and run the finger server while being certain that nobody outside your company can access that information.

Note: *The same rules for specifying Account Access Security Parameters described in the previous section apply to finger access restrictions.*

5.2 The Account Administration Menus

To access the Postmaster's web-based account management interface, log in to the web interface as the Postmaster and with the Postmaster password (refer to Chapter 3 if you're not sure how to do this). After your login information is confirmed, you be taken immediately to the Account Administration menu, which you saw way back in Chapter 3. You can also get to this menu from any other menu by clicking the **Account Admin** menu button. To refresh your memory, here's what the menu looks like:

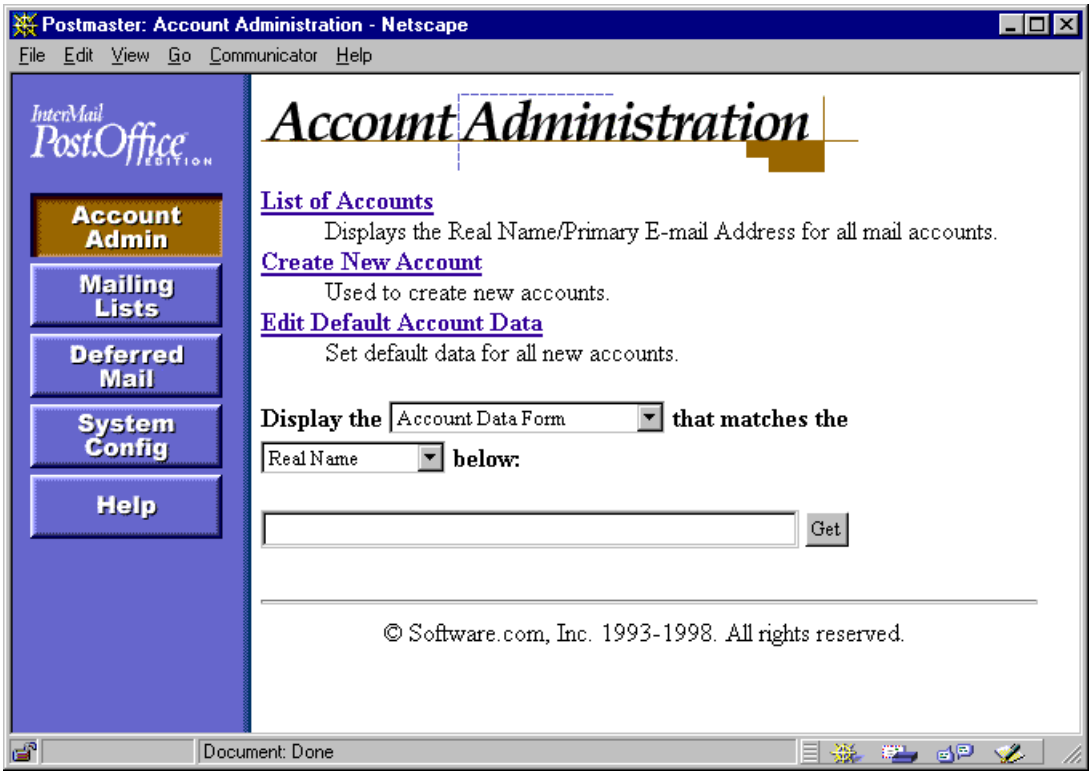

Figure 5-3: Account Administration menu

The Account Administration menu contains three links, as well as a text field and execution button. These links and the forms that they invoke are described throughout this chapter. For now, the only option on this menu that we'll look at is the **List of Accounts** link. This link invokes a menu that displays the list of all accounts on your Post.Office server (including administrative and reserved accounts).

Figure 5-4: List of Accounts menu (as seen when first displayed)

Three types of accounts can be viewed in this menu: **Administrative Accounts**, **General Accounts**, and **Reserved Accounts**. To view accounts, click on the ⊞ graphic next to the appropriate label; this expands the account menu to display the appropriate accounts.

Figure 5-5: List of Accounts menu (showing General Accounts)

Accounts are sorted alphabetically in this menu, by Real Name. By clicking on the name of a specific account in this menu, you can display a form for viewing and/or modifying the attributes of the account.

In the General Accounts sub-menu, **A-Z** (and other) links allow you to display specific subsets of the entire list of accounts. For each account, the user's Real Name and primary e-mail address are displayed. You can also display POP3 mailbox usage information for each account by clicking the **Show Quota** link.

Because finding a particular account in this menu may require several steps, the Account Administration menu includes a shortcut text field that allows you to bypass the List of Accounts menu and go straight to the desired account information. To use this shortcut field, enter the Real Name, e-mail address, or POP3 login name for the desired account in the shortcut text field. You can use a wildcard (*) character to request all accounts that match a particular pattern. Then select the information you want to access from the drop down menu (in the case of account settings, you would select **Account Data Form** from this menu), and click **Get**.

5.3 Creating Accounts

Now that you have some idea about what accounts are all about, you are ready to start creating new accounts yourself. This is the operation that you will perform the most often when working with accounts, since only the Postmaster can do so. Other operations specific to the account – such as setting a new password, changing mail delivery options, etc. – can be handled by whoever the account has been created for.

Accounts are created in the web interface with the New Account Data Form. This form can be invoked from the **Create New Account** link of the Account Administration menu, or the **New Account** link on the List of Accounts menu. Both of these links display the same form, so use whichever one is easier.

This form is reasonably long, so we'll take it one section at a time, with a series of screen shots and explanations of all account fields to assist you in using it.

Note: *The values initially displayed in the New Account Data Form are taken from the Default Account Data Form, discussed in Section 5.3.9.*

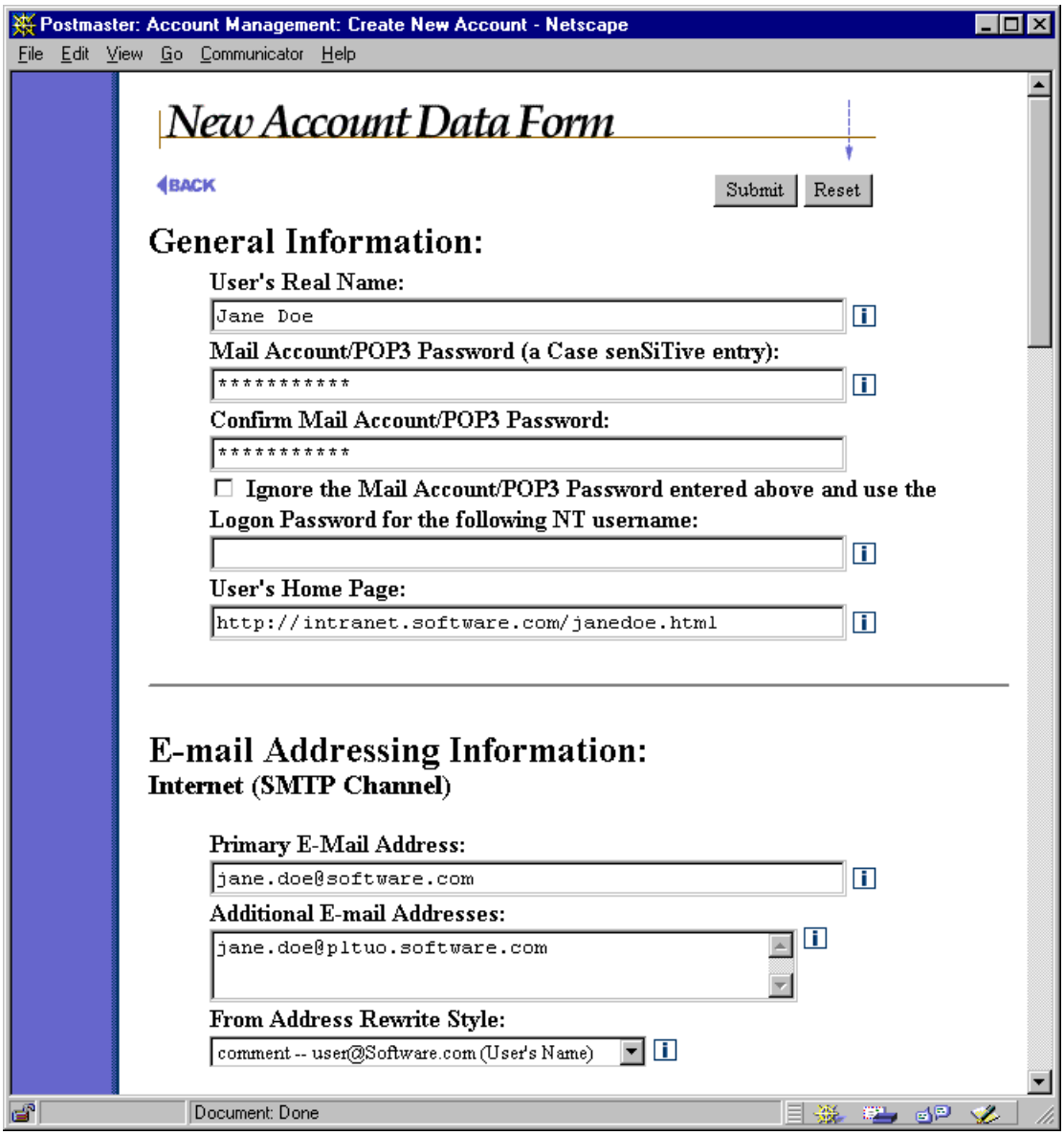

Figure 5-6: New Account Data Form (part 1 of 4)

5.3.1 General Information

The fields in this section of the New Account Data Form are used to define the name of the account's user, as well as account password information.

User's Real Name

This field contains the actual name of the account's user, or a descriptive account name in the case of auto-reply and group accounts. This descriptive name will be included with the e-mail address on messages sent out from this account, depending on the selected From Address Rewriting option (described in Section 5.3.1). The name does not have to be unique, so it is possible for two of your users have the same Real Name.

Accounts are displayed in the List of Accounts menu alphabetically by Real Name. This means that if you use the typical "First Last" name format (for example, "Jane Doe"), accounts will be displayed alphabetically by *first* name in this menu. You can have accounts instead sorted alphabetically by last name if you use the "Last, First" format ("Doe, Jane"), but this may look odd when used with From Address Rewriting (see below).

Mail Account/POP3 Password

This is the password that the user will use to access his account. This password is required in the Authentication Information Form when logging in to the Post.Office web interface. If the user gets his e-mail with POP3 delivery, he must also use this password with his mail client when checking for messages.

For security purposes, the mail account password should be something that is easy to remember but difficult to guess. For example, Jane Doe's password is `TenSany1?`, in honor of her favorite pastime. The use of special characters and capital letters makes passwords safer. However, don't make it *too* cryptic, or you may have so much trouble remembering it that you write it down on a piece of paper that you keep next to your computer (which hardly qualifies as secure!).

Note: *Don't forget to change the initial account password value, even if you plan to use the NT Integrated Passwords feature described below. The default password is* `Lock`*. An account which is given this address will be locked, as described in Section 5.4.3.*

NT Integrated Passwords (NT platforms only)

On NT platforms, the Postmaster has the option of allowing users to use their NT system logon password as their mail account/POP3 password, instead of the password defined in the Mail Account/POP3 Password field for their account. This allows users to have only one password, both for logging in to NT and collecting their mail. This password can be changed using standard operating system utilities.

Hint: Installations which take advantage of this feature frequently define the user's NT Logon Name to be the same as their Post.Office POP3 Logon Name.

This feature is activated on the Account Data Form by enabling the check box labeled **Ignore the Mail Account/POP3 Password entered above and use the Logon Password for the following NT username** and entering the appropriate NT user name. If the desired NT user is a domain user, you should enter `domain\username` (for example, `sales\joe`). If the user is a local user, you should enter the username only.

Warning! This feature requires that special permissions be assigned to the NT user responsible for running Post.Office (the one created during installation) and to those NT users who wish to use their NT logon password as their Post.Office password. Password integration will not work unless these required rights have been assigned (see instructions below).

In order for this feature to function as designed, the following additional conditions must be met:

1. The individual NT account that you entered on the Account Data Form (the one for the user whose e-mail account you're creating) must have "Log on locally" rights on the Post.Office host for verification to occur. You can assign a user this right through the NT User Manager.[28]
2. The Post.Office user (the NT user that was created at installation time and through which Post.Office runs) must also have special access rights. These rights can be set automatically at time of installation. If the person who installed Post.Office did not choose this option, you must add these options before using the integrated passwords feature. These include:

   - Act as part of the operating system
   - Increase quotas
   - Replace process level token

   Again, these options can be set in the NT User Manager. If you don't know how to set these, you can consult the Post.Office FAQ for line-by-line instructions for doing this.
3. Reboot your system after making the necessary changes. The new permissions for the Post.Office user will then take affect for your Post.Office MTA service.

Note: *Post.Office also allows access to the integrated NT passwords feature when using perl scripts. If you batch load your NT users with their NT logon names as their POP3 account names, you can also synchronize Post.Office passwords to be the same as their NT passwords. For additional information on this topic contact* `support@software.com` *or refer to the Post.Office FAQ.*

28 When assigning these access privileges, you must be logged into the NT server system as a local administrator. You cannot make the required changes if you log in as a domain administrator.

User's Home Page

This field defines an optional World Wide Web home page location for an account. A link to this home page is provided with the account's listing in the Mail Account Directory. When specifying a home page in this field, you must enter the full URL, including the protocol identifier (http, ftp, etc.). For example:

```
http://www.software.com/post.office
http://home.someisp.net/john_doe/home.html
```

Note: *Post.Office does not itself host web pages. This home page feature just allows users to include with their directory information a link to a web page hosted by another server*

5.3.2 E-mail Addressing Information

These fields are used to define e-mail addresses and address-related behavior for an account.

Primary E-mail Address

This is the "official" Internet e-mail address of the account. Although additional addresses specified in the field below are equally valid for the account, the primary address is the only one used with From Address Rewriting, shown in the List of Accounts menu, or returned to finger queries. This address, like all Post.Office addresses, must be in legal SMTP addressing format (i.e., `user@domain`), and must be unique throughout the system.

Note: *To set wildcard delivery for account, give it an address that includes a wildcard character ("*") followed by "@" and the local mail domain for which the wildcard account will accept mail. For example, an account with the address "*@software.com" will receive all messages sent to unknown addresses in the local mail domain software.com.*

Additional E-mail Addresses

These are additional e-mail addresses for the account. Mail sent to any of these addresses (or to the primary address) will be accepted by this account and delivered however the account's delivery options are defined. Again, these addresses must be in legal SMTP addressing format, and must be unique throughout the system.

Add as many entries as necessary to accommodate all the desired addresses for the account, remembering that no two accounts can have a matching Internet Address (regardless of whether the address is listed as primary or additional). Additional Internet addresses are useful when a user needs to be able to receive mail at several domains, if your preferred address format changes, or if the user has a commonly misspelled name.

For example, an account with the primary address `john.doe@software.com` might have the following additional addresses:

```
john.doe@sparky.software.com
jdoe@software.com
jon.dough@software.com
```

From Address Rewrite Style

This option modifies mail sent by the user to include the Primary E-mail Address in the **From:** header. This feature is especially desirable if you want to hide hostnames and subdomains from e-mail addresses, since many of your less experienced users may have a return address that includes this information in their mail client's **From:** address.

The available From Address Rewrite Options are **comment**, **quoted**, and **none**. The **quoted** option creates a **From:** address that includes the user's account Real Name (enclosed in "double quotes") followed by the account's primary address (enclosed in <angle brackets>). For example:

```
"Jane Doe" <Jane.Doe@Software.com>
```

The **comment** option creates a **From:** address that includes the primary address of the account, followed by the account Real Name enclosed in (parentheses). For example:

```
Jane.Doe@Software.com (Jane Doe)
```

The **none** option simply leaves the **From:** address as it was written in the mail client. But again, this address may include hostnames or subdomains that you don't want the general public to know about, so we recommend that you use either **quoted** or **comment**.

Note: *To be rewritten, a From: address must include an address of an existing Post.Office account, and From Address Rewriting must be enabled specifically for this account. Refer to Chapter 10 for information on other applicable rules.*

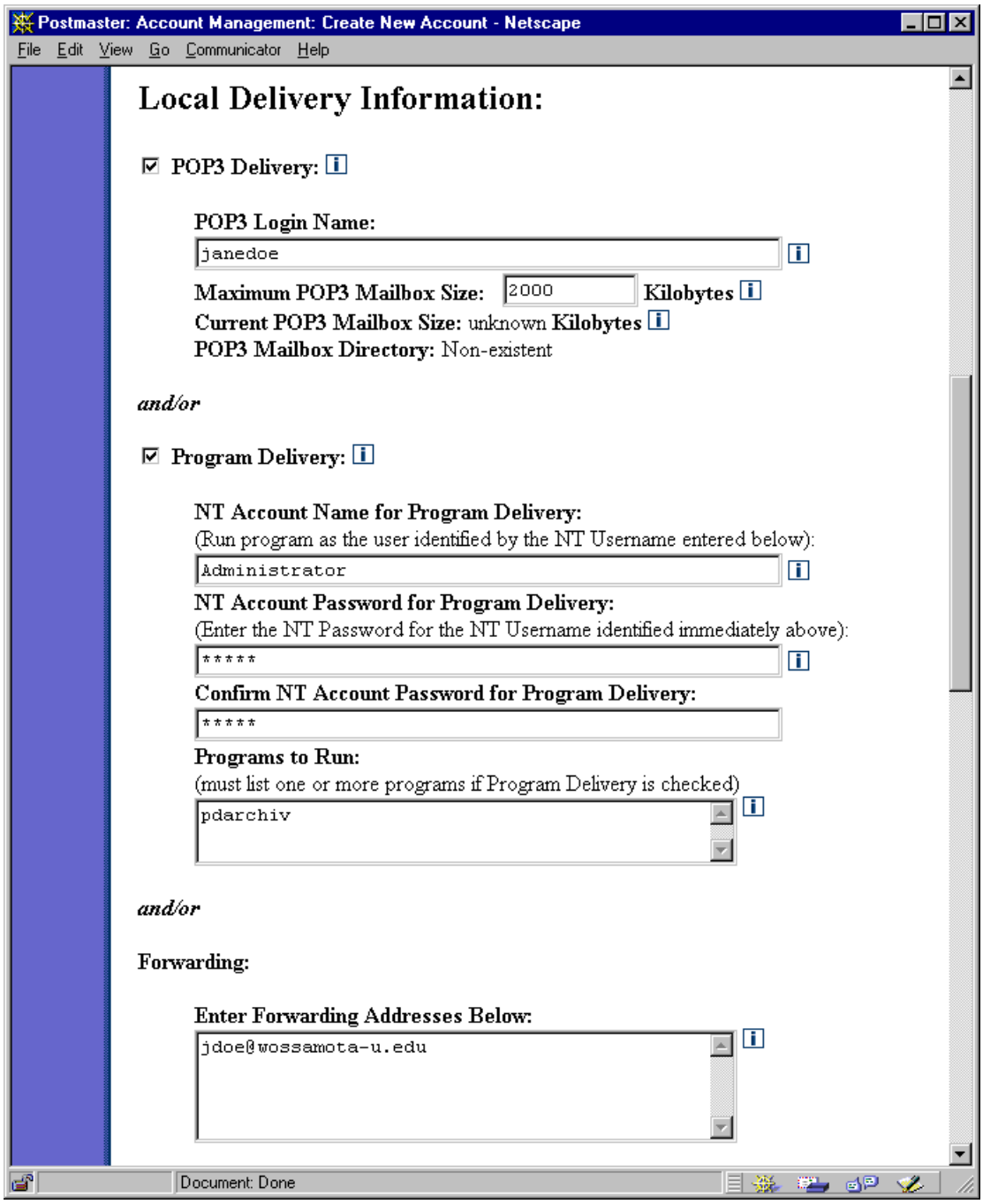

Figure 5-7: New Account Data Form (part 2 of 4)

5.3.3 Local Delivery Information

These fields define the method(s) of mail delivery which will be used to process mail that arrives for the account. Up to four options are available, and an account can include all, some, or none of them.[29] Users can enable or disable these delivery options for their own accounts, but only the Postmaster can specify login names and other potentially-sensitive information.

The following delivery options are available for all general accounts:

POP3 Delivery

The most common method of mail delivery, POP3 delivery stores messages in a "mailbox" on the server system until the user logs in with a mail client to retrieve the messages. If POP3 delivery is enabled, a unique login name must be given in the **POP3 Login Name** field. POP login names can contain just about any characters, but to avoid incompatibilities with various e-mail clients, you should use only letters (A-Z, a-z) and numbers (0-9), with no spaces or other special characters when creating POP login names.

Note: *The POP login name is in no way tied to e-mail addresses, so you may choose any format for specifying POP login names. However, it is important to note that some mail clients cannot accommodate different POP login names and e-mail address user names.*

You can also set a limit on the amount of server storage allowed for the account's mailbox, which is set in the **Maximum POP3 Mailbox Size** field. If the account reaches this limit, the Postmaster will be notified and any new mail sent to it will be "returned to sender." If the **Maximum POP3 Mailbox Size** field is left blank, the **Default maximum POP3 mailbox size** specified in the System Performance Parameters Form will be used; if this default field is also blank, no limit will be enforced and the mailbox can grow to any size.

Hint: *We recommend that you leave the **Maximum POP3 Mailbox Size** field blank for most accounts, and use the system-level **Default maximum POP3 mailbox size** to control the limit for these accounts. This allows you to later make a single change to raise or lower the POP3 mailbox limit imposed on these accounts.*

29 At least one delivery method is required if the auto-reply feature is turned off. However, auto-reply accounts (described in Section 5.1.1) do not require a delivery method.

Forwarding

This method of delivery simply takes incoming messages, modifies the destination address on the incoming envelope, and sends it to the new recipient. This delivery method is similar – both conceptually and in practice – to the forwarding of postal mail from your old residence to your new one. To request mail forwarding for an account, simply enter the appropriate address(es) in the **Forwarding Addresses** field.

There is no limit to the number of forwarding addresses that you can include here. In fact, you can create a group account (described in Section 5.1.1) by entering the addresses of multiple users as forwarding addresses for an account.

Program Delivery

Delivering mail to a program allows you to process messages with a message archive, sorting system, faxing mechanism, or do just about anything else you can devise. This feature is quite useful, but is also quite complicated, so we've set aside a special chapter just for issues related to using Program Delivery. See Chapter 6 for more information.

UNIX Delivery (UNIX platforms only)

This option (not shown in the screen shots in this section) allows for the delivery of e-mail to a UNIX maildrop file, which allows pick-up of mail using legacy mail clients. If UNIX delivery is enabled, you must specify the user's UNIX username in the **UNIX Login Name** field.

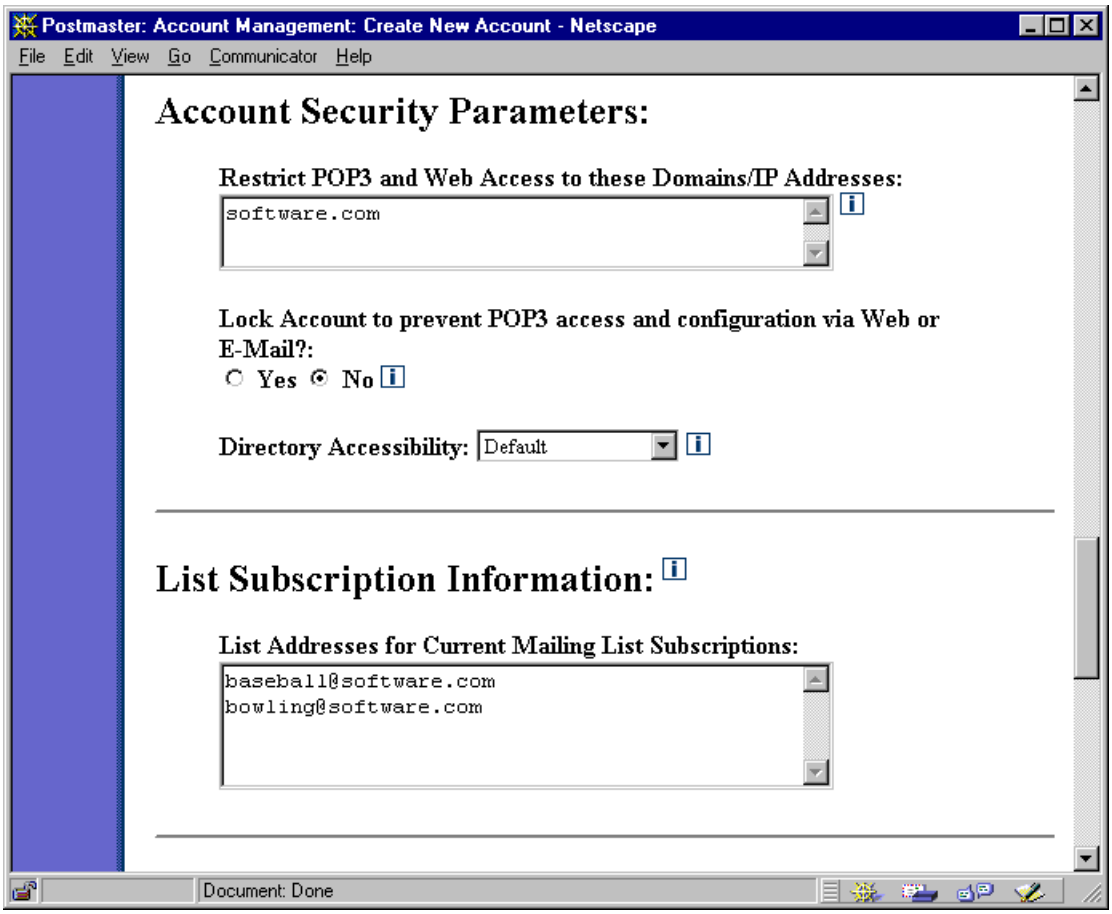

Figure 5-8: New Account Data Form (part 3 of 4)

5.3.4 Account Security Parameters

This section of the New Account Data Form contains security-related account options.

Restrict POP3 and Web Access to these Domains/IP Addresses

This field contains the general access restrictions of the account. These general access restrictions were discussed in Section 5.1.3, and can be used to prevent POP3 delivery and account modification via the web interface. Access and delivery can be restricted to a computer, a domain, or a range of IP addresses. If you leave this field blank, the user's access to the account will be unlimited.

See Section 5.1.3 for examples of Account Security Parameters.

Lock Account

This field allows you to easily restrict all access to an account by anyone but the Postmaster. Locked accounts cannot be modified by users, and although messages continue to be accepted for the account, POP3 delivery is denied as long as the account remains locked. See Section 5.4.3 for more information on locking an account.

Directory Accessibility

This field controls the visibility of an account in the Mail Account Directory listing. There are four selections for this option:

- **Default** gives the account the system's default directory listing, as defined in the System Security Form.
- **Local Only** specifies that the account is visible only in the Mail Account Directory accessible to local users (that is, users who have Post.Office accounts on your system).
- **Local and Remote** specifies that the account is visible in the Mail Account Directory for both local and remote users. This means that users from outside of your system will be able to view the Real Name and Primary E-mail Address of the account in the public Mail Account Directory.
- **Unlisted** removes the account from the Mail Account Directory entirely.

5.3.5 List Subscription Information

All Post.Office general accounts can be subscribed to one or more mailing lists. Typically users will subscribe their own accounts to mailing lists of their choosing after they receive their accounts, but there are many situations in which it is advantageous to have accounts “pre-subscribed” to one or more mailing lists. For example, when a new employee is hired for the sales department of your company’s Springfield office, they may be added to the following mailing lists:

- All employees of the company
- All employees at the Springfield office
- All sales personnel

The field labeled **List Addresses for Current Mailing List Subscriptions** on the New Account Data Form allows you to set mailing list subscriptions for a user when you create the account. For each mailing list that you want to subscribe the new account to, enter the address of the mailing list in this field. In the example above, you would enter addresses like the following:

```
employees@software.com
employees-springfield@software.com
all-sales@software.com
```

Mailing list subscriptions submitted with this form are immediately carried out and are not subject to verification, moderation, or other intermediate steps that are imposed on users who submit subscription requests.

Note: *Mailing list subscriptions can be set in the New Account Data Form when creating an account, but they cannot be set on the Account Data Form, which is used to modify existing accounts. Mailing list affiliations for existing accounts must be set through the List Manager portion of the Post.Office interface.*

Refer to Chapter 7 for more information on mailing lists.

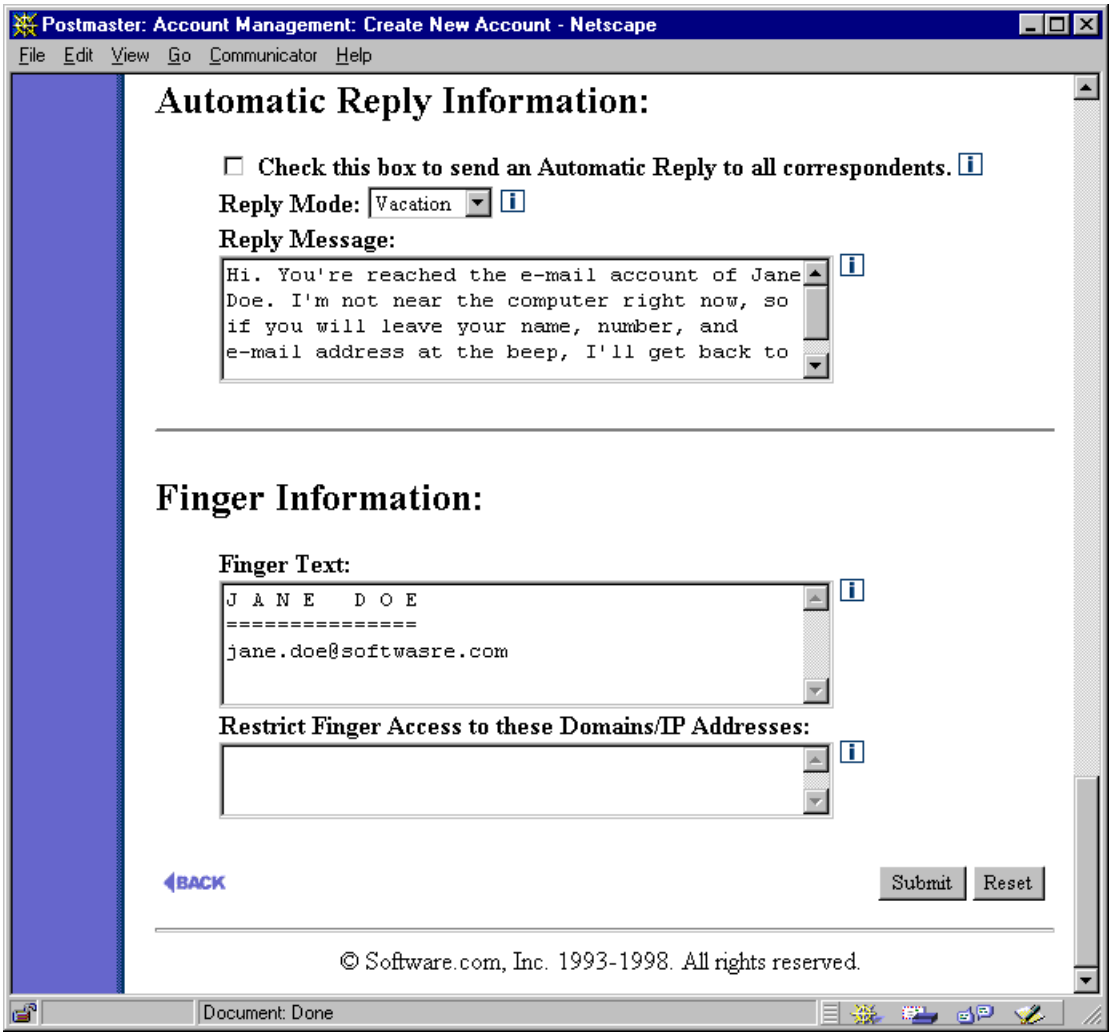

Figure 5-9: New Account Data Form (part 4 of 4)

5.3.6 Automatic Reply Information

As discussed in Section 5.1.1, all accounts have an optional auto-reply facility that can be used to automatically send information to all users who send messages to an account. The three available modes of auto-reply are the following:

- **Vacation** is used when the account's user is on vacation or otherwise temporarily unavailable. Anyone who sends any quantity of mail to the account while this feature is activated will get the account's Reply Message only once.
- **Reply** will send the account's Reply Message to anyone who sends a message to the account, and will do so every time mail arrives for the account. This feature mode can be used to provide automated information, such as a sales brochure or help information.
- **Echo** is the same as Reply, but it includes a copy of the original message as a MIME attachment along with the information in the Reply Message.

To activate the auto-reply facility, enable the check box field labeled **Check this box to send an Automatic Reply to all correspondents**, select an auto-reply mode from the **Reply Mode** drop-down menu, and enter information in the **Reply Message** field. There is no limit on the number or type of characters that can be entered in the **Reply Message** field.

Anti-Looping Precautions

The auto-reply feature includes extensive precautions to avoid generating reply messages to mailing lists (which could create a mail loop, and would be unpleasant for the list's subscribers). Any address which includes the following signs of a mailing list will not be sent an auto-reply:

- The phrases "-request" or "owner-", which are generally considered the convention for mailing list addressing.
- The words "majordomo", "listproc", or "listserv", which are the names of common mailing list programs, as well as "list.manager," the List Manager account of Post.Office.
- The auto reply handler also parses the headers looking for the "Precedence:" (sendmail) and "Auto-forwarded:" headers (X-400) which are commonly used in mailing lists.
- The auto reply handler also requires that the incoming message has a destination address – either the To: or CC: header lines – that matches an address for the account. This prevents auto replies from going out in response to forwarded mail.

5.3.7 Finger Information

The fields in this section of the New Account Data Form are used to set and control the directory information provided for finger queries for this account.[30]

Finger Information

This field contains the actual text that is provided in response to a finger query. There is no limit on the number or type of characters that can be entered in this field, but keep in mind that some displays will only show the last 24 lines or so (the rest will scroll off the top of the screen).

Finger Access Restrictions

This field is similar to the access restrictions on account management and POP3 delivery described earlier in this chapter. You can make your finger information available to anybody by leaving this field blank, or you can limit finger access to a specific domain or range of IP addresses. For example, by limiting finger access to computers within your company, you can make confidential finger information – like home telephone numbers – available only to those who should be in the know.

5.3.8 The Greeting Message

Upon creation of a new account, a greeting message is sent to this account. The greeting message informs the user that an e-mail has been opened for them, and gives them a bit of information about the account, including instructions for accessing a web or e-mail form to modify their account.

Note: *This greeting message is optional – you can determine whether newly created accounts receive it. This option is located on the Mail Routing Form, as described in Chapter 4.*

30 Remember that all addresses assigned to an account are just as valid as its primary address. This means that a finger query to any of the account's address will get the same information.

The first portion of the greeting message is shown below.

```
An electronic mail account has just been opened for you and has been
configured as indicated below.  For information on how to make
changes to your mail account or to obtain explanations about any of
the fields, see the instructions that follow this account summary.

    Your-Name:              [Susie Queue]

    (Note: Your name is sometimes referred to as your account name.)

    Internet-Addresses:     [susie.queue@software.com]
                            [susie.queue@sparky.software.com]

    Finger-Information:     []

================================================================
Here's some information about changing your account:

Only the system administrator can change your name or Internet
addresses. If you want to change your password or finger information
you can do so with a World Wide Web browser or via E-mail.  You
simply fill out a form indicating the desired changes and submit the
form to the mail system. To request the required Information form:

     via the Web:  connect to http://sparky.software.com:81

     via E-mail:   You can get the E-mail form to modify your
                   account by sending a new message to the
                   address, <Accounts@sparky.software.com>,
                   with the word "Information" as the message
                   body like this:

                    To:      Accounts@sparky.software.com
                    Subject: Information

                   Note:  The word "Information" is case sensitive
                   and must be entered exactly as indicated.

After receiving the Information form, make the appropriate changes,
put in your password, and submit the form.  (Note:  If you are using
the E-mail interface you'll need to create a reply message including
the content of the original E-mail, then edit and submit that
reply.)

If you don't receive an error message, the changes have been
accepted.

================================================================
Here is an explanation of each of the fields shown for your account:

...
```

Figure 5-10 The new account greeting message. Only the first portion of the form is shown; additional information describing account attributes is included below the information shown here.

Although the greeting message offers some introductory information for users to manage their accounts, the Postmaster may also want to distribute the Post.Office manuals that were written specifically for these users. Two such manuals exist: the *Post.Office User's Guide*, which is for all users with e-mail accounts in Post.Office; and the *Post.Office List Owner's Guide*, for those users to whom you grant list ownership privileges. Online versions of these documents are available from the Help menu button of the web interface, and are also available from the Software.com web site (`http://www.software.com`). You can also purchase printed copies of these and other manuals from your Post.Office vendor.

5.3.9 Setting Defaults

The secret to streamlining the creation of new accounts is to set default values for as many account attributes as possible. Unlike addresses and POP3 login names, which must be unique throughout Post.Office, most account attributes are things like delivery methods, access restrictions, and finger information; these options typically start out the same for all accounts, so by setting defaults for these fields, you can pretty much ignore them when creating new accounts.

The form for setting default account values, the Default Account Data Form, is identical to the New Account Data Form illustrated in the previous section. This Default Account Data Form can be invoked from the **Edit Default Account Data** link of the Account Administration menu, or the **Default Account** link on the List of Accounts menu. Both links display the same form, so use whichever one is easier.

Default account attributes provide a template for creating new accounts. The information provided in the Default Account Data Form is inserted into the fields of the New Account Data Form whenever you request to create a new account.

We won't show you the Default Account Data Form, since it's exactly like the New Account Data Form, so look back at the previous pages if you want to see this form again. Here's a review of the account fields and some guidelines for typical default values:

User's Real Name. You may find it helpful to specify a default name that reflects your preferred naming convention. Remember that the List of Accounts menu sorts accounts by Real Name, so if you want this menu to display accounts in a certain way, setting a default name here is a good way to set that standard.

Among the popular default Real Name values are:

```
First Last
First M. Last
Last, First
```

Mail Account/POP3 Password. A default password cannot be set in the Default Account Data Form, so don't bother.[31]

Ignore the Mail Account/POP3 Password entered above and use the Logon Password for the following NT username. If you are running Post.Office on an NT platform and want to use this feature, you should set this field to on by default. Enabling this option for a new account requires that you also specify an NT username, so if you forget to add one, you'll be reminded to enter one before the account is created.

Primary E-mail Address. As with the Real Name field, you may find it helpful to set a default e-mail address that includes your domain and reflects your preferred address format. The following are some typical defaults:

```
john.doe@software.com
jdoe@software.com
johnd@software.com
doe@software.com
```

Additional E-mail Addresses. If your users need to receive e-mail in more than one domain, or you want to use multiple addressing formats (like those shown above), you probably also want to put those here.

From Address Rewrite Style. The e-mail clients of your users may include hostnames or subdomains in the **From:** header of their outgoing messages, which you may not want. That's why it's a good idea to select either **quoted** or **comment** as a default for this field.

31 But remember that the default password is Lock. If you create a new account and forget to change this default, the new account will be locked.

Local Delivery Information. Because most accounts you create will probably use POP3 delivery, you may find it useful to enable this delivery option in the Default Account Data Form and set a POP login name that fits your preferred format (for example, “FLast”). If most of your accounts will use an alternative delivery method, such as UNIX delivery or a sorting system used with the Program Delivery feature, set these delivery options.

Restrict POP3 and Web Access to these Domains/IP Addresses. This is a highly-recommend field for setting defaults. Enter the domains, host names, or IP address that you consider appropriate for accessing your e-mail system. Individual accounts may require more or less strict access rules than these defaults, but you should set a default access rule that applies to the general case. The most common account access restriction is to your domain (for example, `software.com`), but you should specify whatever is appropriate for your organization.

Directory Listing. You should set a default here that represents your standard policy for using the Mail Account Directory feature, described in Section 5.8. The recommended default value is (appropriately enough) **Default**, which gives an account the directory listing status defined as the global default in the System Security Form.

List Addresses for Current Mailing List Subscriptions. If you want most or all of your accounts to be subscribed to a particular mailing list, enter the address of this mailing list here.

The **Reply Message** and **Finger Information** fields can be set by individual users, so picking defaults for these is not a high priority. But you (and they) may find it helpful if you provide a template for vacation and finger messages.

Finger Access Restrictions. If you want to use the finger facility as an internal directory service, specify your domain or range of IP addresses in this field. If you want to leave finger access available to the outside world, leave this field blank.

5.4 Viewing and Modifying an Account

After you’ve created all of your e-mail accounts, you may need to periodically go back to those accounts to add a new address, update auto-reply information, or change access restrictions. You might also just want to look over the attributes of an existing account, especially when trying to assist users who are having problems connecting to Post.Office. Account information can be viewed and edited like this at any time with the Account Data Form.

5.4.1 List of Accounts

The simplest method of accessing account information is through the List of Accounts menu. Recall from Section 5.2 that this menu is invoked from the Account Administration menu, looks like this:

Figure 5-11: List of Accounts menu

Each name in this menu is a link to the Account Data Form for that account, which makes this a very convenient place to start when modifying accounts. However, if you have several thousand accounts, it can be tedious to search through this menu. If this is the case for your site, you can also access a specific Account Data Form from the shortcut field at the bottom of the Account Administration menu, which allows you to skip the List of Accounts menu entirely.

5.4.2 The Account Data Form

The Account Data Form contains all of the information related to an account. It can be displayed from the Account Administration menu or from the List of Accounts menu, and lets you view and modify account information at will.

Although the Account Data Form is nearly identical to the New Account Data Form shown in Section 5.3, there are enough differences between the two that the Account Data Form is worth seeing here in the following illustrations:

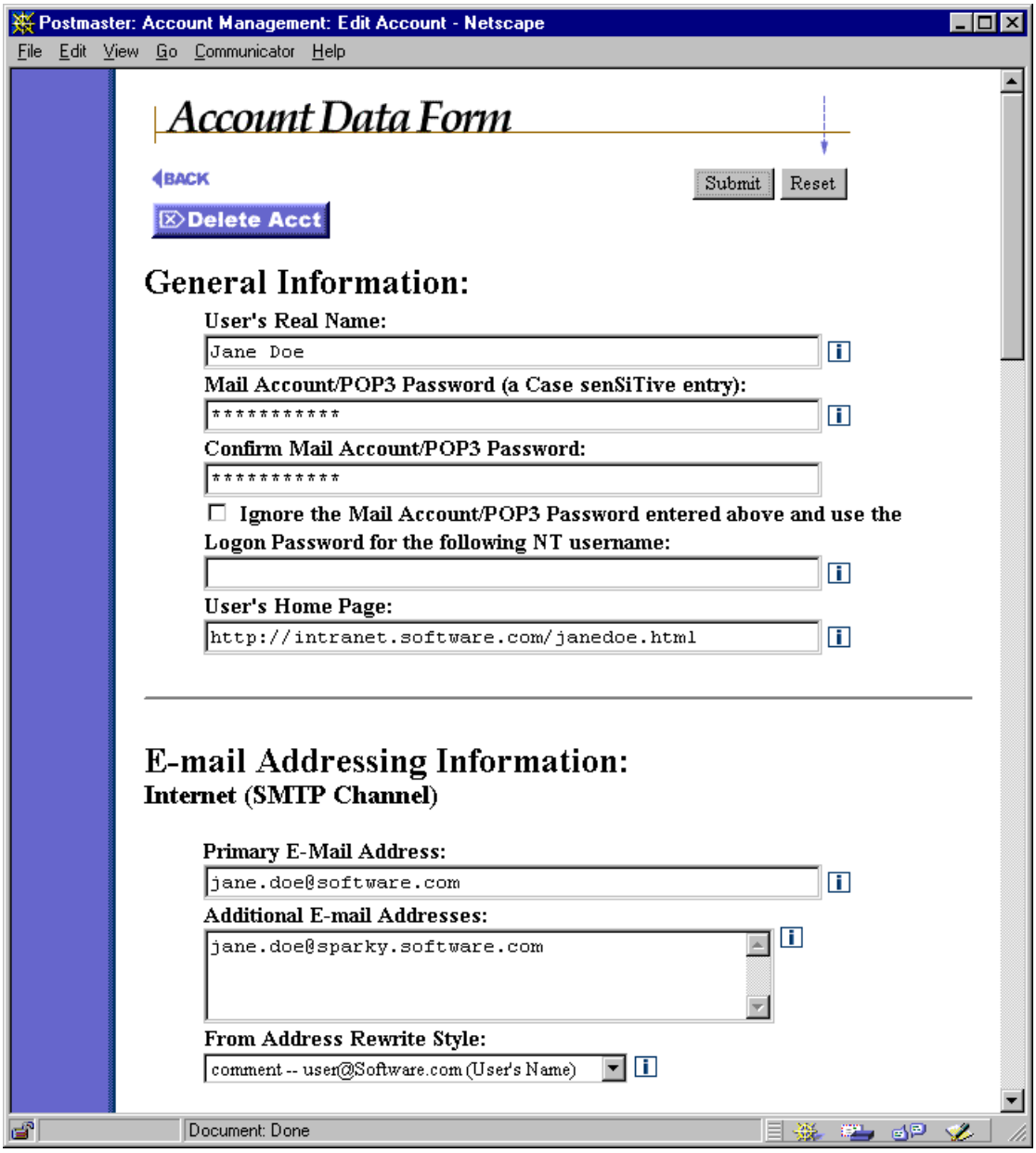

Figure 5-12: Account Data Form (part 1 of 4)

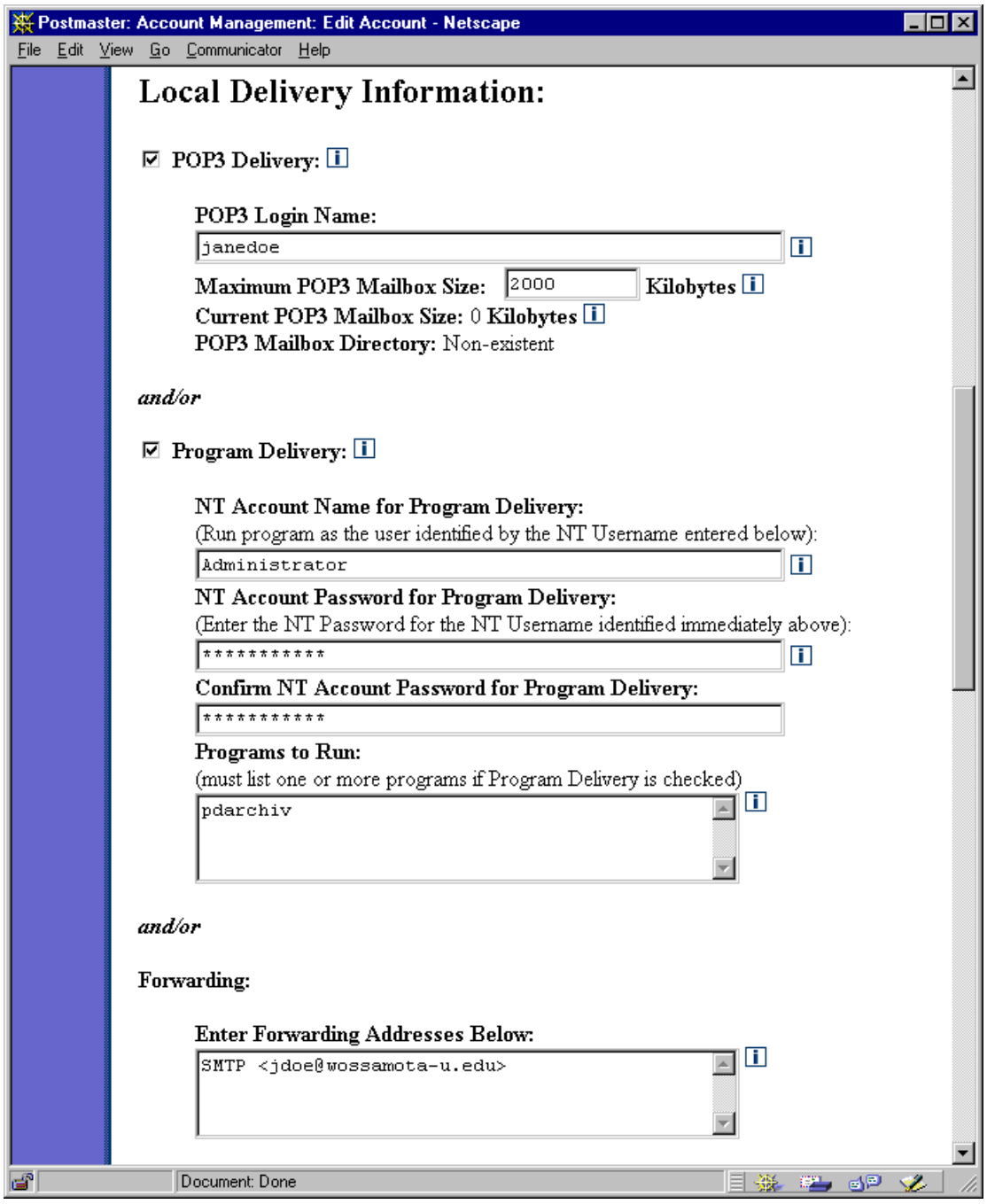

Figure 5-13: Account Data Form (part 2 of 4)

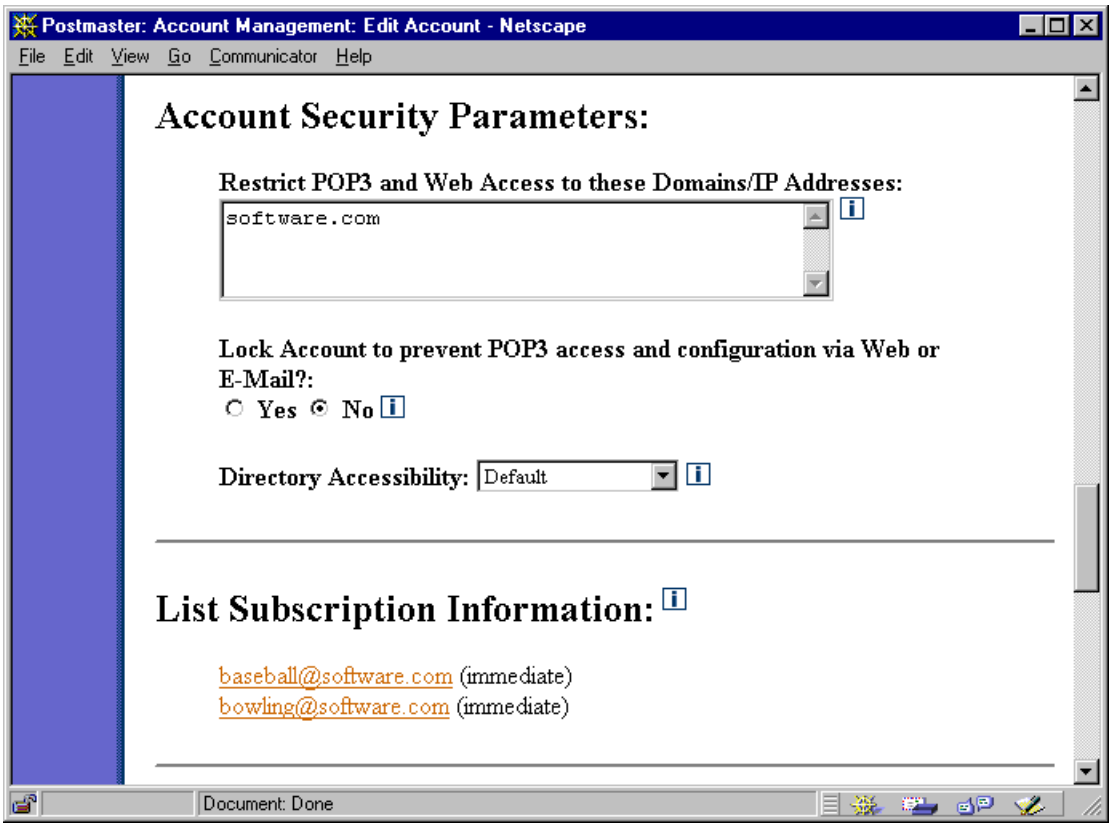

Figure 5-14: Account Data Form (part 3 of 4)

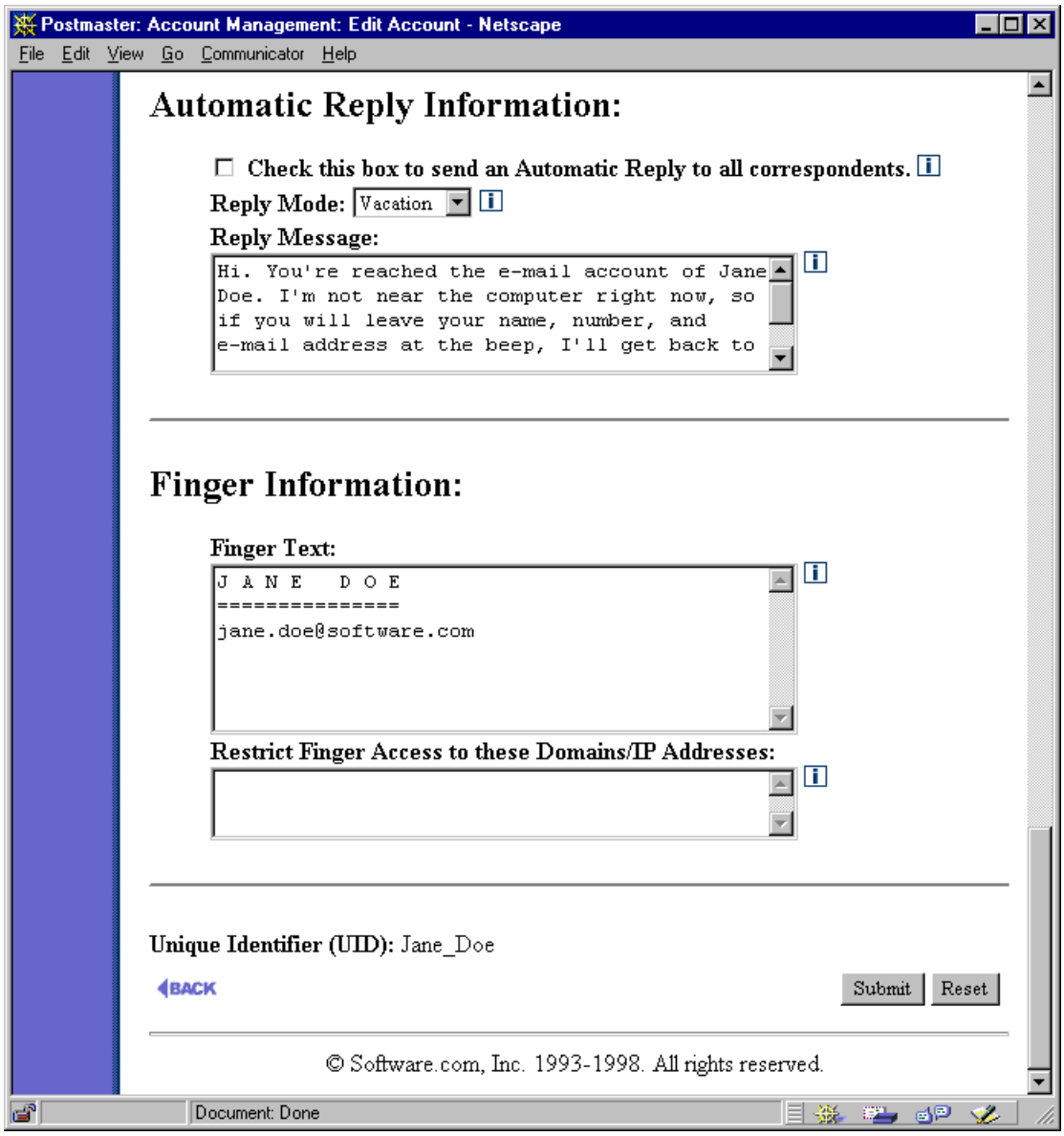

Figure 5-15: Account Data Form (part 4 of 4)

Notice three important differences between the Account Data Form and the New Account Data Form. First, at the top of the form (shown in Figure 5-12), the button labeled **Delete Acct**. This button – as you probably guessed – is used to delete the account, and is discussed further in Section 5.6.

The second difference is that the **List Subscription Information** section of the form does not include the text field for entering mailing list addresses, but does contain a list of mailing list addresses. Each mailing list address is a link to a Mailing List Data Form, which defines the attributes of the mailing list; this form is exhaustively displayed and described in Chapter 7.

Note: *You cannot subscribe an existing account to a mailing list with the Account Data Form, as you did with the accounts you created with the New Account Data Form. Refer to Chapter 7 for instructions on subscribing an existing account to a mailing list.*

The final piece of information on the Account Data Form is the **Unique Identifier (UID)**. This value is used with the Post.Office command-line utilities. The value of the UID is based on the Real Name, and is set at the time of account creation and cannot be modified. Refer to Chapter 11 for information on using the UID with account management utilities.

To make changes to an account, simply modify the appropriate value in the Account Data Form and submit the form. To cancel your changes, click the Reset button or the BACK link.

5.4.3 Locking an Account

Locking an account is a special type of account modification. When an account is locked, it cannot be modified by its user, and although messages continue to be accepted, POP3 delivery requests for the account is not allowed. This option is very useful if you have a user who is paying for an e-mail account on your system, but who hasn't paid his/her bill to you lately. Locking an account lets you "cut off" the user's e-mail access, but unlike simply deleting the account, you can later restore normal operation later by simply unlocking it.

To lock an account, go to the radio button field labeled **Lock Account to prevent POP3 access and configuration via Web or E-mail?** (shown in Figure 5-14) and select **Yes**. After you submit this change, the account's user will immediately be disallowed from changing the account or getting mail via POP delivery. Again, messages will still be accepted for the account, but will remain untouched in the account's POP mailbox until you decide that the user is again worthy of getting e-mail.

To unlock an account, simply select the **No** button in the **Lock Account...** field and submit the form.

5.5 Managing the Postmaster Account

The Postmaster account – that is, the administrative account itself, not the individual accounts of users like you who have Postmaster privileges – is technically an account like any other, and can be viewed and modified in an Account Data Form. While you will seldom need to edit this account, there are two basic operations which require it: the assigning of additional Postmasters, and changing the Postmaster password.

In both of these cases, you must first invoke the Postmaster Account Data Form, a special version of the Account Data Form, which can be done from the Account Administration menu or the List of Accounts menu. The form is almost identical to the regular Account Data Form, with the exceptions noted below.

5.5.1 Assigning Additional Postmasters

The Postmaster, as you probably realize by now, has a lot of responsibilities: setting system configuration options, handling error mail, creating mail accounts and mailing lists, and a bunch of other activities that only the Postmaster can perform. Because this can all become quite a workload, you may decide that you want one or more users to share in your tasks of administering Post.Office. You can do so by assigning new Postmasters.

Just as you were once granted Postmaster privileges by somebody, you can grant this same supreme authority to other users. Any user who has an e-mail address can be granted this status, but remember that Postmaster status allows an individual complete and total access to your mail system – it should not be granted carelessly.

To assign an additional Postmaster (or two, or more), move down the Postmaster Account Data Form to the delivery information section. Notice in the following portion of the Postmaster Account Data Form that the delivery options have been divided into two sections: Required Delivery Information and Optional Delivery Information.

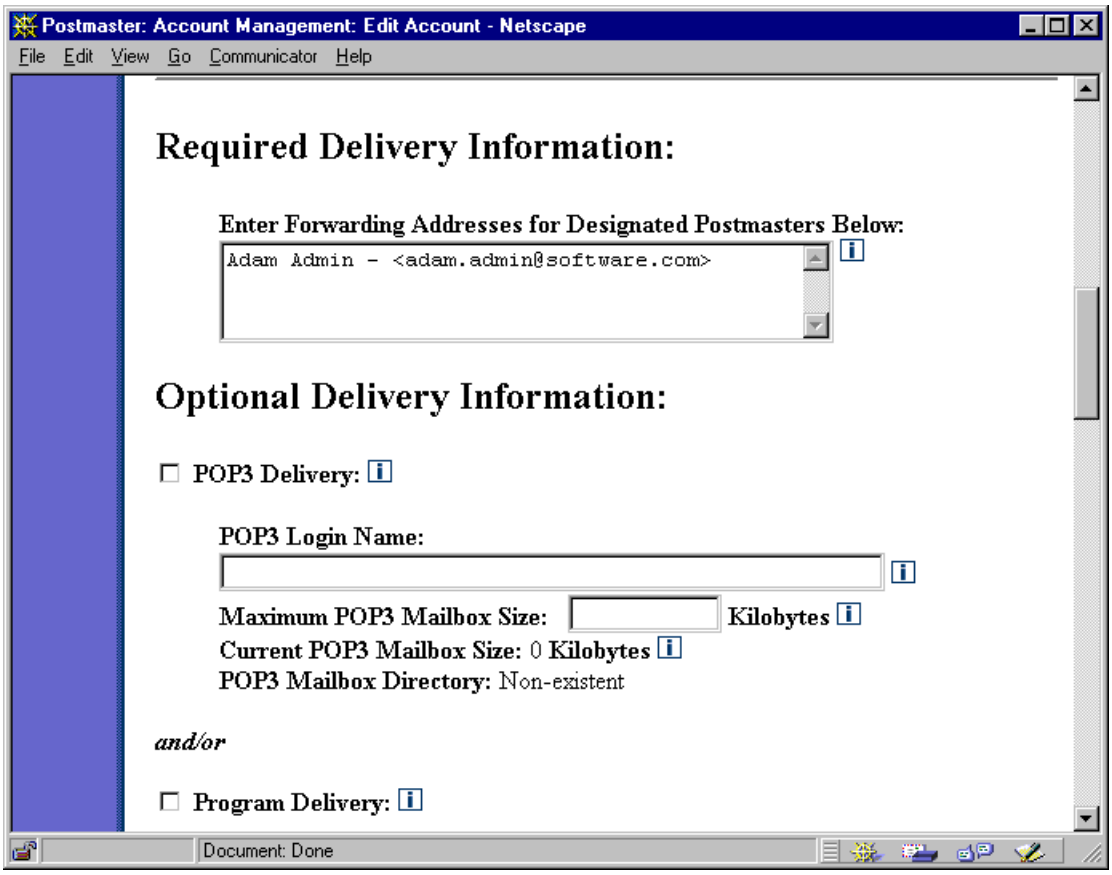

Figure 5-16: Postmaster Account Data Form (part of it, at least)

The Required Delivery Information section of the form specifies the local users who have been granted Postmaster privileges (naturally, this should include you).[32] To add or remove from the ranks of users who have Postmaster privileges, enter the address of the new Postmaster and/or delete the address of the outgoing Postmaster. There's no limit on the number of Postmasters that you can create by adding them to this field, but you must always have at least one address specified here (that is, you must always have at least one Postmaster).

32 The field that contains the addresses of the designated Postmasters is actually a variation of the Forwarding Addresses field of the Account Data Form. Any local user account that receives mail addressed to the Postmaster is, by definition, a Postmaster – the change in the labeling of this field in the Postmaster Account Data Form is just for clarification.

5.5.2 Changing the Postmaster password

Like all accounts, the Postmaster account has a password associated with it. However, the Postmaster password is an extremely important piece of information, and is required for carrying out any and all configuration and account management. It is something you don't want to share, since it gives a user access to every detail of the Post.Office system.

Obviously, you should never, *ever* give the Postmaster password to anybody who should not have Postmaster privileges. However, for security purposes it can still be a good idea to change the Postmaster password every once in a while. You can do so by entering a new password in the **Postmaster Password** field of the Postmaster Account Data Form.

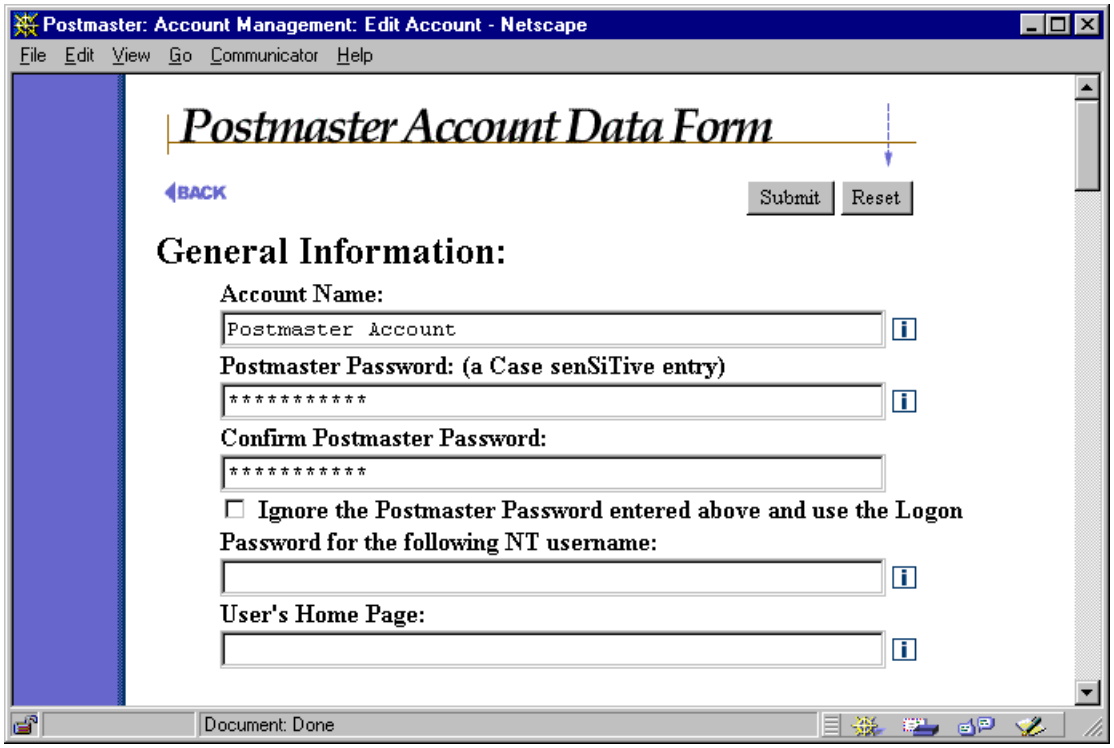

Figure 5-17: Postmaster Account Data Form (another part of it)

As with all passwords, the **Postmaster Password** field is Case senSiTive and requires you to reenter the new password for confirmation. When choosing a Postmaster password, pick something even more difficult to guess than your personal account password – after all, we're talking about total access to your mail server here. But again, if you make the Postmaster password excessively cryptic, you might forget it yourself, of write it down on a piece of paper that you keep in a desk drawer (thus defeating the purpose).

Warning! DO NOT FORGET THE POSTMASTER PASSWORD! It is very difficult to recover this information. If you do forget your Postmaster password, and nobody else knows it either, contact `support@software.com`.

5.6 Deleting Accounts

Accounts can be deleted easily in the web interface by clicking the **Delete Acct** button at the top of the Account Data Form. To refresh your memory, the form (and the button) look like this:

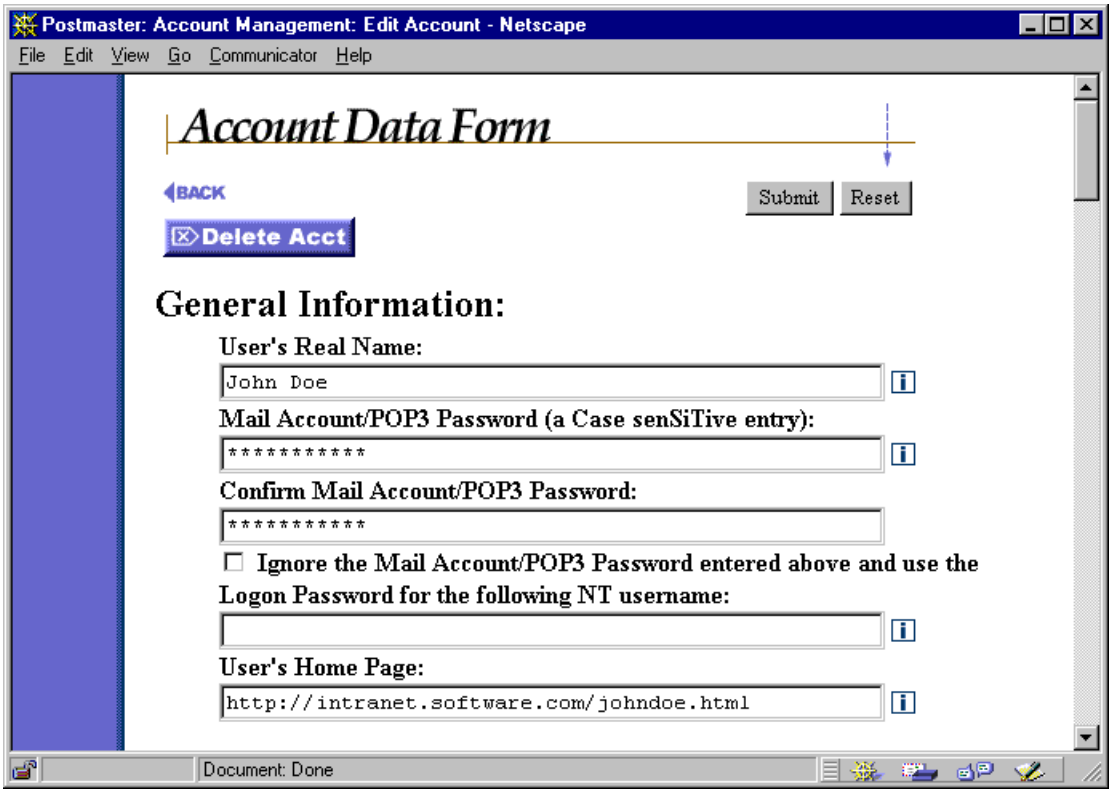

Figure 5-18: Account Data Form (top portion only)

When you click this button, you will go to a confirmation form which reminds you that you are about to permanently delete the account from Post.Office with no hope of recovery.

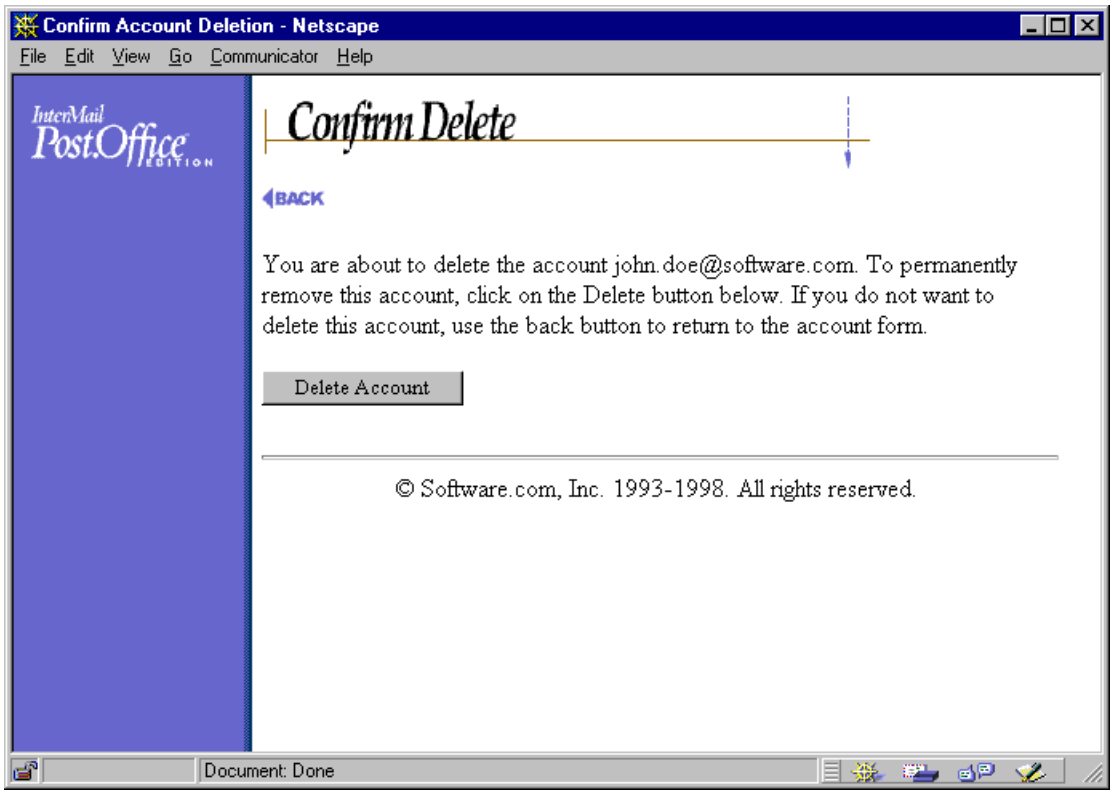

Figure 5-19: Delete Confirmation Form

If you're not sure if you want to really delete the account, click on the BACK link to return to the Account Data Form. If you're absolutely, positively sure that you want to forever obliterate this account, click the **Delete Account** button to complete the deletion.

Deleting an account automatically removes it from any group accounts or mailing lists (provided those group accounts and mailing lists are on the same installation of Post.Office as the account, of course). Remember that deleting an account also deletes the account's mailbox and all messages – read or unread – contained in it.

5.7 Broadcasting Messages to All Accounts

There will probably be occasions when you'll want to send a message to all of your mail account users. For example, you may want to inform users about a new e-mail policy, or alert everybody that mail service will be off-line temporarily. Post.Office allows you to broadcast information like this by sending a message to an All Mailboxes account, which delivers your message to every account on your system that uses POP3 delivery.

The All Mailboxes account is implemented as a special Post.Office mailing list. This means that you can define special security options for specifying who can post to the account, and enjoy all the benefits of other mailing list features. To send a message to all POP3 accounts, address it to the list's posting address:

`all-mailboxes@`*`host.domain`*

For security reasons, the All Mailboxes mailing list is locked upon installation, so you must unlock this mailing list (as described in Chapter 7) before you can use this feature. Also, the policies for this mailing list initially allow only the Postmaster to post messages to it. Refer to Chapter 7 for more information on the All Mailboxes mailing list.

5.8 Mail Account Directory

Post.Office includes an optional Mail Account Directory, which allows your users to get information on the e-mail accounts at your site. You can even allow users throughout your network (or the Internet) view this directory, if you so choose. This is very handy for allowing users to find e-mail addresses for folks whom they'd like to correspond with.

The following is an illustration of the Mail Account Directory visible to your local users (the public directory is practically identical):

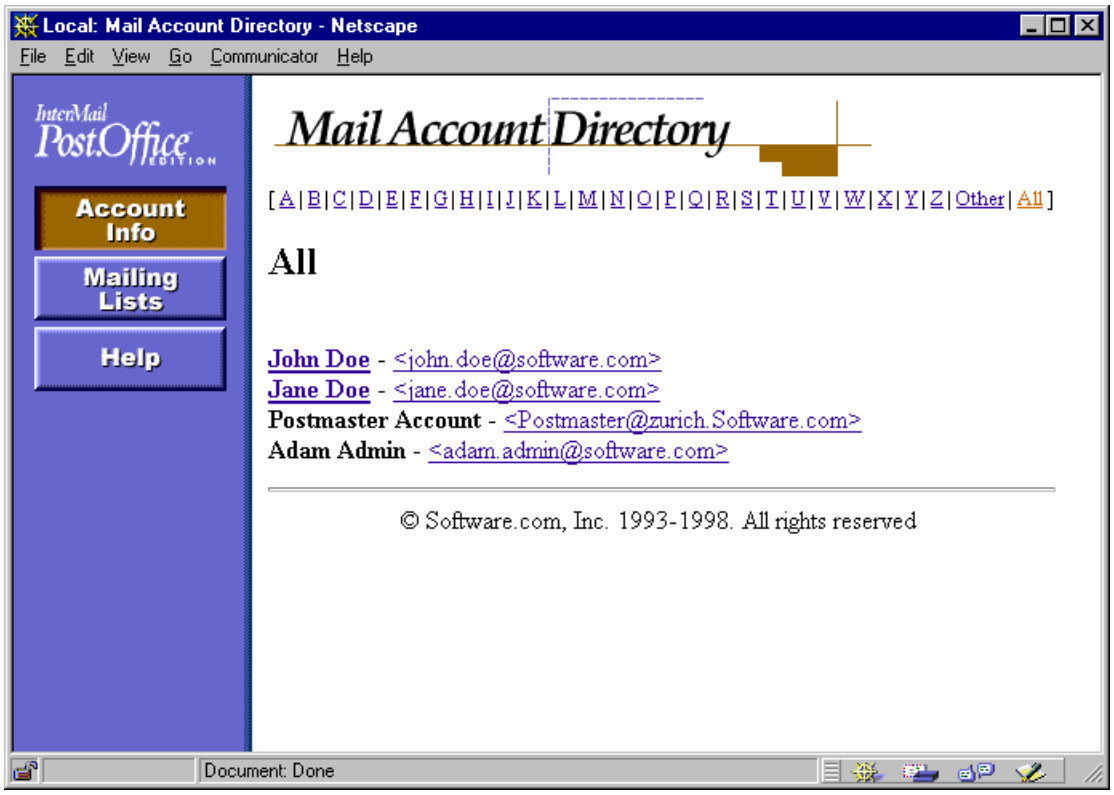

Figure 5-20: Mail Account Directory (local user version)

Like the Postmaster's List of Accounts menu, the Mail Account Directory gives the name and e-mail address associated with each listed account. If a user has a home page defined for his or her account, the user's name is a link to that home page. The location of an account's home page, as well as the visibility of the account in the directory, are defined in the Account Data Form.

Warning! Account data may be considered sensitive, so the Mail Account Directory may not be appropriate for your site (this is especially true for mail servers that are visible to the entire Internet). You should decide how (or if) you want to use this feature, and then use the options on the System Security Form (described in Chapter 4) to show or hide the public Mail Account Directory, and set a global account listing default.

6

Program Delivery

This chapter describes the Program Delivery feature of Post.Office mail accounts, and is intended for advanced users. The topics discussed in this chapter include:

- General information regarding Program Delivery
- Using Program Delivery on NT platforms
- Using Program Delivery on UNIX platforms

6.1 Program Delivery Basics

Along with typical delivery options like POP3 delivery, mail forwarding and UNIX delivery, Post.Office allows you to deliver messages to the program of your choice. This method of delivery – called Program Delivery – allows you to use a message archive, sorting system, faxing mechanism, or do just about anything other type of "plug-in" delivery mechanism that you can devise.

Just as individual users can select POP3 delivery for their accounts, they can also select Program Delivery. However, the Postmaster usually must move the program in question to a special Post.Office directory, set appropriate file permissions, provide a server login username and password, and other administrative tasks required to make this delivery method work. Setting up Program Delivery is therefore really a Postmaster-only activity.

6.1.1 When a Message Is Delivered to Program

You may be a little confused by this notion of delivering mail to a program instead of to something simple like a user mailbox. What the heck does that mean? Well, you've actually been delivering mail to a program all along – when your mail client sends off messages, it delivers them to a program called Post.Office. This program knows what it is supposed to do with such a message and processes it accordingly, which could mean resending it to another e-mail server, storing it in a user's mailbox, or any of the other neat delivery things that Post.Office can do.

The programs used with Program Delivery work the same way. They read as their input the entire contents of a message (headers plus body), do whatever sort of processing they were designed to do with these messages, and then signal to the sender – in this case, Post.Office – that the message was accepted and processed correctly (or return an error to indicate that delivery did not go as planned). One example of this is a message archiving program: when mail arrives to the appropriate account, Post.Office starts up the archiving program, which then reads the message line-by-line from the system's standard input

(STDIN), writes it to whatever file it uses to store messages, and then quits with the appropriate return value to signify success or failure.

Of course, the "starts up the program" part of the above scenario is actually a bit complicated. Like any program, the programs used with Program Delivery must be executed on the server system by a particular user (that is, by a UNIX or NT user account) which has appropriate access to both the executable file and to the directory in which it resides. This is where the Program Delivery concepts of *trusted programs* and a *trusted program directory* enter the picture.

6.1.2 Trusted Programs

A trusted program is an program that you, the Postmaster, deem acceptable for using with the Program Delivery feature. Because programs can access potentially sensitive areas of your server system, not all of them should be "trusted" to handle e-mail messages. Even if executed by a user with malicious intent, trusted programs should still behave nicely. Trusted programs typically consist of mail filters, distribution list software, automatic responders, and the like – useful for processing e-mail, but useless to hackers trying to compromise server security.

You signify your trust in a program by copying it to the trusted program directory (see below) and setting its file permissions to allow it to be executed by one or more users. Remember, you'll be supplying the name of a user account that Post.Office should use to invoke the program, so this user account should have execute access.

Note: *On NT, only trusted programs can be used with Program Delivery. On UNIX platforms, however, you can arrange things so that Program Delivery can invoke any program it can get to, which is known as running in* open *mode. However, open mode represents a significant security risk and is not recommended. See Section 6.3.1 for more details.*

It is critical that the behavior of each trusted program is well understood and known to be safe. In particular, programs that interpret their input as a sequence of commands (such as UNIX shells like `sh` and `csh`, or scripting languages like `perl` and `tclsh`) should not be set up as trusted programs. However, some scripts that run under these command interpreters may be considered safe after careful inspection.

6.1.3 Trusted Program Directory

Post.Office looks for trusted programs in a special directory known as the *trusted program directory*. Only executable files (or a hard or soft link to an executable) found in this directory will be considered to be trusted programs. If a path to the executable file is included in the Program Delivery options for an account, this path will be ignored and Post.Office will look for the program only in the trusted directory; this allows the System Administrator to specify the exact executable files which will be run by the Program Delivery feature.

On NT platforms, the trusted program directory is located in the Post.Office directory and is named `trusted`. For example, the trusted directory may be located at:

```
C:\win32app\Post.Office\trusted
```

On UNIX platforms, you can set any directory to be the trusted program directory (as described in Section 6.3.2).

Remember, the server login account which is used to invoke a program in the trusted directory must also have appropriate access to the trusted directory itself.

6.1.4 Program Delivery Errors

Just as Post.Office will sometimes encounter a problem when processing messages (such as an invalid destination address, etc.), the programs used with the Program Delivery feature may encounter problems which prevent them from completing their task. Such errors generate an error message that is sent to the Postmaster which alerts them to this development. In most cases, these errors are caused by the specified user having insufficient permissions to run the selected program.

When a Program Delivery error occurs, recheck the user permissions and the Program Delivery information for the specified account (username, program, etc.). Post.Office will later attempt subsequent deliveries of the message to the specified program, which allows you to find the source of the problem and correct it.

6.2 NT Program Delivery

Setting up Program Delivery on NT platforms generally involves three steps. First, you must have an actual program that can be used with Program Delivery. Second, you must copy this program to the trusted directory and set the appropriate file permissions for it. Finally, you must specify the Program Delivery method of delivery for an account instead of (or in addition to) POP3 or other delivery methods. This final operation requires you to provide the command-line arguments which invoke the program, as well as the login name and password of the NT account which will run the program.

We'll assume that you already have a program and are ready to start setting up. Refer to Section 6.2.4 for help on writing new programs for use with Program Delivery.

6.2.1 Setting Up Access Rights

The Program Delivery feature requires that special rights be assigned to the NT user responsible for running Post.Office. These access rights can be set automatically during installation. If the person who installed Post.Office did not enable these access rights, you must do so manually before using Program Delivery.

The Post.Office user (the NT user that was created at installation time and through which Post.Office runs) must have the following access rights:

- Act as part of the operating system
- Increase quotas
- Replace process level token

These options can be set in the NT User Manager. If you don't know how to set these, you can consult the Post.Office FAQ for line-by-line instructions for doing this. After making the necessary changes, reboot your system. The new permissions for the Post.Office user will then take affect for your Post.Office MTA service.

6.2.2 Setting Up Programs

Preparing an program for use with Program Delivery is as simple as copying it to the trusted directory, and granting execute access for this executable file. The account to which you grant this access should be the same account that you specify in the Account Data Form when setting up Program Delivery for an account (see below), so you need to know the username and password for this account.

When the program executes, it will use as its working directory the `Post.Office\working` directory, which means that any files created by the program which do not specify a path will be placed here by default. For example, if the trusted directory is located at

```
C:\win32app\Post.Office\trusted
```

then the working directory for trusted program is

```
C:\win32app\Post.Office\working
```

Your programs may create files in directories other than this one, of course, provided that the account which executes the program has appropriate access to the target directory.

6.2.3 Enabling Program Delivery For an Account

Setting up Program Delivery for an NT-based account requires you to enable this option in the Account Data Form, provide the username and password of the NT user who will execute the program, and specify the program to be run (with any arguments). The following illustration shows a snippet from the Account Data Form for NT platforms which includes the relevant fields:

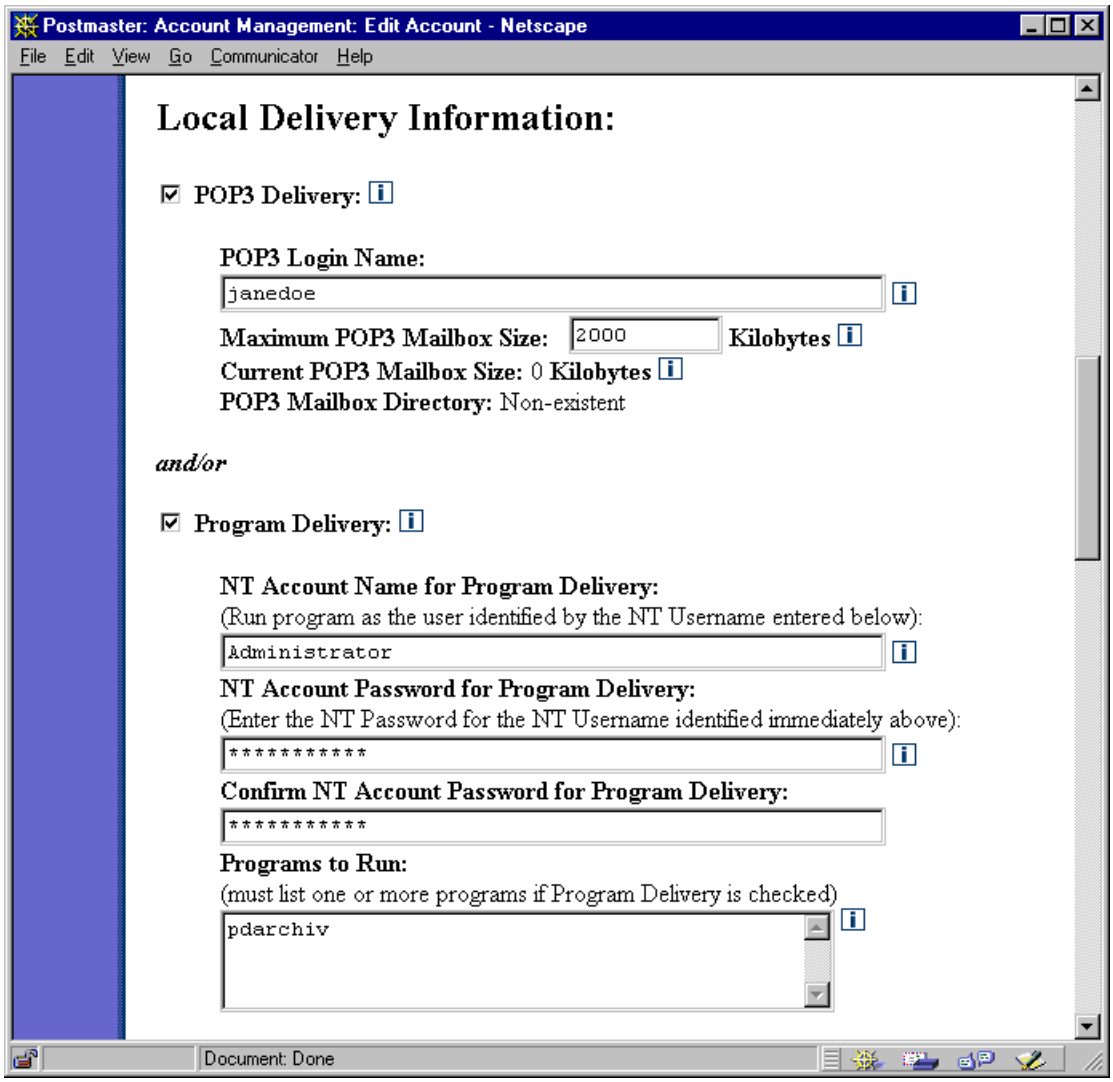

Figure 6-1: Account Data Form on NT platforms (delivery section)

Enable the check box labeled Program Delivery to set this option for the account. You must also provide an **NT Account Name** and corresponding **NT Account Password** (re-entered for confirmation) in the appropriate fields, as well as one or more **Programs to Run**. If the desired NT user is a domain user, you should enter `domain\username` (for example, `sales\joe`). If the user is a local user, you should enter the username only. The program entry (or entries) must include any and all arguments required for execution.

Note: *NT Program Delivery, unlike its UNIX counterpart, does not allow the use of input/output redirection (<, >, >>) or piping (|) among the command-line arguments.*

When mail arrives to an account using this delivery method, the specified program will be run as the NT user specified in the **NT Account Name** field. If the user has sufficient permissions and is to do this, it's then up to the program to read the message from standard input and process it correctly.

6.2.4 Creating NT Programs for Use With Program Delivery

Programs used with the Program Delivery feature of Post.Office can be extremely simple (like a message archiving system) or extremely complex (like an automatic faxing mechanism). However, all programs used with NT Program Delivery must follow the following rules:

1. A delivery program should be a console application (a command-line utility, as opposed to a GUI application).
2. It may be written in C, C++, or any other language that creates a standalone executable that can read from STDIN.
3. It should read from STDIN (e.g. using `gets` to read a line at a time). The entire contents of the message, including headers and signature will be sent. An EOF (end of file) condition will be detected at end of message.
4. An exit code of 0 should be specified for success, non-zero for failure. In the case of failure, the program may optionally write diagnostic information to STDOUT (e.g. with `printf` statements), which will be presented along with other information or error messages to notify the sender or Postmaster of delivery failure.
5. Any data files written to should either use unique filenames or use some form of file locking, as it is very possible for multiple delivery programs to be spawned and running simultaneously.
6. A member of the NT administrator's group should do an appropriate security check of the delivery program for their site (ideally check source for possible threats and compile and link it themselves from the checked source code), then copy it into the trusted directory with the following permissions (which are the default for files created in or copied into that directory):


```
Administrators:          Full Control (All)
PO Service Account:      Read (RX)
SYSTEM                   Read (RX)
```

The following code illustrates a simple mail archiving program, `pdarchiv`, written in C for NT platforms. The program accepts messages and appends them to a message file in the `C:\temp` directory.

```
/* pdarchiv - demo for Post.Office NT Program Deliver */

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <string.h>
#include <windows.h>

void main(int argc, char *argv[])
{
      char szLineBuf[1024];
      FILE *out;
      time_t rawtime;
      char szTime[32];
      HANDLE lock;

      /* loop until we obtain lock */
      while (1)
      {
            /* try to open lock file exclusive */
            lock = CreateFile("C:\\temp\\archiv.loc",
                  GENERIC_READ, 0, NULL, OPEN_ALWAYS,
                  FILE_ATTRIBUTE_NORMAL, NULL);

            /* check for archive file already open */
            if (lock == INVALID_HANDLE_VALUE)
            {
                  printf("sleep\n");
                  /* another copy is running, take a quick nap
                     and try again */
                  Sleep(1000);
                  continue;
            }

            /* else we have the lock */
            break;
      }

      /* try to open archive file in append mode */
      out = fopen("c:\\temp\\archive.txt", "a+");

      /* create timestamp */
      time(&rawtime);
      strcpy(szTime, ctime(&rawtime));
      szTime[strlen(szTime)-1] = '\0';

      /* write separator */
      fprintf(out, "---- START NEW MESSAGE --- %s ------\n", szTime);

      /* now read and log what comes in our STDIN - should be the
         message file */
      while (1)
      {
```



```
                /* Read line from standard input */
                if (gets(szLineBuf) == NULL)
                      break;

                /* write it to archive file */
                fprintf(out, "%s\n", szLineBuf);
        }

        /* one more blank line */
        fprintf(out, "\n");

        /* close archive file, which frees it for other
           copies of pdarchiv */
        fclose(out);

        /* release the lock */
        CloseHandle(lock);

        exit(0);
}
```

Remember, a program used with Program Delivery must exit with the value of 0 (zero) to indicate successful delivery. If the program exits with any other value, Post.Office will assume that an error occurred in the program and that delivery of the message was not completed. Such failed deliveries will cause the message to be held for subsequent delivery attempts, and the Postmaster will be notified.

Anything written by the program to STDOUT will be displayed in the diagnostic message sent to the Postmaster in the event of failed delivery. This allows the program to report specific errors to the user.

6.3 UNIX Program Delivery

Configuring Program Delivery on UNIX platforms is a little more involved than it is on NT, but the principles are the same: for each account using this mode of delivery, you provide the login name of the UNIX account that will run the program, as well as the program itself and any required parameters. Unlike NT, however, UNIX Program Delivery involves two modes of operation which differ substantially in their security implications, which you absolutely must understand before attempting to use this feature.

Program Delivery can be globally disabled by the presence of a file in the trusted program directory named `NO-PROGRAM-DELIVERIES`. This file is installed to the trusted directory with Post.Office, so Program Delivery is initially disabled on UNIX platforms. Remove (or rename) this file to enable Program Delivery.

6.3.1 The Two Modes of UNIX Program Delivery

The UNIX program delivery module in Post.Office can operate in one of two modes, depending on the level of security desired. The module determines which operating mode

to use by checking for programs in the trusted program directory. If none are found, the system operates in *open* mode, which allows users to run any command on the system.

If there is at least one file in the trusted program directory, the system runs in *secure* mode, which restricts users to running one of the trusted programs. Since only the system administrator (i.e. root) of the machine is allowed to add or remove trusted programs, the secure mode is indeed very secure.

When the program delivery module is set up to run in secure mode, it can be as secure as you need it to be. Since the trusted programs are the only programs on the system that it will run, regardless of how accounts are set up, the security vulnerability of a system running Post.Office is limited to this small collection of programs.

Again, programs that interpret their input as a sequence of commands (such as shells like sh and csh, or scripting languages like perl and tclsh) should not be set up as trusted programs.

Secure Mode

The following algorithm is used to deliver mail to a user with a valid shell when Post.Office is set up in secure mode:

- Post.Office sets up a restricted environment consisting only of the variables TZ and AGENT.
- Post.Office permanently gives up root permissions by changing to those of the controlling user (using `setuid(2)`).
- Post.Office switches to the controlling user's home directory if possible (remaining in `/tmp` if a failure occurs).
- Post.Office performs two checks -- making sure there are no characters in the command that have special meaning to a shell,[33] and that the program to be run is a trusted program.
- Post.Office runs the trusted program (using `execve(2)`) without invoking a shell such as `/bin/sh`.
- Finally, Post.Office feeds the message to the running program.
- If the program exits abnormally or if any output is produced by the program, an error message is sent to the Postmaster.

Open Mode

Operating the program deliver module in open mode requires a lot of trust (perhaps too much) on the part of the system administrator.

---

33 There are 12 special characters: $ ^ & ( ) | ‘ ; < > CR and LF. So, for example, you will not be able to run two programs connected by a pipe when using Post.Office program delivery in secure mode.

The Postmaster(s) must be trusted not to set up accounts with improper system permissions, since they can assign an arbitrary UNIX login to any account. Such an account can then be used to run commands as the assigned user, provided the user has a valid shell.

More importantly, the networks to which you are connected must be trusted. This is certainly not the case if you are connected to the global Internet, where so many tormented souls regularly attempt to break into your system that the secure mode should be used. If one of these system crackers were to compromise a Post.Office system, they might be able to set up accounts to provide them with privileged information, or to damage mail files. Of course they will never gain root privileges since no external program is ever run as root.

When opting for the open mode of operation, the system administrator can (and should) take precautions that minimize the risks. First of all, choose the Postmaster carefully. If nobody is qualified and responsible, the best choice is to do it yourself or set up Post.Office to run in the secure mode. Secondly, set up special accounts such as `bin`, `sys`, `adm`, and so forth with shells that are not valid for delivering mail to programs. (note that leaving the shell field blank does not accomplish this since a default of `/bin/sh` is assumed.) In the open mode of operation, it is especially important not to override the checking of valid shells in `/etc/shells`.

The following algorithm is used by Post.Office when delivering mail via the program delivery facility to a user with a valid shell:

- Post.Office sets up a restricted environment consisting only of the variables TZ, PATH, and AGENT.
- Post.Office permanently gives up root permissions by changing to those of the controlling user (using setuid(2)).
- Post.Office switches to the controlling user’s home directory if possible (remain in `/tmp` if a failure occurs).
- Post.Office runs `/bin/sh` with the command line specified in the account.
- Post.Office feeds the message to the running program.
- If the program exits abnormally or if any output is produced by the program, an error message is generated.

6.3.2 Configuring Post.Office for Program Deliveries

The following instructions explain the steps that must be performed by a system administrator to enable program deliveries on UNIX platforms. It is important that the system administrator understands all of the implications of installing any setuid-root program (that is, a program that acquires root permissions when it is run) on the system. Due to the security issues involved, the program delivery module is disabled by default and must be activated explicitly.

The commands shown in the examples assume that the executable programs have been installed in `/opt/Post.Office`. In the executable directory are several sub-directories, including `/local/` and `/trusted/`, which are where the program delivery module and the trusted program directory, respectively, are located. If you do not have root privileges on the machine running Post.Office, you will have to find a system administrator to do these steps for you.

Enabling the Program Delivery Module

The program delivery module can be activated by performing two simple steps as root. The resulting mode of operation is the open mode, so further configuration is required to set up the secure mode (which is *highly recommended* for most situations) with a list of trusted programs (see below).

Whenever the program delivery module finds a file in the trusted program directory named NO-PROGRAM-DELIVERIES, it refuses to deliver mail to any program. If an account is configured to deliver mail to a program, Post.Office generates a verbose error message to the Postmaster.

You must remove this file for program deliveries to work, and can create a new one at any time to disable the feature.

```
cd /opt/Post.Office/trusted
rm NO-PROGRAM-DELIVERIES
```

Note: *The file name must be typed exactly as shown – in all capital letters with dashes – in order to disable the program delivery feature.*

In order to run programs as a controlling user, the program delivery module needs to be setuid-root. If the setuid-root permission bit is not set, messages destined for users' programs are deferred until either the setuid bit is enabled, or the maximum queue time expires and the message is returned to the sender.

```
cd /opt/Post.Office/local
chmod u+s Program-Deliver
```

Set up the Trusted Program Directory

If you want to set up Post.Office to run in the secure mode, you must set up some trusted programs. To do this, simply copy each program to the trusted program directory, or create a link in the directory to the program. This short example shows one way to set up a program called `mail-filter` as a trusted program:

```
cd /opt/Post.Office/trusted
ln -s /usr/bin/mail-filter mail-filter
```

Note: *It is important to remember that programs that interpret their input as a sequence of commands to execute (such as `sh`, `tclsh`, or `perl`) should not be set up as trusted programs. However, some scripts that run under such programs may be considered safe after careful inspection.*

Set Up the List of Valid Shells

If you want to allow users with login shells other than `sh`, `csh`, or `ksh` to use the program delivery feature, you need to set up `/etc/shells`. Simply list all of the allowed shells, one per line, complete with their full pathnames. Note that if you are creating the `/etc/shells` file for the first time, you need to include entries for any of the six default shells that you wish to allow. Here's an example of a possible `/etc/shells` file:

```
% cat /etc/shells
/bin/csh
/bin/ksh
/bin/tcsh
/usr/bin/csh
/usr/bin/ksh
/usr/bin/tcsh
%
```

It is possible (though generally not advisable) to override the valid shell check by putting an entry in `/etc/shells` that contains the character sequence `MAIL/ANY/SHELL`. This may be used to allow any user to use the program delivery feature of Post.Office, but to restrict user access to FTP (which also does a valid shell check). An example `/etc/shells` file for this purpose might look like this:

```
% cat /etc/shells
/bin/ksh
/usr/bin/ksh
/MAIL/ANY/SHELL/
%
```

Setting Program Delivery Options

When delivering mail to a program, the permissions acquired by Post.Office are those of the UNIX login specified in the account. To be safe, no program will be run as root even if it is specified as the UNIX login for the account. Instead, a specific user ID and group ID will be assumed when delivering mail for a root user. These IDs are specified in the UNIX Delivery Configuration Options Form, which is invoked from the **Set Special Delivery Configuration for UNIX** link of the System Configuration menu.

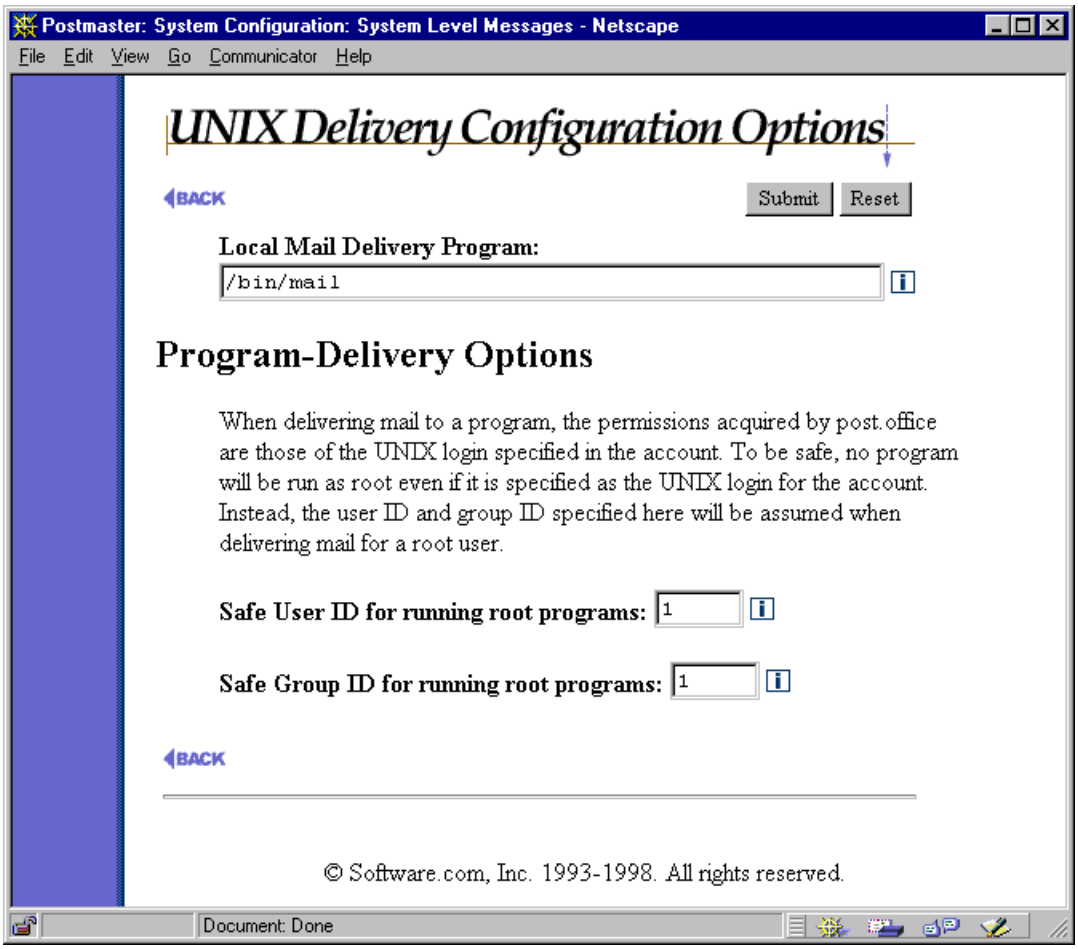

Figure 6-2: UNIX Delivery Configuration Options Form

Enter a user ID and group ID in the **Safe User ID** and **Safe Group ID** fields, respectively, and submit the form. These values are used for checking user and group permissions when the program invoked by Program Delivery is executed.

Disabling Program Delivery

You can disable the program delivery module simply by replacing the NO-PROGRAM-DELIVERIES file (which must be all capital letters with the dashes as shown). So long as this file remains in the trusted program delivery, Post.Office will not deliver any mail to programs. To do this, simply use the following commands:

```
cd /opt/Post.Office/trusted
touch NO-PROGRAM-DELIVERIES
```

Note: *Again, the file name must be typed exactly as shown – in all capital letters with dashes – in order to disable the program delivery feature.*

6.3.3 Enabling Program Delivery For an Account

As on NT platforms, setting up Program Delivery for a UNIX-based account requires you to enable this option in the Account Data Form and specify the program to be run. The following illustration shows a snippet from the Account Data Form for UNIX platforms which includes the relevant fields:

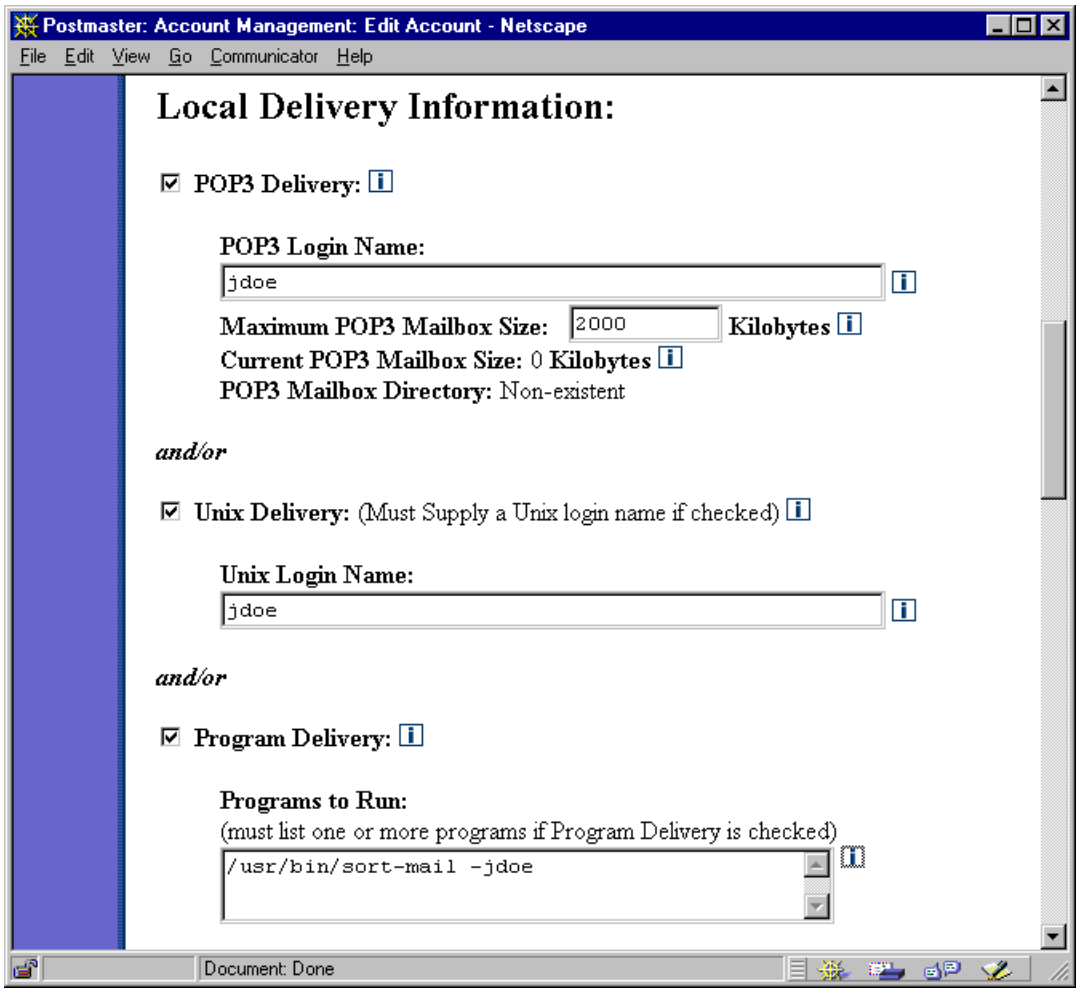

Figure 6-3: Account Data Form on UNIX platforms (delivery section)

Enable the check box labeled Program Delivery to set this option for the account. You must also provide a **UNIX Login Name** in the field above, and one or more **Programs to Run** in the field below. The program entry (or entries) must include any and all arguments required for execution.

Note: *Although the UNIX login name specified on this form is also used for the UNIX Delivery option, this name is not required to be the UNIX username of the person who receives mail through this Post.Office account (for security reasons, this may even be undesirable). This means that you can create a special user account on the server which is used only for invoking programs used with Program Delivery.*

When mail arrives to an account using this delivery method, the specified program will be run as the UNIX user specified in the **UNIX Login Name** field. If the user has sufficient permissions to do this, it's then up to the program to read the message from standard input and process it correctly.

7

Mailing Lists

This chapter contains information regarding the mailing list manager portion of Post.Office from the Postmaster's perspective. Included in this chapter are the following topics:

- An introduction to mailing lists and how they fit in with Post.Office.
- The attributes and characteristics of a mailing list.
- Instructions for creating, modifying, and deleting mailing lists via the web interface.
- Instructions for Postmaster moderation of a mailing list.
- Descriptions of the portions of the Post.Office user interface made available to local users, list owners, and remote users.
- An overview of the e-mail interface to list manager functions made available to list owners and end users.

Because the Postmaster is only one piece of the user puzzle, several areas of the end user operations in the mailing list manager are not fully described here. The *Post.Office User's Guide* contains detailed information on the end user operations and forms that are used to subscribe to a mailing list, submit messages to a mailing list, and unsubscribe from a mailing list. The *Post.Office List Owner's Guide* contains information for list owners to assist them in setting up and managing their mailing lists. You should refer to these other two manuals for specific instructions on end user and list owner activities. If you intend to personally manage one or more mailing lists, you should have your own copy of the *List Owner's Guide* handy.

7.1 Introduction to Mailing Lists

A *mailing list* is a group of users who share information on a common topic. Mailing lists allow electronic messages to be distributed to all of the list's *subscribers* by submitting a message to a single address. When an e-mail message is sent to the list address, it is forwarded to all subscribers of the list, each of whom receive a copy of the original message with their regular e-mail. Mailing lists are administered by one or more *list owners*, who are responsible for the day-to-day operation of the mailing list.

Mailing lists are similar to bulletin board services (BBS) and the Internet's USENET newsgroups, except that they use the medium of e-mail. A mailing list can be used for something as important as a list of all your company's employees, or as trivial as a television sitcom fan club.

A typical application for a mailing list is the creation of an employee list. For example, the MegaHuge, Inc. company might have a mailing list that includes all employees of the company, and which has the address `all@megahuge.com`. When a message is sent to this address, it is forwarded to all employees of MegaHuge.

7.1.1 Who Does What With Mailing Lists

Working with the mailing list manager is similar to using the account management portion of Post.Office, but offers substantially different operations to different classes of users. Whereas account management operations are divided between the Postmaster and the users who have local e-mail accounts, operations in the mailing list manager involve four classes of users: the Postmaster, list owners, users with local e-mail accounts, and users without local mail accounts (that is, the rest of the known world). Each of these four classes of users have access to a specific portion of the mailing list manager, and it's pretty important for you to understand what all of these users can do.

Note: *Obviously, these roles are not mutually exclusive, and one person can act as any or all of these types of users. For instance, all list owners are by definition local users. Also, by accessing the Public Mailing Lists area of the Post.Office web interface, local users can act like remote users.*

Postmaster

Make no mistake about who's in charge of the mailing list manager: as the Postmaster, you still have the ultimate say in what goes on the system. However, since you probably can't (and don't want to) personally maintain each and every mailing list, the system has been set up so that you can easily delegate that authority to one or more mailing list owners. These list owners, as described in the next section, have a broad range of options that they can set for their mailing lists, but only you can create lists, assign owners, and set limits for the amount of mail activity allowed for each mailing list.

These are the activities that the Postmaster – and *only* the Postmaster – can do with mailing lists:

- Create mailing lists
- Assign and change mailing list ownership
- Set a limit for the number or subscribers allowed on each mailing list
- Set limits for the amount and size of mail that is accepted for each mailing list
- Set and change the e-mail addresses associated with each mailing list

As Postmaster, you can also perform all list owner operations available in the web interface, such as:

- Set policies for subscription, unsubscription, and posting
- Add or remove subscribers

- Approve or reject subscription and unsubscription requests
- Approve or reject submitted messages
- Edit submitted messages before they are posted
- Set the available delivery options
- Set a schedule for digest deliveries
- Delete the mailing list

It should be noted, however, that *none* of the list owner options available in the mailing list manager's e-mail interface are available to the Postmaster. This is because the e-mail forms for moderation are sent only to the owner of the mailing list, and the list owner's password is required information for processing any administrative request. If the list owner elects to handle all moderation via e-mail, this means that the Postmaster will have no ability to moderate subscription requests and messages, or modify submitted messages before they are posted to a mailing list.

Although restricting the Postmaster from operations available to your users may seem akin to letting the animals run the zoo, remember that the mailing list manager has been designed to allow the Postmaster pass the job of list ownership to somebody else. If you have problems with sharing and just want to keep complete mailing list authority to yourself, you are certainly welcome to give ownership of all mailing lists to yourself.[34] But doing so may severely cut into your Postmaster duties (you've been warned).

List Owners

List owners are a special class of users to whom the Postmaster has delegated authority over specific mailing lists. Any user who has an e-mail account on your Post.Office server is eligible to own any number of mailing lists, over which he/she has wide-ranging administrative authority, including the ability to define most mailing list attributes. In fact, with the exception of a few limits and security parameters, a list owner's powers over his/her mailing lists are equal to those of the Postmaster.

Among the tasks that can be performed by list owners are the following:

- Set policies for subscription, unsubscription, and posting
- Add or remove subscribers
- Approve or reject subscription and unsubscription requests
- Approve or reject posted messages
- Edit submitted messages before they are posted
- Set the available delivery options
- Set a schedule for digest deliveries

34 Your local e-mail account, that is; you *should not* assign list ownership to the Postmaster account.

Although the Postmaster can access all areas of list owner activity in the Post.Office web interface, the same is not true for the e-mail interface, in which only a list owner can execute specific commands. The e-mail interface operations that can be executed exclusively by the list owner include:

- Approve or reject subscription requests
- Approve or reject unsubscription requests
- Approve or reject messages
- Distribute the message digest (available only in the e-mail interface)
- Change the long description of the mailing list

Section 7.10.2 gives you a glimpse of the portions of the Post.Office web interface which are available to list owners. However, list ownership is such a substantial endeavor that there is an entire manual devoted to the subject: the *Post.Office List Owner's Guide*. Refer to this manual for detailed descriptions of all list owner operations.

Local Users

All users with e-mail accounts on your Post.Office server – known as *local users* – can easily interact with the mailing list manager through the same web interface that they use to manage their mail accounts. From this interface, all local users can do the following:

- Get a list of the available mailing lists on Post.Office
- View descriptions of mailing lists
- Request subscription to mailing lists
- View their current list of subscriptions
- Request unsubscription from mailing lists

The local user web forms for executing list manager operations are shown in Section 7.10.1. In addition to the web interface, an e-mail interface provides access to the same functionality described above, with the following addition:

- Request the list of subscribers for a mailing list (available only in the e-mail interface)

All local user operations related to the mailing list manager are described in the *Post.Office User's Guide.*

Remote Users

The biggest difference between the account management portion of Post.Office and the mailing list manager is that users outside of your system – literally anyone in the world with an Internet connection – can use it. While this may sound alarming, be assured that these outsiders are confined to a relatively modest and limited area of the interface, and can *never* modify any part of Post.Office. So relax.

A special part of the Post.Office web interface is provided for these "public" users – known as *remote users* – and allows them to carry out the following mailing list operations:

- Get a list of the available mailing lists on Post.Office
- View descriptions of mailing lists
- Request subscription to mailing lists
- Request unsubscription from mailing lists

In addition to the web interface, an e-mail interface provides remote users access to the same functionality described above, with the following additions:

- View their current list of subscriptions (available only in the e-mail interface)
- Request the list of subscribers for a mailing list (available only in the e-mail interface)

"Wait a minute," you're probably thinking, "these are exactly the same tasks that my local users can do." And you're right – anything that a user with a mail account on your system can do with mailing lists (short of owning a mailing list) can also be done by users from outside of your system. However, all mailing lists make a distinction in their subscription policies between local and remote users. By defining a subscription policy that refuses all subscription requests from remote users, you create a "private" mailing list that is never seen by, or made available to, the teeming masses.[35]

Section 7.10.3 gives descriptions and illustrations of the mailing list management interface available to remote users.

7.1.2 Warning: Use Mailing Lists Wisely

Generally, one message sent to a Post.Office user is one message – one copy of the message is received, and one copy is delivered to the recipient's mailbox. However, the same is not true when dealing with mailing lists, since a single message sent to a mailing list becomes many copies of the message, which are then delivered to many users. Although this gives mailing lists their usefulness, it comes at a price: instead of processing a single message, Post.Office must individually process perhaps thousands of copies of the original message, one for every subscriber. Not surprisingly, the performance of Post.Office – and the server on which it is installed – can be impacted if these mailing lists are abused.

But just how much of an impact? Well, it could be anywhere from "practically no effect" to "practically no mail service," depending on a very large number of variables, such as:

- The number of subscribers on the mailing list

35 Similarly, a mailing list can open up subscription to the teeming masses (remote users) but refuse all subscription requests from your own local users. Why you would ever do this is unclear, but you certainly can if you want. Refer to Section 7.3.3 for more information on the possible subscription policy combinations.

- The size of the message sent to the mailing list
- Whether the message will be distributed individually or as part of a large collection of messages (known as a *digest*)
- Your hardware configuration (number and speed of microprocessors, disk space, available memory, etc.)
- The number of other messages also requiring attention by Post.Office (including from other mailing lists)

The impact of a single mailing list message on Post.Office could even be "quite definitely no mail service" in extreme combinations of the above factors, so it cannot be stressed enough that mailing lists *must be contained* and kept from growing to the point where they become a severe problem. Each mailing list includes a series of limits which allow you to do just that, and setting these limits will be your primary method of containing them within acceptable bounds.

Although we wish we could give you an exact breakdown of the type of system strain you should expect given the number and size of mailing lists that you want to create, there are simply too many variables involved to do so. It is therefore up to you to decide how best to contain mailing lists on your particular installation of Post.Office. We recommend that you use these four simple rules:

1. Start small
2. Do the math
3. Use common sense
4. See rules 1-3 :-)

Each of these rules is explained in the following sections.

Rule #1: Start Small

The primary method available to the Postmaster for controlling mailing lists is through the setting of list limits. For each mailing list in Post.Office, the Postmaster can set the following limits:

- **Maximum Number of Subscribers**. The total number of subscribers that the mailing list can have.
- **Maximum Kilobytes Per Message**. The largest message (in kilobytes) that Post.Office is willing to accept for posting to the mailing list.
- **Maximum Messages Submitted Per Day**. The total number of messages that can be submitted to the list each day.
- **Maximum Total Kilobytes Submitted Per Day**. The total number of kilobytes from all the messages submitted to the mailing list during each day. This is the most effective limit for curtailing the total impact of the mailing list.

When creating a mailing list, you should set low values – such as the Post.Office default limits – for all of the above limits, and adjust them higher on a list-by-list basis. Unfortunately, this will involve some trial and error, and additional administrative tasks for you at first. But it can also make the difference in keeping your mail system functioning well, which is the important thing here.

Note: *Of course, we're assuming here that you want to create more than a handful of mailing lists. If only a few mailing lists are all that you need, obviously you can be looser with the above limits.*

Rule #2: Do the Math

Even if a Postmaster sets what he/she thinks are pretty strict values for the limits mentioned above, they may fail to add up the maximum potential load of the mailing list – that is, its cumulative impact on memory, storage space, and microprocessor load – to get a "worst case" estimate of its impact on server performance. A mailing list will rarely (if ever) reach this worst case, but this scenario must always be considered before creating a mailing list.

In the following example, a Postmaster named Susie wants to create two mailing lists for her users. Susie is using a mid-level NT server system with about 500 e-mail accounts (a very small installation). She has read the above warning, decides to be very cautious considering her modest hardware, and sets limits and policies for both of the new mailing lists as follows:

- She sets the Maximum Number of Subscribers limit to 100.
- She sets the Maximum Kilobytes Per Message limit to 100 kb, which means that only messages of this size or smaller will even be accepted for posting to these mailing lists. This prevents any big attachments files from being posted.
- Susie sets the Maximum Messages Submitted Per Day limit to 100, meaning that the mailing list will accept only an average of one message per subscriber per day.
- A limit of 100 messages (which have a maximum size of 100 kb, as defined above) per day means that there can be a maximum of 10 Mb worth of messages submitted to each mailing list per day (100 kb x 100), so Susie sets the Maximum Total Kilobytes Submitted Per Day limit to 10,000 kb.
- Since both of the mailing lists she is creating deal with low-priority information, Susie doesn't want messages from these mailing lists being sent at the same time as regular mail. She therefore decides to use the digest method of message delivery, which sends all mailing list postings to subscribers in one big message, instead of one-at-a-time as they are posted. She sets the digest schedule for weekly, which means that all of the mail posted to the list during the week will be distributed to subscribers each Sunday at midnight (typically the time of the fewest mail transactions).

- After the mailing lists are created, 100 of her users subscribe to each mailing list and for whatever reason begin posting messages like gangbusters.
- A week later, Susie comes to work on Monday to find that her mail server system has run out of disk space, has stopped accepting mail, and is generally in a bad state.

What happened? Susie didn't do the math. While the limits she set for the mailing lists may have looked good, when all of the possible load on her system is added up, it's obvious that the server never had a chance in the "worst case" scenario.

Let's review Susie's limits:

- Maximum Number of Subscribers: 100
- Maximum Kilobytes Per Message: 100 kb
- Maximum Messages Submitted Per Day: 100
- Maximum Total Kilobytes Submitted Per Day: 10,000 kb
- Digest Schedule: weekly

While the limit on individual message size is reasonably low, the limit on the total storage size of messages posted to the list (10 Mb) was way too high. But where Susie really got into trouble was in setting the digest delivery schedule for weekly; this means that all of the mail received by the mailing list each week is saved up into one large message, which has a maximum size of seven times the amount of mail received each day:

10 Mb/day x 7 days = 70 Mb

Although 70 Mb is a very large message, Post.Office can certainly handle it given enough time. But this huge message isn't being sent once to one person, it's being sent *to every subscriber of the mailing list*. So the total load on the system from this digest message is equal to the size of the message multiplied by the number of recipients:

70 Mb/subscriber x 100 subscribers = 7000 Mb

That's right – the server attempted to deliver 7 *gigabytes* of mail for one mailing list. And because Susie was running two mailing lists, the numbers for the entire system were twice as bad:

7 GB/list x 2 lists = 14 GB

Susie's server stopped accepting mail because, when the clock struck midnight on Sunday, it attempted to deliver and store *14 gigabytes of mail*, which it didn't have the disk space for. Even if it did, the transaction would have taken a *very* long time.

You can avoid Susie's fate when creating mailing lists in Post.Office by simply doing the math:

1. Multiply the number of subscribers that you're allowing by the maximum message size to determine the largest possible impact of a single mailing list message. If it's more than you're comfortable with, reduce one or both of these limits.

2. If a mailing list supports the digest method of delivery, determine the largest possible digest message that could be generated (70 Mb in poor Susie's case) and multiply that by the number of subscribers. If the possible digest distribution is too large, schedule digest delivery more often (several days during the week, or even several times each day) to reduce the size of each distribution.
3. Finally, add the maximum load of this mailing list to that of all of your existing mailing lists. This should prevent you from adding so many small mailing lists that they cumulatively clobber server performance. If adding the new mailing list would "max out" your server, either reduce its limits, scale back the traffic allowed to other mailing lists, or just don't create the new mailing list. Of course, if you want to support more mailing list traffic, you can also get better server hardware.

Rule #3: Use Common Sense

Okay, the example illustrated in Rule #2 is very extreme, and is given here more as an attention-getter than an indication of reasonable mailing list usage. Most mailing lists do not receive 100 messages daily, as in the case of Susie's mailing lists, since very few subscribers have the desire or time to read 100 messages every day (10 messages/day is considered an active mailing list). Also, any mailing list subscriber who uses a modem to dial in to his/her mail provider will not long tolerate a mailing list that requires them to download a 70 MB digest message (on a 14.4 kb modem, this would take more than one full day). So while such a mega-mailing list can certainly exist – and will certainly cause problems – it would also be *extremely* unpopular among subscribers and wouldn't be hosting such traffic for very long.

The best advice for working with mailing lists is the best advice for most things in life: use your common sense. For mailing lists, this means asking yourself such questions as:

- "Does this mailing list deal with critical information, like employee announcements, or is it a 'for fun' mailing list that isn't so important?"
- "How many messages a day does a subscriber to this mailing list really want to read?"
- "Are subscribers to this mailing list going to be exchanging large attachments?"
- "Will subscribers be getting their mail via modem? If so, how long would they be downloading messages from this mailing list?"

Use the answers to these questions to decide on the appropriate limits for each mailing list. Not only will these limits keep you happy by preventing the overloading of the server, it will also make subscribers happy by preventing the mailing list from dumping unreasonable amounts of mail on them.

Common sense is also required in deciding the number of mailing lists that you will create. Post.Office can support up to 30,000 mailing lists on a single installation, but the number of typical mailing lists that you can safely host depends largely on the configuration of the server system where you're running it. If you have a top-of-the-line UNIX server with dozens of microprocessors, an obscene amount of memory, and nearly unlimited storage capacity, then you can indeed host 30,000 mailing lists with

Post.Office.[36] However, if you're trying to squeak by on a single-processor NT system with 32 Mb of RAM, you simply don't have the resources required for that much system load.

Again, your best bet is to start small – create a few mailing lists and add more only after you have monitored your system's performance and are sure that you can support the ones that you've already got.

Rule #4: See Rules 1-3

Folks, mailing lists are fun and useful when they're kept under control ... but they have the potential to create severe system overload if left completely unchecked. Make sure you understand the potential problems before you start creating mailing lists. It really is that important.

7.1.3 Mailing Lists vs. Group Accounts

As discussed in Chapter 5, group accounts – that is, accounts which simply forward mail to a group of users – are similar to mailing lists. In fact, group accounts were used as "virtual mailing lists" by many Post.Office users before the arrival of the mailing list manager. However, group accounts lack most of the convenient features of mailing lists, and will likely be phased out by users in favor of the real mailing lists.

This does not mean that group accounts no longer have value – they do. In some cases, in fact, group accounts provide behavior preferable to mailing lists. You can use both to accomplish the same basic function – forwarding mail sent to a single address to multiple users – so the one you choose should depend on the additional features provided by each.

36 Assuming that all 30,000 mailing lists have reasonable limits and mail traffic.

The following table lists these similarities and differences between the two types of mail distribution mechanisms:

Feature	Group Account	Mailing Lists
Forwards mail sent to a single address to multiple users	Yes.	Yes.
Prevents users from receiving duplicate messages	Yes. This means that when a message is sent to both you and to a group account of which you are a member, you will receive only one copy of the message.	Yes, optionally. By default, a message sent to both you and a mailing list to which you're subscribed will be delivered to you twice: one copy from the sender, and one copy from the mailing list (which may be modified and/or delayed before it gets to you). However, all mailing lists include an option to suppress duplicates.
Requires management by the Postmaster	Yes. This can be a burden for the Postmaster to maintain.	No. The tasks related to administering a mailing list are passed off to one or more users, known as list owners.
Allows users to add and remove themselves from the list of recipients (subscribers)	No. The Postmaster must add or remove users.	Yes. Users can subscribe and unsubscribe themselves, or can be added/removed by the list owner and Postmaster.
Allows Postmaster to set limits on the amount of mail activity	No.	Yes. The Postmaster can set limits on the number of subscribers, the number of posted messages per day, and the size of posted messages.
Allows submitted messages to be approved or rejected (moderated)	No.	Yes.
Allows submitted messages to be modified before distribution	No.	Yes.
Allows requests for being added/removed from the list to be approved or rejected (moderated)	Yes, but not directly – a user can always send a request for group membership to the Postmaster, who can grant or ignore this request at their leisure.	Yes.

Feature	Group Account	Mailing Lists
Provides a regular statistical report on mail activity	No.	Yes, optionally. All mailing lists include an option for sending a daily report on list activity to the list owner.
Automatically sends messages to all users added to or removed from the list	No.	Yes. These messages can be defined by the list owner and can be disabled if desired.
Can have submitted messages delivered to a program or archive mailbox	Yes. Like any account, a group account can use the POP3 Delivery and Program Delivery options.	Yes, but not directly – this functionality can be achieved by subscribing an account which uses the POP3 Delivery and Program Delivery options.
Offers list members a choice of receiving messages immediately or in a group of messages	No. All mail to the group account is immediately distributed.	Yes. Subscribers can choose to receive messages immediately or as a group in a single message (called a *digest*)
When accepting messages for delivery, distinguishes between messages from members and non-members	No. Anyone and everyone can submit messages to the group.	Yes. List owners can close posting of messages sent to the list by non-subscribers (or even subscribers).
Allows users to get the list of individual group/mailing list members	No. The Postmaster must provide this information.	Yes. Users can request the subscriber list, which can also be secured and kept from public view.
Allows users to view descriptions of the group/mailing list	No.	Yes.

As you can see, there's a lot to consider when choosing between a group account and a mailing list. In most cases, a mailing list will be the method of choice. However, if you don't plan to really use any of the features of mailing lists beyond simply forwarding mail to groups of users, a group account will certainly be adequate.

7.2 The List Management Menus

To access the Postmaster's mailing list manager interface, log in to Post.Office as the Postmaster. Initially, you will see the Account Administration Menu. Click on the menu button at the left labeled **Mailing Lists** to display the Mailing List Administration menu.

Figure 7-1: Mailing List Administration menu

The menu contains four links, as well as a text field and execution button. These links and the forms that they invoke are described throughout this chapter. For now, the only option on this menu that we'll look at is the **List of Mailing Lists** link, which invokes the List of Mailing Lists menu. Similar to the List of Accounts menus, the List of Mailing Lists menu gives an alphabetical list of the mailing lists in the system.

Figure 7-2: List of Mailing Lists menu

All of the attributes of a mailing list are accessed through this menu, so whenever you want to add a subscriber, change a list policy, moderate submitted messages, or do anything else regarding an existing mailing list, this is where you start.

7.3 Anatomy of a Mailing List

Before you can start creating lists and assigning owners to manage them, you really do need to know what mailing lists can do, what types of options they can have, and what it all means for your job as the mail administrator. This section gives you a peek at the form for defining a mailing list, and walks you through each and every single field contained therein. The form is extremely long, so this will take a little while. Your best bet is to read over this section once and then refer back to it as you get deeper into this chapter and want to know more information about a particular option or field.

The following is the first of several illustrations of the Mailing List Data Form, which is used to modify existing mailing lists, create new lists, and set mailing list defaults. Don’t worry about how you get to this form or default values for the fields; all that is explained later in this chapter. For now, just concentrate on the purpose of each option.

Note: *For the sake of brevity, the screen shots and descriptions of this enormous form given in the following pages will not be repeated in this manual. This form is referred to repeatedly in Section 7.4, so put a bookmark, sticky note, or other appropriate signifier on this page for easy reference.*

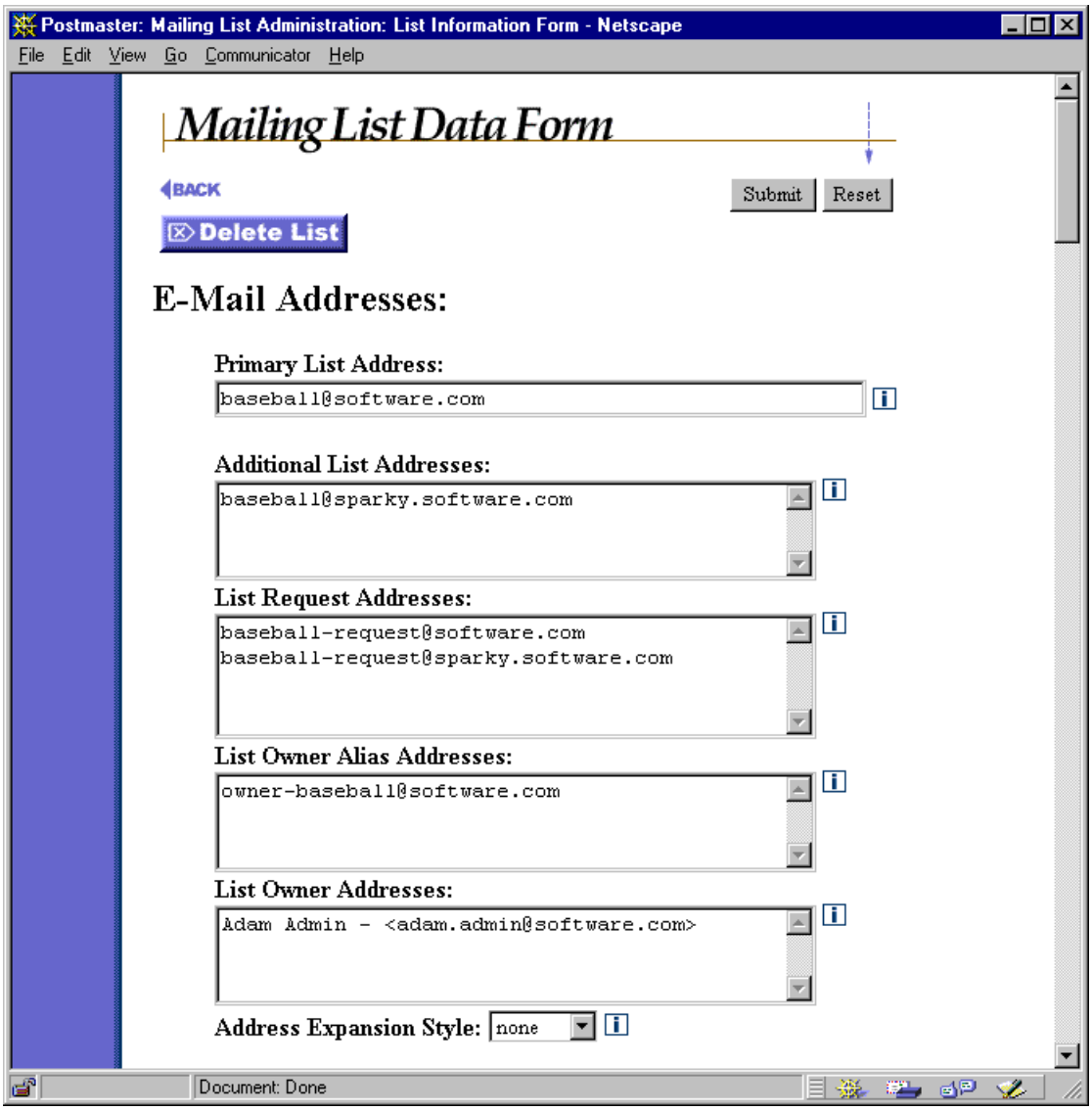

Figure 7-3: Mailing List Data Form (part 1 of 7)

7.3.1 E-mail Addresses

Like mail accounts, mailing lists have associated e-mail addresses. In fact, mailing lists have not one, but three different types of e-mail addresses. Every mailing list-related address must be unique across the system, so setting these addresses is your central task when creating a mailing list.[37]

Primary List Address

The most important mailing list address, this is where users will send messages that they want to post to the list. This is a required field for mailing list creation. You can use any format you want in creating a mailing list address, which need only be unique in the system, but for simplicity we recommend that make the local portion of the address identical to the List Name (located in another section of this form).

For example, for a mailing list with the List Name `employees`, a suitable address would be:

```
employees@host.domain
```

Again, this addressing format is simply a convention, and is not a requirement.

Additional List Addresses

Most mailing lists have only one posting address, but like user accounts, they can have any number of additional addresses, with each address being equally valid for the mailing list. These additional address can be useful if a mailing list needs to receive mail at several domains, or if the desired address format changes. Additional addresses are optional, so you need not enter values in this field.

List Request Addresses

This is the address(es) for the administrative e-mail account (also known as the request handler) that corresponds to the mailing list. This is a required field for mailing list creation. This account is responsible for sending welcome and farewell messages, sending and receiving verification tokens, and receiving e-mail forms and commands for the mailing list.[38] You can use any format you want in creating the Request Address, but again the suggested convention is to append "-request" to the local portion of the mailing list address.

---

37 Fortunately, Post.Office gives you a shortcut form for creating a mailing list which requires you to select only one address for the new list; the other addresses are then generated automatically. This form is described in Section 7.4.3, but *don't skip ahead yet* – you really should read through this entire introductory section before you go and create new mailing lists.

38 This is also the return address on the envelope for all messages sent out by the list, which allows returned messages to be safely handled.

For example, for a mailing list with the address

```
employees@host.domain
```

you would specify a Request Address of

```
employees-request@host.domain
```

List Owner Alias Addresses

These addresses provide a convenient way for users to contact the owner of a mailing list. This is a required field for mailing list creation. Each address specified here acts as a forwarding account; any e-mail sent to these addresses will be sent to all owners of the mailing list. They also provide anonymity for the list owner, who can give out this address to subscribers instead of his or her personal address, and allow list ownership to change over time without requiring users to learn a new address for contacting the owner.

Like other mailing list-related addresses, you can use any format you want in creating an owner alias address. However, we recommend inserting "owner-" before the local portion of the address to get the owner alias.[39] For example, for a mailing list with the address

```
employees@host.domain
```

you would specify an owner alias of

```
owner-employees@host.domain
```

Hint: *Keep your mailing list addresses simple, and don't change them unless absolutely necessary. These addresses may be published on web pages or be distributed by other mailing lists. Changing addresses after the mailing list is operational can cause great confusion among the user masses.*

List Owner Addresses

This is where you specify the local user(s) who will own the mailing list. Any user with an account in Post.Office can be a list owner, with each owner having equal authority over the list. This is a required field for mailing list creation. List owners cannot add an owner or change ownership of the list; only the Postmaster can set or modify this information.

39 Yes, we know that using "owner-listname" and "listname-request" is not exactly consistent. The reason that we recommend these address formats is that other mailing list manager programs use them, and we'd prefer that Post.Office fit in with the rest of the software world. Again, if you don't like these formats, you're welcome to use your own.

Address Expansion Style

This option controls the appearance of the **To:** line in the header of outgoing mailing list messages.[40] The list owner can modify this option, which can be used to distribute the subscriber list. Because of the privacy issues involved, this option could be considered a sensitive one.

The three choices for Address Expansion Style are **none**, **group**, and **expand**.

If **none** is selected, the destination address for mailing list postings is the primary list address. For example:

```
To: surfing@software.com
```

If the **group** expansion style is selected, the destination of the message includes the List Name of the mailing list, as well as the address of each subscriber to the list. For example:

```
To: surfing: jane.doe@software.com, joe.smith@software.com, ...
```

If **expand** is selected, the destination of the message includes the address of each subscriber to the list, but no mailing list address or List Name. For example:

```
To: jane.doe@software.com, joe.smith@software.com, ...
```

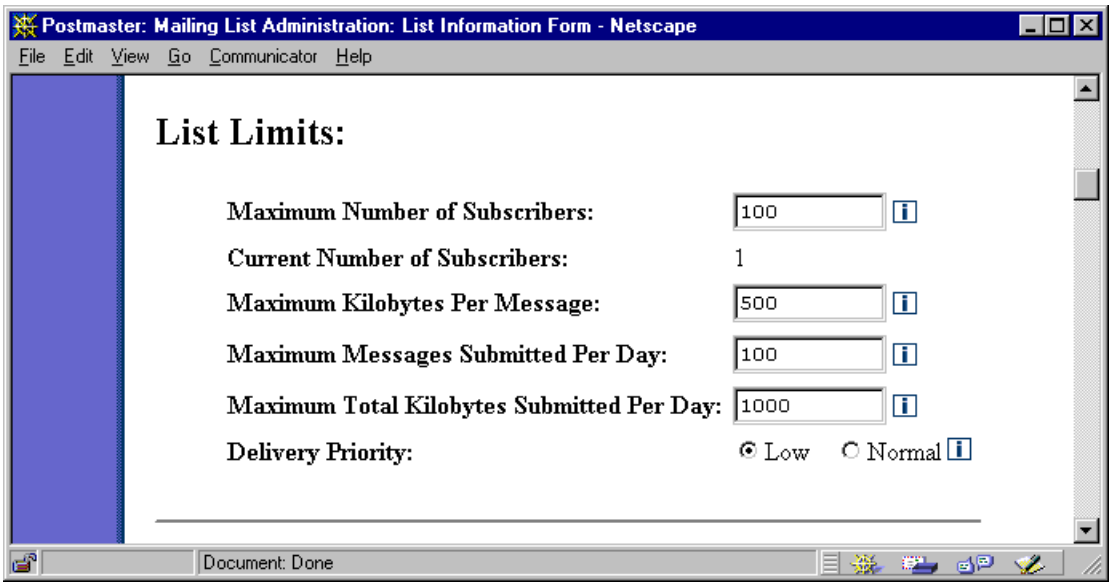

Figure 7-4: Mailing List Data Form (part 2 of 7)

40 This feature applies only to mailing list messages distributed with the immediate mode of delivery; messages distributed with the digest mode of delivery do not use Address Expansion. See section 7.3.6 for a description of these delivery types.

7.3.2 List Limits

As discussed in Section 7.1.2, the primary method available to the Postmaster for controlling mailing lists is through the setting of list limits. These limits allow you to put restrictions on the amount of e-mail that a mailing list is allowed to receive, and the number of subscribers that each mailing list can have. These limits can be an extremely important tool for are controlling large mailing lists that consume system resources.

The following limits can be set for each mailing list:

Maximum Number of Subscribers

This limit sets the total number of subscribers that the mailing list can have. Once it reaches this subscriber limit, any new subscription requests – submitted by individual users, the list owner, or the Postmaster – will be rejected. For convenient comparisons, the **Current Number of Subscribers** is displayed below this field.

Maximum Kilobytes Per Message

This limit determines the largest message (in kilobytes) that Post.Office is willing to accept for posting to the mailing list. Large messages can dramatically impact server performance, especially when they are distributed to a large number of subscribers. Any message submitted to the mailing list that exceeds this limit will be returned to sender.[41]

Maximum Messages Submitted Per Day

Post.Office keeps a count of the number of messages submitted to each mailing list each day.[42] This limit lets you specify the total number of messages that can be submitted to the list each day. If the number of submitted messages reaches this limit, any new messages submitted to the mailing list before midnight (when the daily counter is reset) will be returned to sender.

Maximum Total Kilobytes Submitted Per Day

Not only does Post.Office count the number of messages submitted to each mailing list, but it is also adds up the size of each message. As with the limit on the number of messages, if the total number of kilobytes from all the messages submitted to the mailing list today reaches this limit, any new message will be returned to sender.

---

41 If the system-wide maximum message limit is smaller than the message size limit for a mailing list, the system limit takes precedence. However, since mailing list postings will be distributed to many more users than regular messages, mailing list message size limits should be significantly lower than the system-wide message size limit.

42 Note that this is different from the number of messages *posted* to the list. If 100 messages are submitted, but the list owner rejects all of them and zero messages are posted, Post.Office still counts 100 messages against the daily total.

Delivery Priority

This isn't actually a limit, but it is another Postmaster-defined field which list owners cannot modify. This option determines the priority of e-mail sent out by the mailing list. The available values are **Normal** (processed the same as other messages) and **Low** (processed after all other messages). The recommended setting is **Low**, since it causes mailing list messages to be processed after all regular mail has been handled.

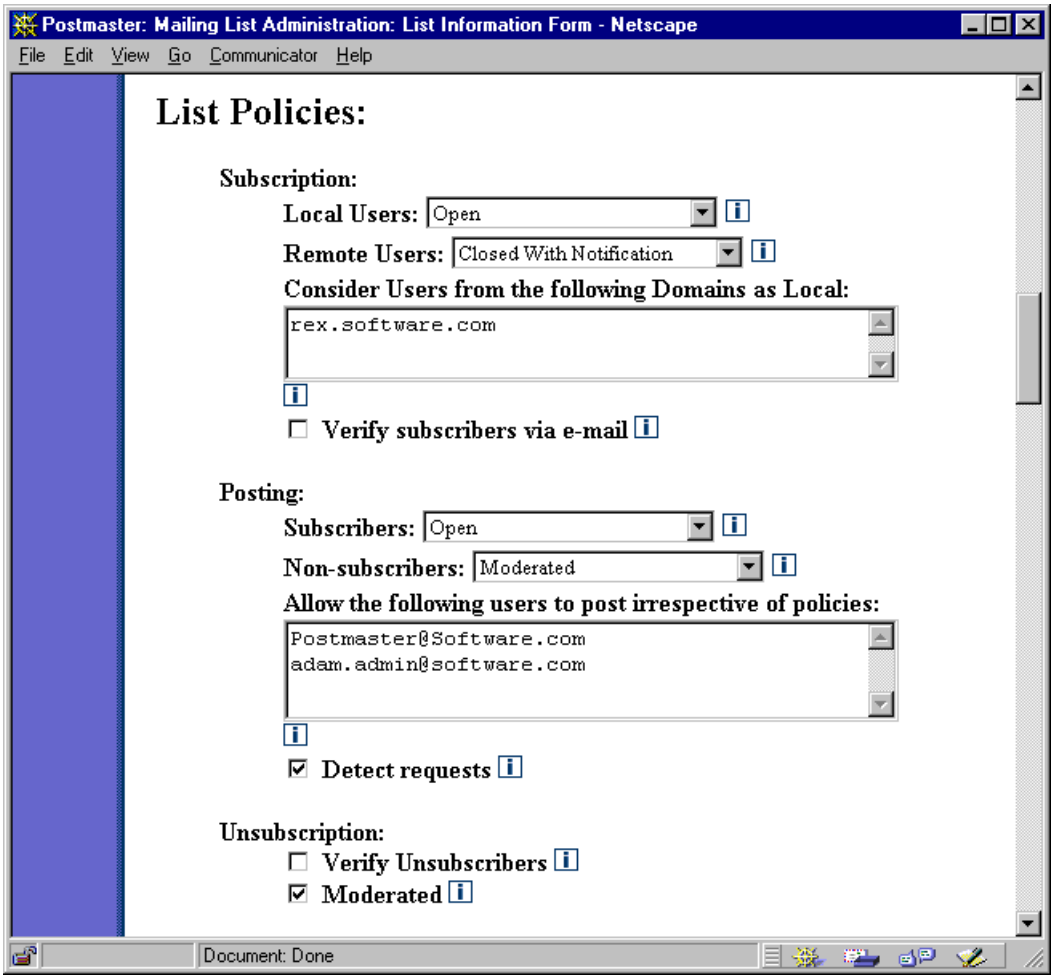

Figure 7-5: Mailing List Data Form (part 3 of 7)

7.3.3 List Policies

The setting of mailing list policies is the primary setup activity performed by list owners. Because these policies relate to the methods that the list owner will use to administer the mailing list, they are not as important to you as the Postmaster, so we'll assume that you won't be paying much attention to these options. However, you should still understand the available policies and what they mean for the system.

Subscription

These policies specify rules for the handling of subscription requests. The list owner can choose to deny all subscription requests, hold subscription requests for their own approval or rejection, or just open it up and let folks subscribe themselves. Furthermore, there are different policies for local users (those who have e-mail accounts in this Post.Office server) and remote users (the rest of the known world). Setting different subscription policies for these two types of subscribers is useful when mailing lists deal with potentially sensitive information, as in the case of a company's internal mailing list.

For both **Local Users** and **Remote Users**, the available subscription policy selections are:

- **Open**. This mode allows users to immediately subscribe themselves to the mailing list upon request.
- **Moderated**. This mode accepts the user's subscription request, but holds it for the attention of the list owner, who can then approve or reject it.
- **Closed with Notification**. This mode rejects all subscription requests, and notifies the list owner when a user unsuccessfully attempts to subscribe.
- **Closed without Notification**. Same as Closed with Notification, but does not send a notification to the list owner.

When the subscription policy for remote users is set to Open or Moderated, the mailing list becomes what is known as "public." The rather important implications of this term are discussed in Section 7.10.3.

Consider Users from the following Domains as Local. You can use this field to extend the subscription policy definition of "local" users to include some remote users by entering the domain of the chosen users here. Any remote user whose domain is listed here will enjoy the same subscription policy as folks who have e-mail accounts on your installation of Post.Office. For example, if this field contains the values

```
megahuge.com
dough-main.net
rover.software.com
```

then any user whose return address contains one of these domains – such as `john.doe@megahuge.com` – will have the same subscription rights to this mailing list as local users, even if they don't have accounts in your installation of Post.Office.

Verify Subscribers via e-mail. When enabled, this option will attempt to verify the identity of all attempted subscribers before submitting their subscription requests. The verification is done by sending a message containing a verification token to the user's e-mail address. By responding to the message and submitting the token, the user verifies that he/ she is who they claim to be.

This option is especially useful for public mailing lists, since users can claim to be anyone when requesting subscription. By verifying subscribers with this option, you can prevent mischievous computer users from submitting subscription requests on behalf of unsuspecting fellow commuters on the info autobahn.

Note: *Local users are exempted from verification if they submit their subscription requests via the Post.Office local user web interface, since these users "verified" themselves by logging in to the Authentication Form.*

Posting

As with subscription policies, there are two different posting policies for two different classes of users. With posting policies, the distinction is made between subscribers (official list members) and non-subscribers (the rest of the known world).

For both **Subscribers** and **Non-subscribers**, the available subscription policy selections are:

- **Open**. This mode immediately accepts messages and posts them to the list.
- **Moderated**. This mode holds messages for the attention of the list owner, who can view held messages, approve or reject them, and even modify the contents of a message before approving it for posting.
- **Closed with Notification**. This mode rejects all submitted messages, and notifies the list owner when messages are submitted and rejected.
- **Closed without Notification**. Same as Closed with Notification, but does not notify the list owner when messages are submitted.

Allow the following users to post irrespective of policies. As its label indicates, this field allows you to specify users who are allowed to post to the mailing list, regardless of policies. This means that you or the list owner can close posting to subscribers and non-subscribers alike, and reserve the ability to post to specific users (typically the list owner). To grant this "super poster" status to a user, enter his or her e-mail address in this field. When a message is submitted to the mailing list, it will be immediately distributed to subscribers if the return address of the sender is listed here.

***Note**: Because no password or security information is required for this feature, it's possible for someone else to post to a list by falsifying their return address to match one of the addresses given here. If you're looking for a 100% secure method of reserving posting only to yourself or the list owner, leave this field blank, set the posting policies to **Moderated**, and reject all postings but yours and the list owner's.*

Detect Requests. This posting policy allows you to filter out messages submitted to the mailing list which appear to be requests that were supposed to be sent to the list's request account. As discussed in the *Post.Office User's Guide*, users can subscribe and unsubscribe from a mailing list by sending commands via e-mail to the list's Request Address. However, some users don't quite understand this, and may mistakenly post e-mail commands to the mailing list, which can result in stacks of messages sent out to all of the list's subscribers that contain nothing but the word "subscribe."

When the humor of this cluelessness wears off and it simply becomes an annoyance, you can enable the **Detect Requests** option, which will check each message posted to the list to determine whether it contains e-mail commands. A message is considered to be a request if the body of the message contains three or less non-whitespace lines, and the subject or body of the message includes any of the following words: subscribe, unsubscribe, add, delete. If a message is rejected because it contained a request command, the message will be returned to the sender and the list owner will be notified.

Unsubscription

Unsubscription policies are similar to subscription policies, and include the following options:

Verify unsubscribers. As with the subscription verification option, enabling this field will cause Post.Office to verify the identity of all attempted unsubscribers before removing them from the subscriber list. The user's unsubscription request will be processed only after submitting the verification token via e-mail.

***Note:** Again, local users are exempted from verification if they submit their unsubscription requests via the Post.Office local user web interface.*

Moderated. Like the options for moderated subscription, this option will refer all unsubscription requests to the list owner for approval or rejection.

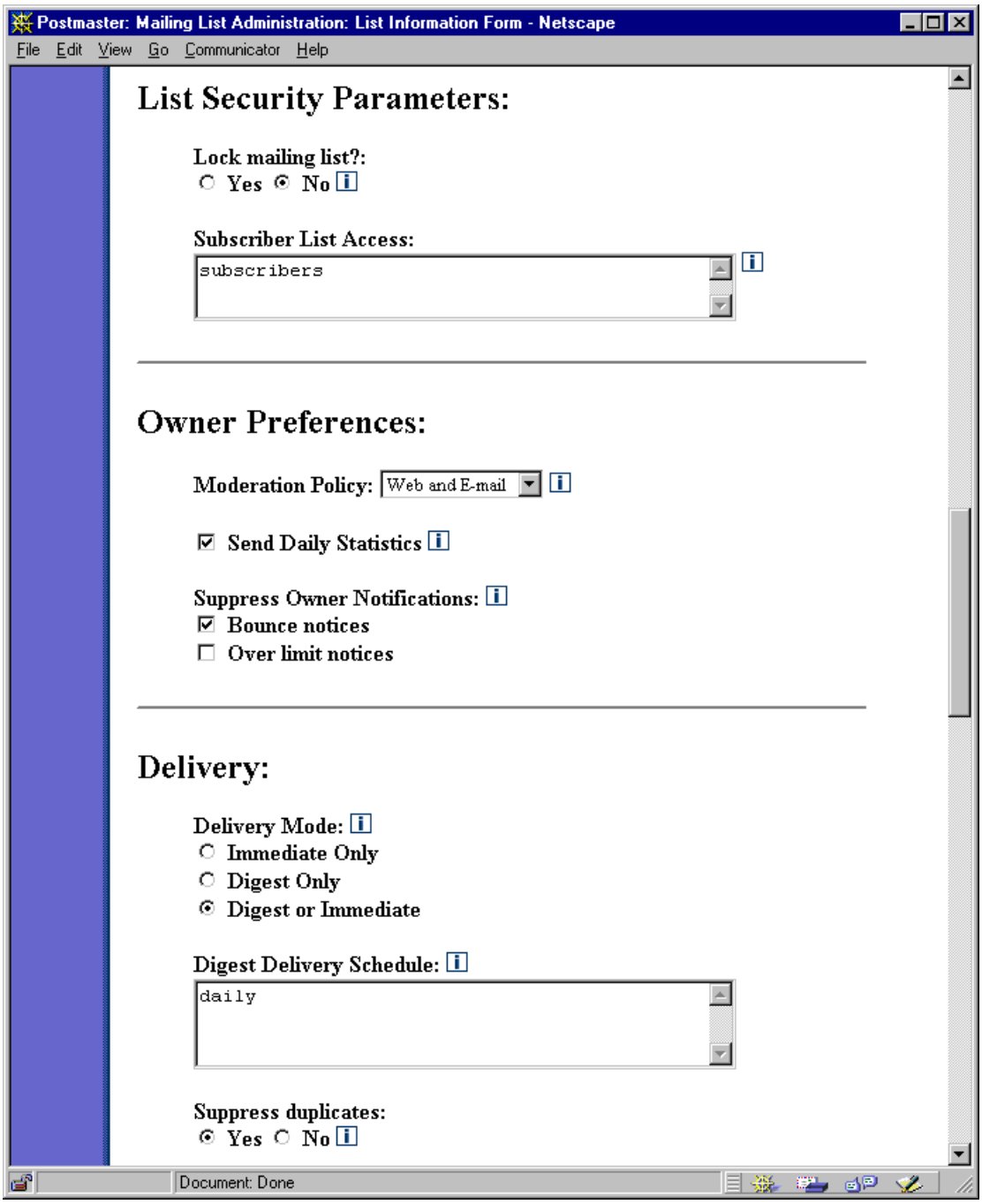

Figure 7-6: Mailing List Data Form (part 4 of 7)

7.3.4 List Security Parameters

The fields in this section of the form allow you to set security options for a mailing list.

Lock Mailing List

This field allows you to lock a mailing list, which suspends all list activity. When a mailing list is locked, no postings will be accepted by the list, and the list owner will be prevented from modifying or accessing the list in any way. To lock a mailing list, select **Yes** for this field.

See Section 7.7 for more information on locking a mailing list.

Subscriber List Access

The mailing list manager e-mail interface allows any user to request a copy of the list of subscribers for any mailing list. However, there are many cases in which this data may be considered sensitive and inappropriate for public access. The Subscriber List Access option allows you or the list owner to set rules for who can and can't use the e-mail interface to get the list of subscribers fore this mailing list.

The **Subscriber List Access** field is similar to the fields for setting account access and finger access rules, described in Chapter 5. The following algorithm is used to determine if a user can access the subscriber list by submitting the who command via e-mail:

1. If the field is empty, access is allowed.
2. If the keyword "none" appears in the field, access is denied.
3. If the keyword "subscribers" appears in the field, and the return address of the e-mail request is a subscriber to the mailing list, the subscriber list is sent to the user.
4. If the client's machine name is in the field, or is within one of the named domains, access to the subscriber list is allowed.
5. If the client's IP address is within one of the listed networks, access is allowed.
6. In all other cases, access is denied.

7.3.5 Owner Preferences

The fields in this section of the form determine how the list owner will interact with the mailing list.

Moderation

If the list owner decides to moderate postings, subscription requests, or unsubscription requests, this field specifies the method that they will use for that moderation. The available selections are:

- **Web and e-mail**. When this mode is selected, subscription requests, unsubscription requests, and/or messages will be held for list owner attention in Post.Office, and the list owner will be notified each time a new subscription request or message is submitted. These requests may be moderated by submitting e-mail commands, or by using the web interface. However, since subscription requests and/or messages are held indefinitely when this mode is selected, list owner action is required when this method of moderation is used.
- **Web only**. This mode is similar to the **Web and e-mail** mode above, but will *not* notify the list owner when new un/subscription requests or messages are submitted. This mode should be used only by very diligent list owners who will regularly log in to the web interface to perform moderation, since failing to do so will leave users or messages in perpetual limbo.
- **E-mail only**. Unlike the above modes for web moderation, this mode will *not* hold messages or subscription requests in Post.Office when submitted. List owners will be notified of new submissions, as with the **Web and e-mail** mode above, but will be required to submit the associated e-mail commands to approve the message or request. Selecting this method means that the system will *not* hold moderated messages, subscription requests, or unsubscription requests; they will simply be forwarded to the owner, who must personally submit them.

Note: *Because only the list owner can submit moderated messages, subscription requests, and unsubscription requests when the moderation mode is* ***E-mail only****, the Postmaster cannot moderate a mailing list that uses this method of moderation.*

Send Daily Statistics

This field controls whether the list owner will receive a daily report on the activity of the mailing list. This statistical report is sent daily at midnight. If your server hosts a large number of mailing lists, the cumulative impact of sending daily statistics messages to the owners of all of these lists may cause your mail server performance to suffer at midnight.

Suppress Owner Notifications

These fields control whether or not the list owner will receive certain notifications. By default, the owner will receive a notification from the mailing list whenever a list operation is disallowed because one of the list's limits have been reached. The owner will also receive a notice whenever a list posting is bounced (returned) by another mail server.

7.3.6 Delivery

The fields in this section of the form control options for the delivery of postings to the mailing list's subscribers.

Delivery Modes

This option determines the delivery types available to subscribers. The three choices are: Immediate only, Digest only, and Digest or Immediate.

The immediate mode of delivery is just that: immediate. When messages are posted to the list, they are immediately sent out to all of the mailing list's subscribers who have selected this delivery mode. This is great for important list postings, such as an announcement to all employees that paychecks have arrived. However, for more trivial mailing list postings – like the fourteenth message in a debate on whether ferrets or hedgehogs make the better house pet – the immediate mode of delivery is unnecessary, and you may find it annoying to have such messages trickling in one-at-a-time with the rest or your e-mail.

Enter the digest mode of delivery. The idea behind the digest mode is that users receive all messages from the mailing list for a certain time period in one great big message. All mailing lists that support the digest mode of delivery have a corresponding digest schedule, which defines the days and times that the digest is sent out. When the appropriate hour comes, all subscribers using this mode of delivery are sent a digest message that includes the contents of all of the messages posted to the list since the previous digest was sent. The most common digest schedule is daily at a specific hour, but the list owner can specify any days of the week, and any hours in the day, for their list's digest delivery.

Note: *The digest mode of delivery does not support attachments, so subscribers who use this mode of delivery cannot receive files attached to messages posted to the mailing lists. Also, digest messages do not use the Address Expansion feature described in Section 7.3.1.*

The **Delivery Modes** field includes the following options:

- **Immediate Only**. When only the immediate mode of delivery is supported, all subscribers are added to the list with this delivery mode, regardless of the mode selected when requesting subscription.
- **Digest Only**. When only the digest mode of delivery is supported, all subscribers are added to the list with this delivery mode, regardless of the mode selected when requesting subscription.
- **Immediate or Digest**. When both delivery modes are supported, users will get whichever delivery mode they requested when subscribing.

Note: *If you later change the delivery modes supported by a mailing list, the change will have no effect on current subscribers. This field only determines the delivery types available to new subscribers.*

Digest Delivery Schedule

If the digest mode of delivery is supported by a mailing list, the **Digest Delivery Schedule** field specifies the schedule for the distribution of the digest. By default, the digest will be distributed daily at midnight, but can be distributed at any number of days and/or hours during the week when you want delivery to take place.

Days of the week are specified in this field by entering the first three letters of the day in all lower-case type (for example, “tue” for Tuesday). Hours are specified as single digits, can include “a.m./p.m.” or “am/pm”, and can be given in 12-hour or 24-hour format. Minutes cannot be specified when setting a digest schedule, just days and hours.

For example, each of the following examples specifies a delivery time of Monday at 5:00 p.m.:

```
mon 5 pm
mon 5 p.m.
mon 17
```

In addition to specific days, the digest delivery schedule can also be given as daily or weekly. The daily option delivers the digest every day at the specified time, or at midnight if no time is specified. The weekly option delivers the digest each Sunday at midnight, or at the specified time. If the delivery schedule is given as a time with no day, the delivery schedule is daily at the given time.

Suppress Duplicates

This option provides a method for preventing users from getting multiple copies of messages sent to the mailing list. By default, if a message is sent to both an individual user and a mailing list to which that user is subscribed, the user will get two copies of the message: one from the sender, and one from the mailing list. By suppressing duplicates, you can prevent users from unnecessarily receiving multiple copies of a single message.

When duplicate suppression is enabled, users will get only one copy of a message even if the message is delayed or altered by the list owner. For this reason, suppression of duplicates may not always be desirable. Again, whether or not you use this option depends on the mailing list.

Note: *Even if duplicate suppression is turned on for a mailing list, some remote subscribers may still end up with multiple copies of a message. For example, if a subscriber is itself a mailing list on another mail server, Post.Office has no way of knowing which users are subscribed to the other mailing list. There's simply no way that one Post.Office can know enough about every other mail server in the world to ensure that duplicates never occur.*

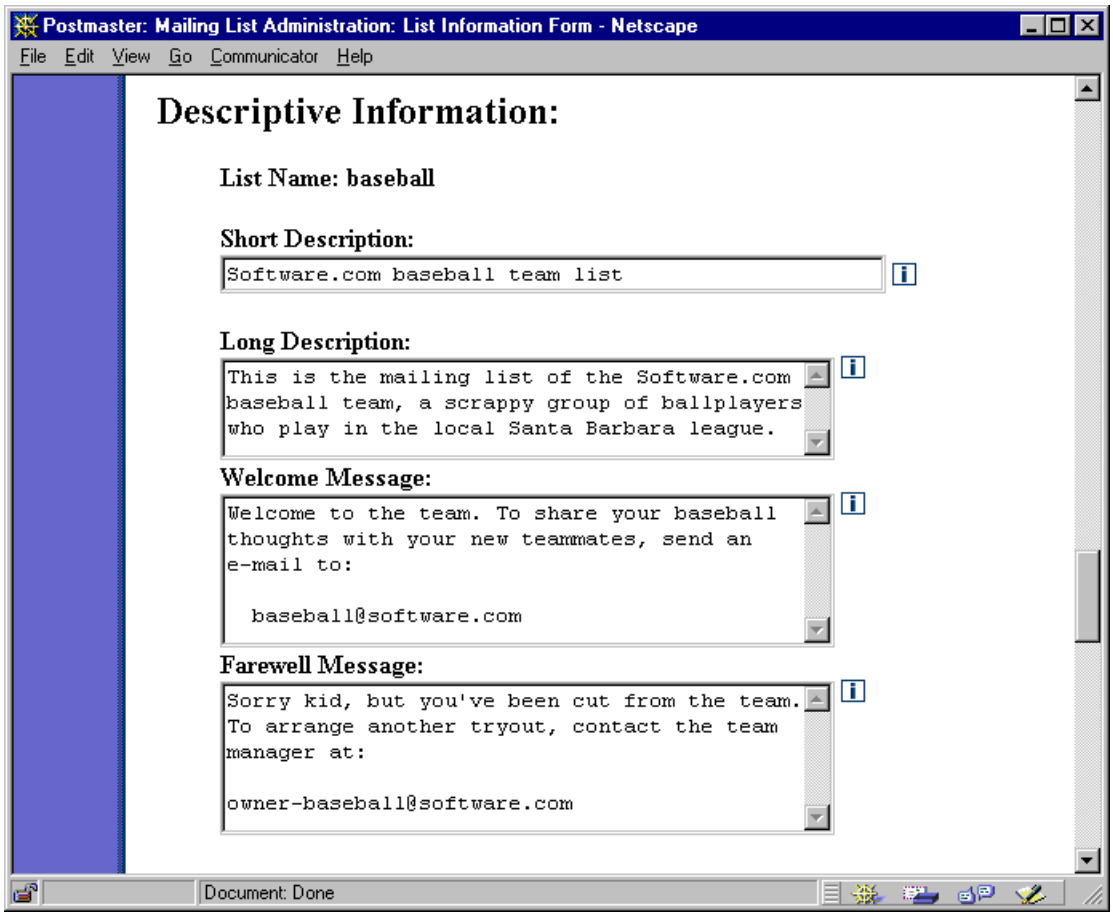

Figure 7-7: Mailing List Data Form (part 5 of 7)

7.3.7 Descriptive Information

There are several attributes of a mailing list that are intended to give users information regarding the mailing list and what it's all about. Most of these are optional and do not require values to be specified. These descriptions will typically be set by list owners, who can modify any or all of these values at any time.

List Name

The List Name is a unique name that identifies the mailing list in commands submitted to the e-mail interface, and is one of the few required pieces of information that you must supply when creating a mailing list. It is shown in a Mailing List Summary Form in the end user web interface, but is otherwise used only when submitting list commands via e-mail. This attribute cannot be modified after a mailing list is created.

The List Name can include letters (A-Z, a-z) , numbers (0-9), and the addition (+), subtraction (-), and underscore (_) characters. The List Name cannot contain any spaces or non-printing characters.

Note: *The List of Mailing Lists menu (Figure 7-2) sorts its display of mailing lists by List Name. Furthermore, this sort is case-sensitive, which means that mailing lists with capitalized List Names will be displayed before mailing lists whose List Names begin with a lower-case letter. Consistency is therefore important when assigning List Names – use all capital letters, all lower-case letters, whatever works for you.*

Short Description

This is an optional short description or title for the mailing list. This description is displayed in several areas of the web interface for end users, and is used to give potential subscribers a better idea of the purpose for the mailing list. This field can be set by list owners and may include just about anything, or nothing at all. You can enter up to 80 characters in this field.

Long Description

This is another optional description of the mailing list. The long description is displayed to end users in a Mailing List Summary Form in the web interface, and can also be requested by users through the e-mail interface. Long descriptions generally include the posting address of the mailing list, a fuller explanation of its purpose, and an indication of the list policies for subscribing and posting. The list owner can modify this field from the e-mail interface, as well as the web interface.

Welcome Message

This is an optional greeting message which is sent to all new subscribers of the mailing list, regardless of how they were added to the subscriber list. Information typically included in the welcome message are things like a fuller explanation of list policies, an e-mail address for contacting the list owner, the schedule for digest deliveries, and some basic instructions for using the e-mail interface. If this field is left blank, no welcome message is sent to new subscribers.

Farewell Message

Similar to the welcome message, the farewell message is sent to users after they have been removed (voluntarily or otherwise) from the subscriber list. This message applies only to unsubscription operations, so if the mailing list is deleted by you or the list owner, no farewell message is sent; this allows you to exterminate resource-hogging mailing lists without creating a slew of new messages for the server to handle. If this field is left blank, no farewell message is sent to users who are removed from the subscriber list.

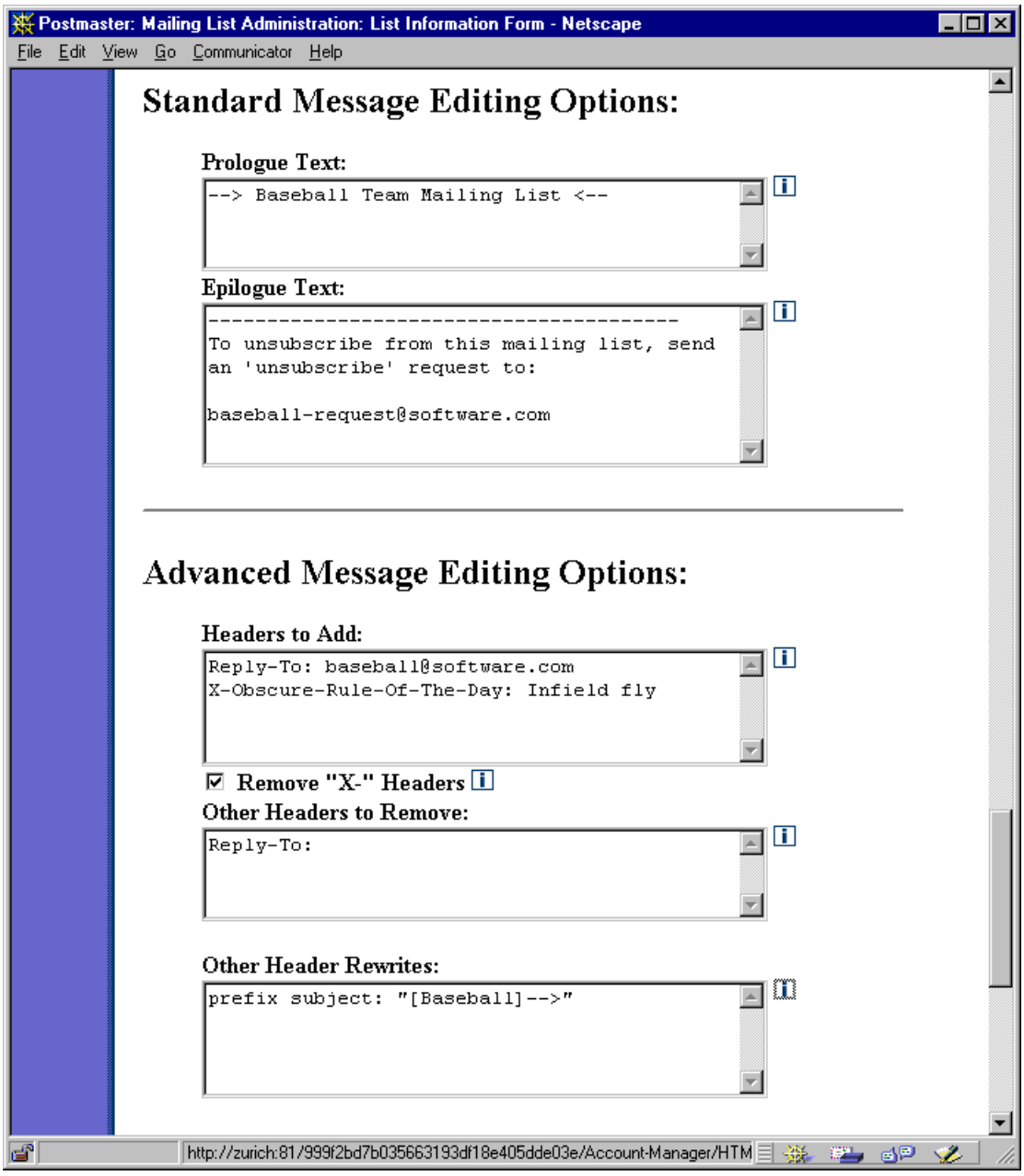

Figure 7-8: Mailing List Data Form (part 6 of 7)

7.3.8 Message Editing Options

Along with the basic policies and descriptions, mailing lists have a series of additional options for the automatic editing of mailing list postings. All of these options can be set and modified by the list owner.

Standard Options

The Standard Message Editing Options are used to specify text that will be inserted before and after the body of each message posted to the mailing list. The text in the **Prologue Text** field is inserted before the body of the message, and the text from the **Epilogue Text** field is inserted after the message body.

The prologue typically contains the List Name and/or short description of the mailing list, which marks the message as an "official" mailing list posting. The Epilogue typically includes an e-mail address for contacting the list owner, an address for submitting e-mail commands, and instructions for unsubscribing.

The following is a sample mailing list posting with a prologue and epilogue:

```
To: constitution@software.com
From: tjeffers@software.com
Subject: Preamble

----------------------------------------
CONSTITUTIONAL CONVENTION INTEREST GROUP
----------------------------------------

We, the people, in order to form a more perfect union, establish
justice, ensure domestic tranquillity, provide for the common
defense, promote the general welfare and secure the blessings of
liberty, to ourselves and our posterity do ordain and establish this
constitution for the United States of America.

- TommyJ

----------------------------------------
CONSTITUTIONAL CONVENTION INTEREST GROUP
----------------------------------------
To unsubscribe from this mailing list, send message containing the
word "unsubscribe" to:
   constitution-request@software.com

To get a list of valid e-mail commands and instructions on their
usage, send message containing the word "help" to the above address.

Send any problem reports or questions to:
   owner-constitution@software.com
```

In this example, the prologue – which announces the name of the mailing list – makes up the first three lines of the message body. The epilogue, which is inserted after the poster's original message and signature, includes the relevant addresses for the mailing list and instructions for unsubscribing.

Advanced Options

The options in this section pertain to the adding or removing of e-mail headers. All of these fields are optional, and should be used only by people who are well-versed with the standards for specifying optional e-mail headers.

Headers to Add. This field is used to add headers to each message posted by the mailing list. As with Prologue text, inserting a message header can be used as a way to mark "official" list postings. You can also use this field to specify a "Reply-To:" header if you want users to send responses to the mailing list (instead of the author of the message). The syntax for each header entered in this field must be valid according to RFC-821, such as the following:

```
Reply-To: surfing@software.com
X-Mailing-List-Manager: Post.Office
```

Remove "X-" Headers. When enabled, this option removes all message headers that start with "X-" from messages before they are posted to the mailing list. Some mail clients add this type of header to messages, which in rare cases can cause incompatibilities between mail clients.

Other Headers to Remove. Any other RFC-821 style headers specified in this field will also be removed from messages before they are posted to the mailing list. There is no wildcard matching in this field, so to be removed, a header must be identical character-for-character to a value in this field. For example, if you set a "Reply-To:" header in the Headers to Add field above, you probably want to remove the "Reply-To:" headers added by various mail clients by setting the following header to remove:

```
Reply-To:
```

Note: *Only the header itself – the part before and including the colon – should be entered here; don't include the information specified by the header (the text to the right of the colon).*

Other Header Rewrites. This field allows you to insert a prefix or suffix into the text of an existing header. Although all headers may be rewritten with this feature, it is only recommended for use with the subject header. By inserting some text before or after the original subject of a mailing list posting, you can allow subscribers to easily identify list-related messages and use mail filters to sort them.

To request header rewriting, enter the keyword `prefix` or `suffix`, followed by the header that you want to rewrite (i.e., Subject:), followed by the text of your prefix/suffix enclosed in "double quotes". For example, you might enter a prefix such as

```
prefix Subject: "[Surf list] "
```

or a suffix such as

```
suffix Subject: "-(Cycling list)"
```


When a header is rewritten, its original text is preserved just as the original sender wrote it; the text specified in this field is merely inserted immediately before or after the user-defined text. For example, using the rewrites specified above, a message with the subject “go this weekend?” would have the following respective subjects when sent to subscribers:

```
Subject: [Surf list] go this weekend?
Subject: go this weekend?-(Cycling list)
```

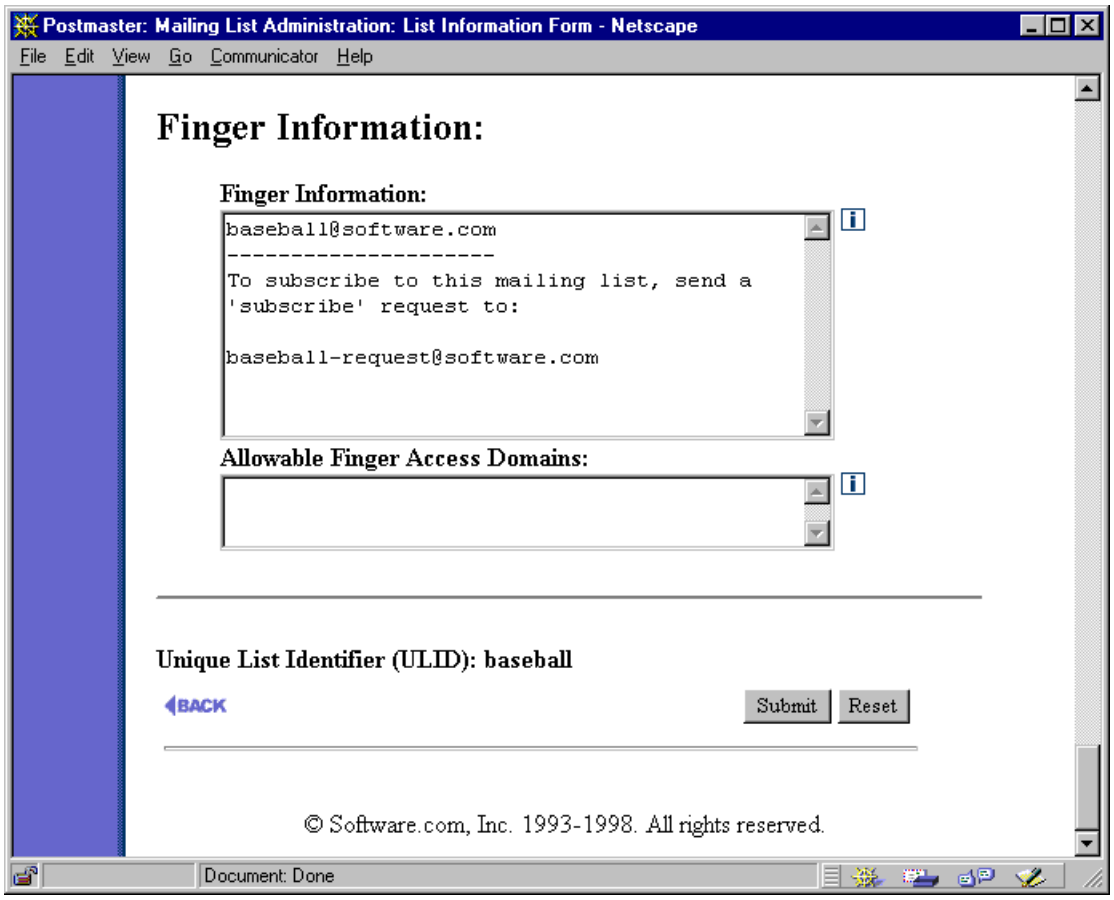

Figure 7-9: Mailing List Data Form (part 7 of 7)

7.3.9 Finger Information

As with e-mail accounts, mailing lists can have finger information associated with them. The following fields pertain to this information for the mailing list:

Finger Information

As in the Finger Information field in the Account Data Form, this field is used to specify the information returned by the finger service when a user requests this information for the mailing list. There is no limit on the number or type of characters that can be entered here.

Allowable Finger Access Domains

Like the finger access restrictions field in the Account Data Form, this field is used to restrict access to the mailing list's finger information by specifying the domains or IP addresses that are given access to view this information. The same rules that apply to accessing account finger information apply to mailing list finger information.

7.3.10 Unique List Identifier

The final piece of information on the Mailing List Data Form is the **Unique List Identifier (ULID)**. This value is similar to the Account Identifier (UID) described in Chapter 5, and is used with the Post.Office command-line utilities. The value of the ULID is based on the List Name, is set at the time of list creation, and cannot be modified.

Refer to Chapter 11 for information on using the ULID with mailing list-related command-line utilities.

7.4 Creating a mailing list

Now that you have a good understanding of the various bits and pieces of a mailing list, you are ready to start creating new mailing lists. This is the operation that you will perform the most often when working with the mailing list manager, since it's the one task that you cannot delegate to the owner of the list. For this reason, we've tried to greatly streamline the process of list creation, since an important administrator like yourself obviously has more significant and interesting things to do with your time.

Like mail accounts, the first step to speeding up the list creation process is the setting of default mailing list attributes, defined in the following section. Once you have established a good set of defaults, you can begin creating mailing lists in one of two ways: the long way, or the short way. These two methods of list creation are described in Sections 7.4.2 and 7.4.3.

7.4.1 Setting Defaults

The secret to streamlining the creation of new mailing lists, like new e-mail accounts, is to set defaults for any and every field. This is especially true for mailing lists, which have more than 40 attributes, only four of which must be unique across the system. By setting defaults for all of the other fields, list creation can be quick and painless.

The form for setting default mailing list values – like other mailing list-related web forms – is invoked from the Mailing List Administration menu. To refresh your memory, this menu is displayed when the **Mailing Lists** menu button is selected, and looks like this:

Figure 7-10: Mailing List Administration menu

To set mailing list defaults, click on the **Edit Default Mailing List...** link on this menu. This displays a Mailing List Data Form, like the one shown in Section 7.3. We won't subject you to another round of illustrations of this form, so just refer back to Figure 7-3 through Figure 7-9 if you want to see this entire form again.

What You Should Set

The most important mailing list defaults for you to set are the limits, which control the amount of activity that mailing lists are allowed. These fields are critical for controlling mailing lists and preventing them from impacting your server performance, so you should *never* create a mailing list without giving it some set of limits.

The following limits can be set for each mailing list. These limits can be viewed by the list owner, but can be set and modified only by the Postmaster.

- Maximum Number of Subscribers
- Maximum Kilobytes Per Message
- Maximum Messages Submitted Per Day
- Maximum Total Kilobytes Submitted Per Day
- Delivery Priority

The Delivery Priority field should, in most cases, be **Low**, so this is the recommended default for this field. This causes outgoing message from the mailing list to be processed by Post.Office only when messages of normal priority have been processed (that is, normal, non-mailing list messages).

As mentioned in Section 7.1.2, the setting of mailing list limits is extremely important. The amount of mail that is appropriate for a mailing list on your system depends on what type of system you have, how much memory is available, how much storage is available, how many users depend on your system for their mail, the reliability of your Internet connection, and about a million other variables. In general, you should start with low limits and adjust them up as needed.

What You May Want to Set

In addition to the important stuff, there are a number of mailing list attributes which certainly don’t require a default value, but for which you may wish to set one anyway. The first type of these are the fields which require values that are unique across the system:

- Primary List Address
- List Request Addresses
- List Owner Alias Addresses
- List Name

A mailing list cannot be created without a unique value for each of these fields. Since no two lists can have the same List Name, Request Address, etc., you can never save more than one new mailing list with the default value for these fields. However, you may want to use these default fields to specify some naming convention, which serves as a reminder for your preferred format for these mailing list fields. For example, if you want to follow the Post.Office convention of assigning mailing list addresses, you could set default values like the following:

Field Name	Suggested Value
Primary List Address	listname@host.domain
List Request Addresses	listname-request@host.domain
List Owner Alias Addresses	owner-listname@host.domain
List Name	listname

These defaults provide you with a template for specifying the addresses. Again, this is just a convenience that you can use if you find it helpful.

Another type of mailing list attribute for which you may want to set default values are those that define important policies. Although the list owner can modify these values, he/she may find it helpful if you provide some guidance on these particular items:

- Subscription Policy – Local Users
- Subscription Policy – Remote Users
- Consider Users from the following Domains as Local
- Allow the following users to post irrespective of policies
- Detect requests
- Delivery Mode
- Digest Delivery Schedule
- Prologue Text
- Epilogue Text
- Subscriber List Access

As always, you should choose defaults that best suit your particular environment, but the following are good guidelines for default values:

Field Name	Suggested Value
Subscription Policy – Local Users	Open or Moderated
Subscription Policy – Remote Users	Closed (with or without notification). This means that, by default, mailing lists will not be visible to remote users.
Detect requests	On
Consider Users from the following Domains as Local	Any domain besides your local mail domains from which users should be considered "local" by your mailing lists (for example, if you have a second mail server handling mail for other domains).
Allow the following users to post irrespective of policies	The Postmaster address for your site; this allows you to post to lists at any time.
Delivery Mode	Both digest and immediate
Digest Delivery Schedule	Something different from existing digest schedules
Prologue Text	Information that includes the List Name or short description, so subscribers realize that messages are from a mailing list.
Epilogue Text	Information that includes the Owner Alias Address, instructions for unsubscribing, and instructions for getting help through the e-mail interface.
Subscriber List Access	"subscribers". This allows only members of the mailing list to receive the subscriber list through the e-mail interface. Because of the privacy issues involved, this field may be a sensitive one, so select a default that reflects your organization's policies on these matters.

If you're looking for the ultimate in security, you should set default values for all posting and subscription polices to Closed with Notification. This means that, by default, new mailing lists are unusable, since nobody can subscribe to it or post messages to it. However, this also guarantees that the list owner will look over and change the policies for their mailing list before any mailing list activity occurs, which may be desirable for your organization.

Warning! It is important to avoid creating many mailing lists that have the same digest schedule. This can cause severe slowing of server performance at the appointed time.

Yet another type of mailing list attribute for which you should consider picking defaults are those which aren't all that important, but which are a tad complicated:

- Allowable Finger Access Domains
- Headers to Add
- Remove "X-" Headers
- Other Headers to Remove
- Other Header Rewrites

The *Post.Office List Owner's Guide* provides descriptions of these fields with instructions for using them. But let's be realistic – only one in ten computer users even knows what a manual looks like, so we can't assume that your list owners will take the time to learn about these options. It's probably safe to assume that list owners will change only the highly-visible mailing list attributes (like the Long Description and Welcome Message), and leave the rest of them alone. Therefore, if you want any kind of values for these fields, you should probably set defaults for them.

Finally, if you are designating mailing list administration to only a few people, you may want to set a default user e-mail address in the List Owner Addresses field. This allows you to automatically assign ownership to a specific local user, even yourself, if that's the way you choose to administer the mailing list manager.

What You Don't Need to Set

Many of the optional but mailing list-specific fields on the Mailing List Data Form – such as the Short Description or Welcome Message – will almost certainly be changed by the list owner at the earliest opportunity, so defaults for these fields are unnecessary. Obviously, you're welcome to set defaults for these if you like, and they can be useful for demonstrating to new list owners the type of settings that you recommend (for example, giving them welcome and farewell message templates, or selecting a particular group of policies that you favor for novices). But don't lose any sleep over these.

7.4.2 New Lists – the Long Way

Once you've set up whatever default mailing list attributes you'll be using, you can create a mailing list from the Mailing List Administration menu (Figure 7-10) by clicking on the **Create New Mailing List (long form)…** link. This displays a Mailing List Data Form, just like the one shown in Figure 7-3 through Figure 7-9.

Note: *Yes, we're starting you out with the longer of the two form-creation methods. Like those unforgettable high school algebra lessons, you should first understand the "real way" of creating a mailing list before you get to know about the shortcut (and then never use the original technique again). This is for your own good. Trust us.*

Required Fields

Initially, the Mailing List Data Form contains all of the default values that you specified. To complete the creation of the mailing list, you must set new values for the attributes that must be unique for each mailing list. To review, the following are the required fields:

Primary List Address. The address to which users send messages that they want to post to the mailing list.

List Request Addresses. The address(es) for the administrative e-mail account (also known as the request handler) that corresponds to the mailing list.

List Owner Alias Addresses. Addresses which forward messages to the owner(s) of the mailing list.

List Name. The unique name that identifies the mailing list in commands submitted to the e-mail interface.

Also, the List Owner Addresses field must contain at least one e-mail address for a user with an account in Post.Office. The other address fields can each be just about anything, provided they are all unique across the system and that all addresses fields contain valid SMTP e-mail addresses. For simplicity, as well as compliance with existing mailing list management programs, we recommend that you use the following address conventions (again, these are simply suggestions):

Field Name	Suggested Value
Primary List Address	listname@host.domain
Additional List Address	listname-list@host.domain
List Request Addresses	listname-request@host.domain
List Owner Alias Addresses	owner-listname@host.domain
List Name	listname

If you specified defaults for all of the other mailing list attributes, simply submit the Mailing List Data Form with new values for the above required fields to create the mailing list. That's it – you're done with this one. While the new mailing list will eventually be given specific descriptions, welcome/farewell messages, policies, etc., creating these for each mailing list is the list owner's job. You can certainly spend a lot of time setting up each new mailing list with specific information if you so choose, but you'll probably be a lot happier if you delegate these mundane tasks to the list owner.

7.4.3 New Lists – the Short Way

Although the process described in the previous section isn't very cumbersome, the Mailing List Data Form is indeed large and loaded with dozens of complex fields. For just this reason, we've given you a shortcut for creating mailing lists that allows you to specify only the few important mailing list attributes. The remaining attributes are either taken from the default mailing list values or are generated automatically, which can save you a significant amount of time when creating multiple mailing lists.

This shortcut is called the New Mailing List – Short Form, and is invoked from the Mailing List Administration menu (Figure 7-10) by clicking on the **Create New Mailing List (short form)…** link.

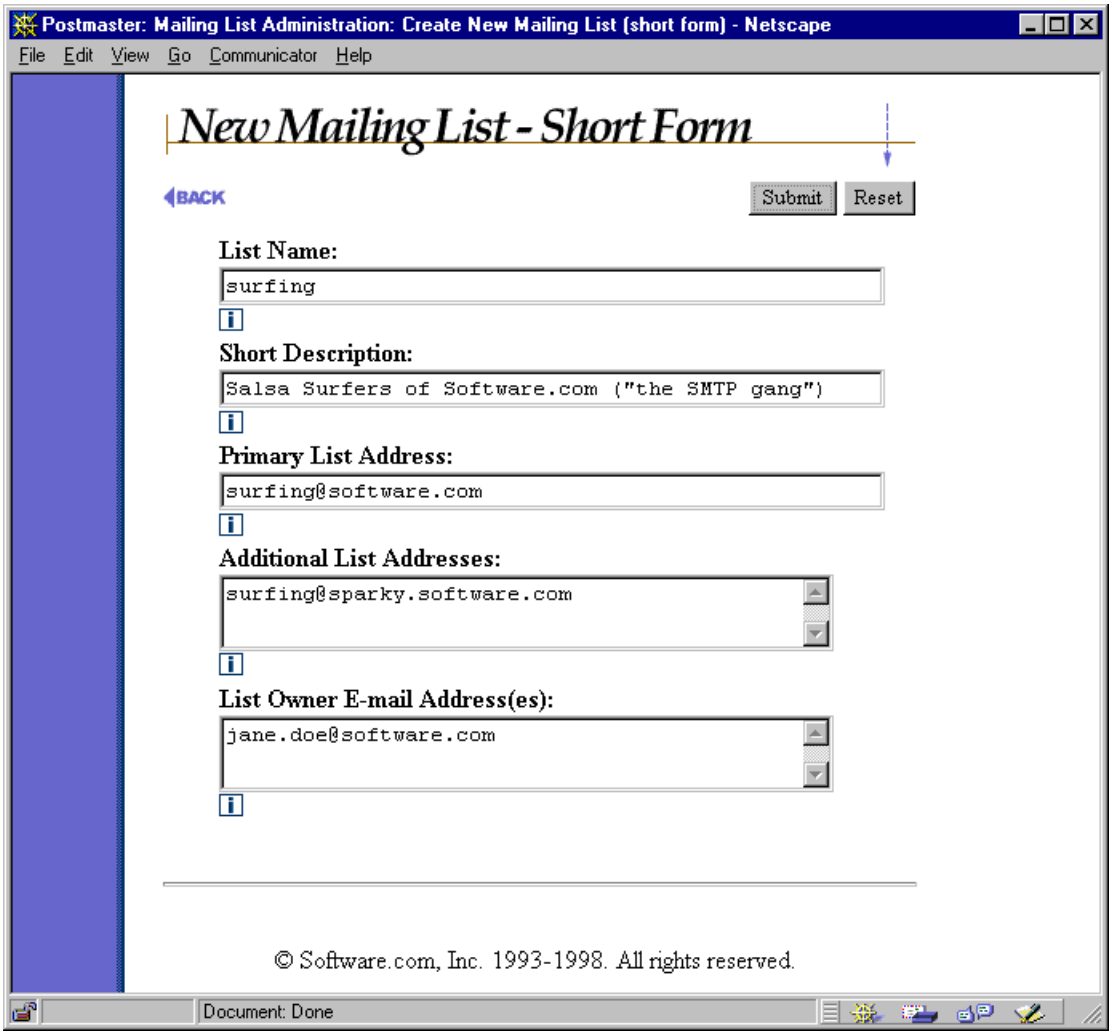

Figure 7-11: New Mailing List – Short Form

This shortcut form contains fields for the following mailing list attributes:

List Name. The unique name that identifies the mailing list in commands submitted to the e-mail interface.

Short Description. An optional short description or title for the mailing list. You aren't actually required to enter a value here, but the short description is shown in several areas of the web and e-mail interfaces.

Primary List Address. The address to which users send messages that they want to post to the list.

Additional List Addresses. Other valid addresses for posting messages to the list. Additional addresses can be useful if you use multiple addressing formats or multiple domains, but they are not required.

List Owner E-Mail Address(es). The owner(s) of the mailing list.

By filling in values for the above fields and submitting this form, you create a new mailing list that has these settings, as well as the default values you specified for the remaining mailing list attributes. An owner greeting form, like the one shown in Figure 7-12, is sent to the owner(s) of the new mailing list.

You may have noticed that two fields that were required for creating lists with the "long" form – the List Request Address and the List Owner Alias Addresses – are not available on the "short" form. How can this be, when these addresses must be unique throughout the system? The answer is that Post.Office generates these addresses for you, based on the posting address(es) that you supplied for the mailing list. The generated values follow the Post.Office convention of appending "-request" after the local portion of the list address to create the Request Address, and inserting "owner-" before the list address to create the List Owner Alias Addresses.

Values for both the List Request Address and List Owner Alias Addresses are similarly generated for each posting address that you provide (that is, the primary address plus any additional addresses). For example, if you entered the following information in the New Mailing List – Short Form:

```
Primary Address:      surfing@software.com
Additional Addresses: surfing@sparky.software.com
```

then the following addresses would be automatically generated when the form is submitted:

```
List Request Address:   surfing-request@software.com
                        surfing-request@sparky.software.com
List Owner Aliases:     owner-surfing@software.com
                        owner-surfing@sparky.software.com
```

You can verify that these addresses were created by bringing up the Mailing List Data Form for the new mailing list and inspecting the associated addresses. As with any mailing list attribute, you can always modify these addresses later if you want.

7.4.4 List Owner Greeting Message

When the mailing list is created, the local user (or users) who you dubbed as the list owner will receive a greeting message that announces the creation of the mailing list.

Note: *This greeting message is optional – you can determine whether the owners of newly created mailing lists receive it. This option is located on the Mail Routing Form, as described in Chapter 4.*

The following is a sample greeting message:

```
An electronic list account has just been opened for you, and has
been configured as indicated below.  See the instructions below the
account summary for information on how to make changes to your mail
account as well as for explanations about each of the fields.

    List-Addresses:        surfing@software.com
                           surfing@sparky.software.com

    List-Name:                          surfing
    Remote-Subscriber-Policy:           open
    Verify-Subscriptions:               yes
    Verify-Unsubscriptions:             no
    Moderate-Unsubscriptions:           no
    Subscriber-Posting-Policy:          open
    Nonsubscriber-Posting-Policy:       closed
    Digest-Schedule:                    daily

============================================================
Outsiders can subscribe by connecting to:

http://sparky.software.com/guest/RemoteListSummary/surfing

You can maintain the list via:

http://sparky.software.com
```

Figure 7-12: List owner greeting message

Included in this message are the values you set for such list parameters as the List Name and policies for subscription, posting, and unsubscription. The owner greeting message also includes two URLs: one is the regular web address for logging into Post.Office, while the other is the address to a Mailing List Summary Form for public subscription to the mailing list. List owners can distribute this URL to potential subscribers to allow them to subscribe via the remote user web interface, as described in Section 7.10.3.

7.5 Modifying a Mailing List

Although routine changes to the mailing list – such as adding and removing subscribers, changing a policy, and setting a new welcome message – can (and should) be done by the list owner, the Postmaster's supreme authority over the system includes the ability to modify all existing mailing lists, regardless of who owns them. This section describes the portions of the Postmaster's interface that you can use to carry out such modifications.

Note: *This section describes the modification of mailing lists from the Postmaster's perspective; that is, while a user is logged into Post.Office as the Postmaster. If you are the Postmaster, but are also a list owner, you will find it simpler to manage your owned mailing lists by logging in to Post.Office as your local user account and using the end user interface. This interface is described in the Post.Office List Owner's Guide.*

7.5.1 Changing the List Settings

Mailing list attributes are set in the Mailing List Data Form, just like the one you used to set mailing list defaults and create new mailing lists (the long way, that is). This form should be familiar to you by now; if it isn't, look over Section 7.3 for a refresher.

You can access the Mailing List Data Form for a mailing list in two ways: the first is via the shortcut field at the bottom of the Mailing List Administration menu; the second is from the List of Mailing Lists menu, which displays all of the mailing lists in Post.Office. We'll start with the Mailing List Administration menu, which looks like the following:

Figure 7-13: Mailing List Administration menu

Notice the drop-down menu and text field at the bottom of the browser window. This field is similar to the Account Management menu shortcut field described in Chapter 5, and allows you to directly access mailing list-related forms without generating the entire list of mailing lists. To display the Mailing List Data Form for a specific mailing list, enter its List Name or one of its addresses in the text field, select **Mailing List Data Form** from the drop-down menu, and click **Get**.

Note: *You can use partial strings with a wildcard (*) in the shortcut field to get a list of all mailing lists that match a particular name or address pattern.*

The more general method of accessing mailing list data is to display the List of Mailing Lists menu, and then select a particular mailing list. This menu is displayed when you click the **List of Mailing Lists** link on the Mailing List Administration menu.

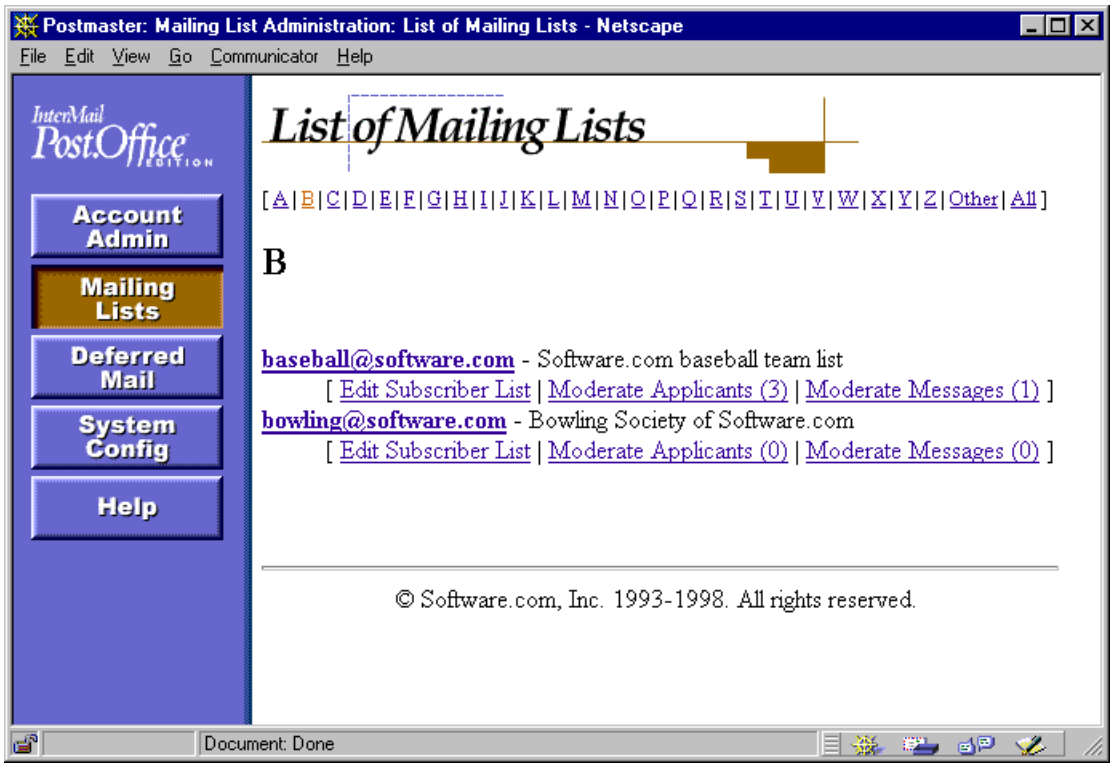

Figure 7-14: List of Mailing Lists menu

This menu includes the primary address and short description of each displayed mailing list. Each mailing list address is the link to a Mailing List Data Form, so click on the appropriate address to view and/or modify the list's attributes.

We won't subject you to viewing this entire form again, so refer back to Figure 7-3 through Figure 7-9 if you want another look at the Mailing List Data Form. As with all other forms, you can make changes by modifying the contents of one or more form fields and clicking **Submit**. To cancel your changes, click **Reset** or BACK.

Note that the List of Mailing Lists menu includes three additional links for each mailing list. These links allow you to access forms for performing other mailing list-related operations: editing the subscriber list, moderating subscription and unsubscription requests, and moderating messages (the numbers next to the moderation links indicate the number of applicants or messages that are currently waiting for moderation). Subscriber list operations are discussed in the following section, while moderation issues are covered in Section 7.6.

7.5.2 Adding and Removing Subscribers

One of the most important attributes of a mailing list is its list of subscribers. The subscriber list can be modified in the List of Subscribers Form, which is displayed by clicking the **Edit Subscriber List** link for a mailing list in the List of Mailing Lists menu (Figure 7-14), and which looks like the following:

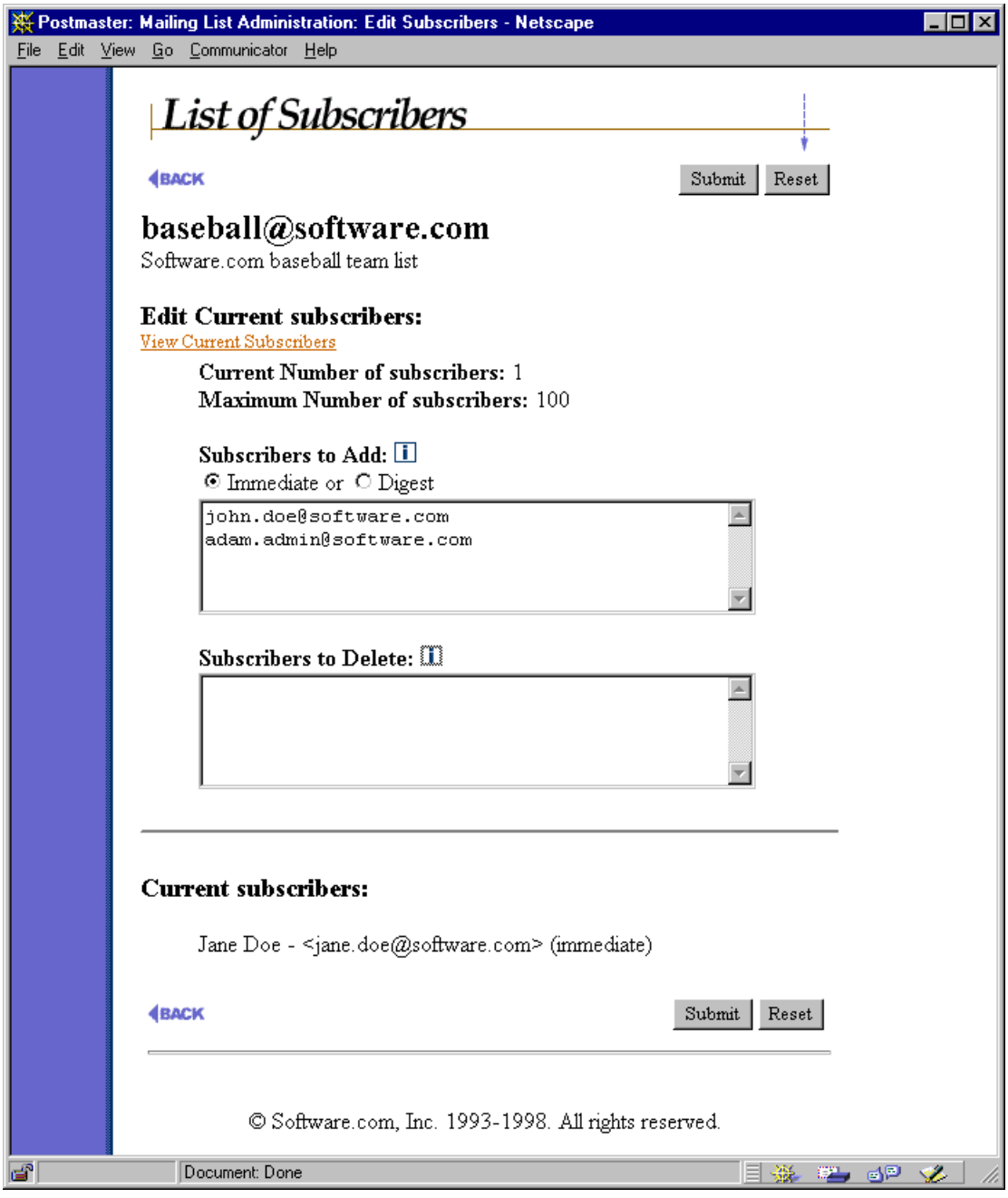

Figure 7-15: List of Subscribers Form

To add subscribers, enter an e-mail address for each new subscriber in the **Subscribers to Add** field, and select a delivery method for these subscribers.[43] To remove subscribers, enter the address of each soon-to-be non-subscriber in the **Subscribers to Delete** field. To commit the changes to your subscriber list, submit the form.

Subscribers added or removed in this manner are exempt from the intermediate steps endured by users who attempt to subscribe or unsubscribe themselves. This means that no verification or moderation will take place for these users, even if the policies for the list include those options. If a list has closed subscription policies, users can still be added to the subscriber list in this form.

Automatic Unsubscription

It's possible for a subscriber to be automatically removed from a mailing list. This automatic unsubscription occurs when the mailing list receives bounced (returned) messages sent to a particular subscriber. Instead of eternally sending messages to accounts that may no longer exist, the mailing list keeps its subscriber list current by removing any account that generates too many return bounces (as defined in the Error Response Parameters Form, discussed in Chapter 4).

Automatic unsubscription affects only remote users, so the users whose accounts are stored in your installation of Post.Office needn't worry about it. However, if you store e-mail accounts and mailing lists on separate mail servers, problems with your site's accounts may cause return bounces.

43 You can only select one delivery mode for all subscribers that you add in one operation. If for some reason you want to add some users with the digest mode, and other users with the immediate mode, you must add one group at a time.

7.5.3 Viewing Current Subscribers

The list of current subscribers can be viewed from the List of Subscribers Form described above by clicking on the **View Current Subscribers** link. You can also go directly to the list of current subscribers from the shortcut field on the Mailing List Administration menu (Figure 7-13 above), which includes the menu entry **Subscribers.** Both of these methods display a View List Subscribers Form:

Figure 7-16: View List Subscribers Form

The e-mail address and delivery mode of each subscriber are given here. Use the **A-Z** links to search through subsets of subscribers, or click on the **All** link to view a list of all list subscribers.

To go to the List of Subscribers Form, click on the **Edit Current Subscribers** link. If you want to go all the way back to the List of Mailing Lists menu, click the BACK link.

7.6 Moderating a Mailing List

If a mailing list has policies to moderate subscription requests, messages, or unsubscription requests, the list owner will be required to periodically approve or reject new requests or messages. This moderation can be done by the list owner through the web interface and/or e-mail interface, depending on how the moderation policy of the mailing list has been set.

Although moderation is generally the jurisdiction of the list owner, the Postmaster also has the power to moderate messages and subscription requests for any mailing list. However, this Postmaster moderation power applies only to mailing lists that have a moderation policy that includes the web interface (i.e., the moderation policy is **Web only** or **Web and e-mail**). When one of these moderation modes is selected, the Postmaster can use the same moderation web forms as the list owner.

Note: *When the **E-mail only** mode of moderation is selected, no subscription requests or messages are held by Post.Office while awaiting moderation; instead, they are simply forwarded to the list owner, who must manually submit them him/herself via e-mail to approve them. Since there is technically nothing in the mail server to moderate in this case, the Postmaster can't moderate any mailing list that uses **E-mail only** moderation.*

7.6.1 Applicants

Both subscription and unsubscription requests that are held for moderation can be resolved with the Applicants to Moderate Form. As with other mailing list-related forms, you can get to the Applicants to Moderate Form from two locations. The first is from the shortcut field on the Mailing List Administration menu (Figure 7-13), which includes the menu entry **Applicants to Moderate**. The second is from the List of Mailing Lists menu (Figure 7-14), which includes a **Moderate Applicants** link for each mailing list. Both methods display an identical Applicants to Moderate Form, which looks like the following:

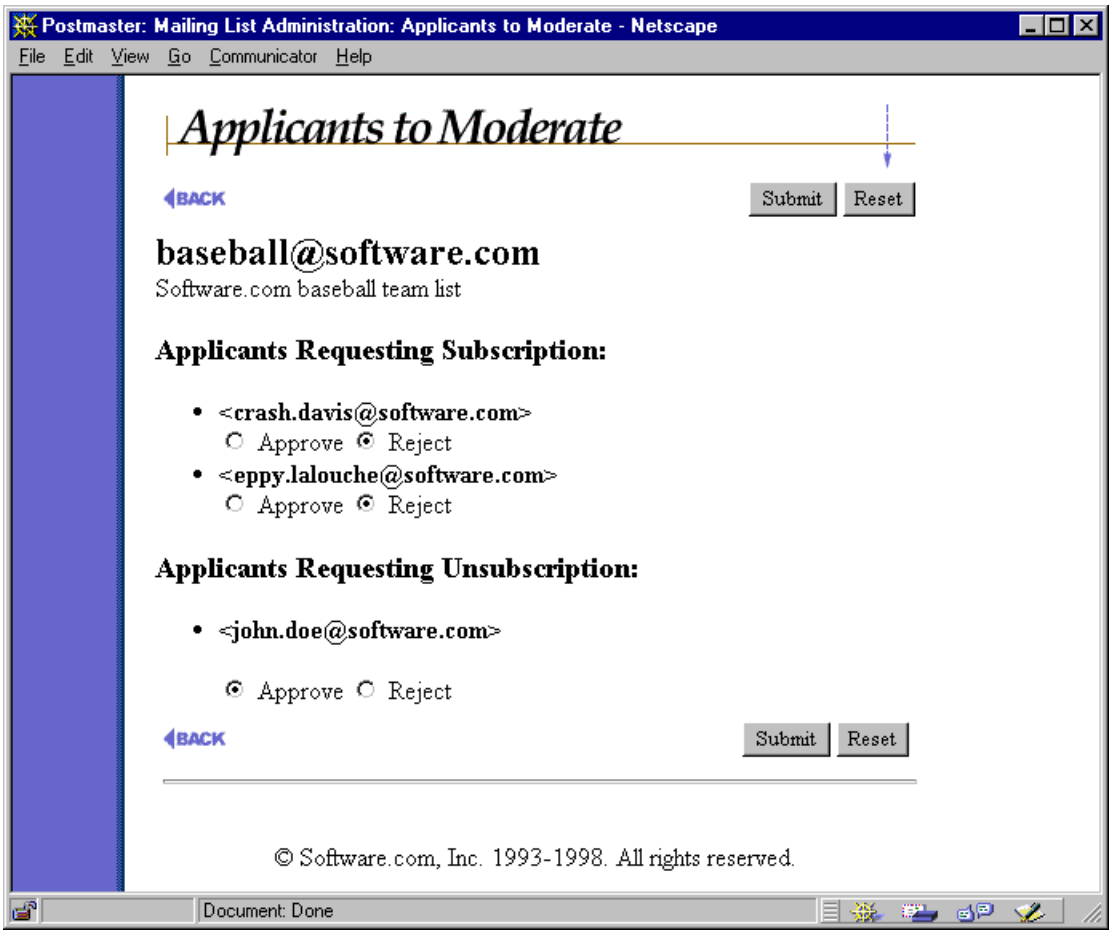

Figure 7-17: Applicants to Moderate Form

This form includes the radio buttons **Approve** and **Reject** beneath each would-be subscriber or unsubscriber. Select the appropriate radio button for each user and submit the form to carry out your judgment. Approved users are immediately added to the subscriber list and will receive the list's welcome message (if one exists), while rejected users are notified (politely) that their request was denied.

If you approve or reject only some of the applicants, the unmoderated leftovers will continue to be held for future judgment.

7.6.2 Messages

Messages that are held for moderation can be resolved with the Messages to Moderate Form. As with other mailing list-related forms, you can get to the Messages to Moderate Form from two locations. The first is from the shortcut field on the Mailing List Administration menu (Figure 7-10), which includes the menu entry **Messages to Moderate**. The second is from the List of Mailing Lists menu (Figure 7-14), which includes a **Moderate Messages** link for each mailing list. Both methods display an identical Messages to Moderate Form, which looks like the following:

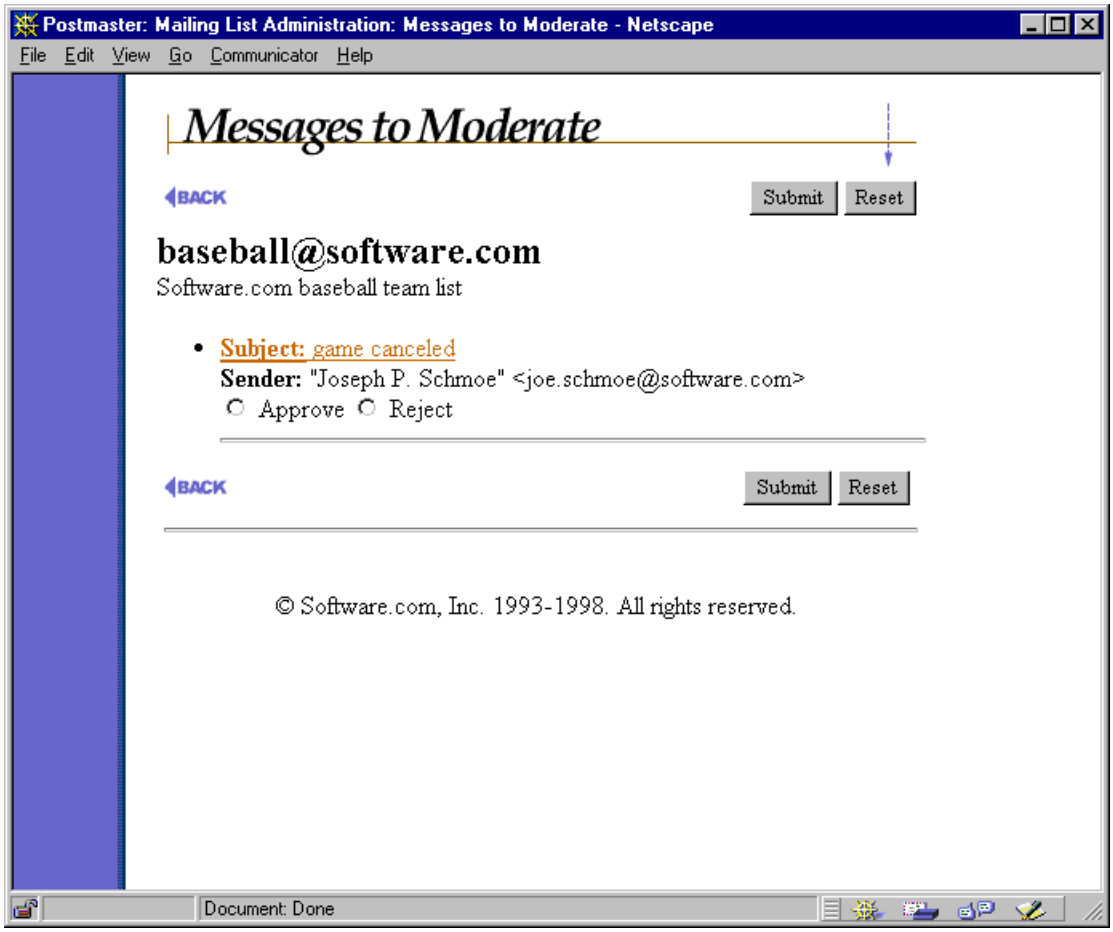

Figure 7-18: Messages to Moderate Form

For each waiting message, the form displays the subject of the message, the sender, and the radio buttons **Approve** and **Reject**. You can give each message your blessing or denial by selecting the appropriate radio button and then submitting the form. As with subscription requests, approving/rejecting only some of the messages causes the remaining submissions to be held for subsequent moderation.

If you want to view the actual contents of a message before deciding on its fate, you can do so by clicking on the subject line of the message. This displays the Moderated Message Form, which allows you to read the message and approve or reject it.

Figure 7-19: Moderated Message Form

As with the Messages to Moderate form, you can approve or reject the message by selecting the appropriate radio button (**Approve** or **Reject**) and submitting the form. You can also click on the **Edit Message Text** link, which invokes yet another form, the Message Text Form. This final message form is used to modify the text of a message before it is posted to the mailing list.

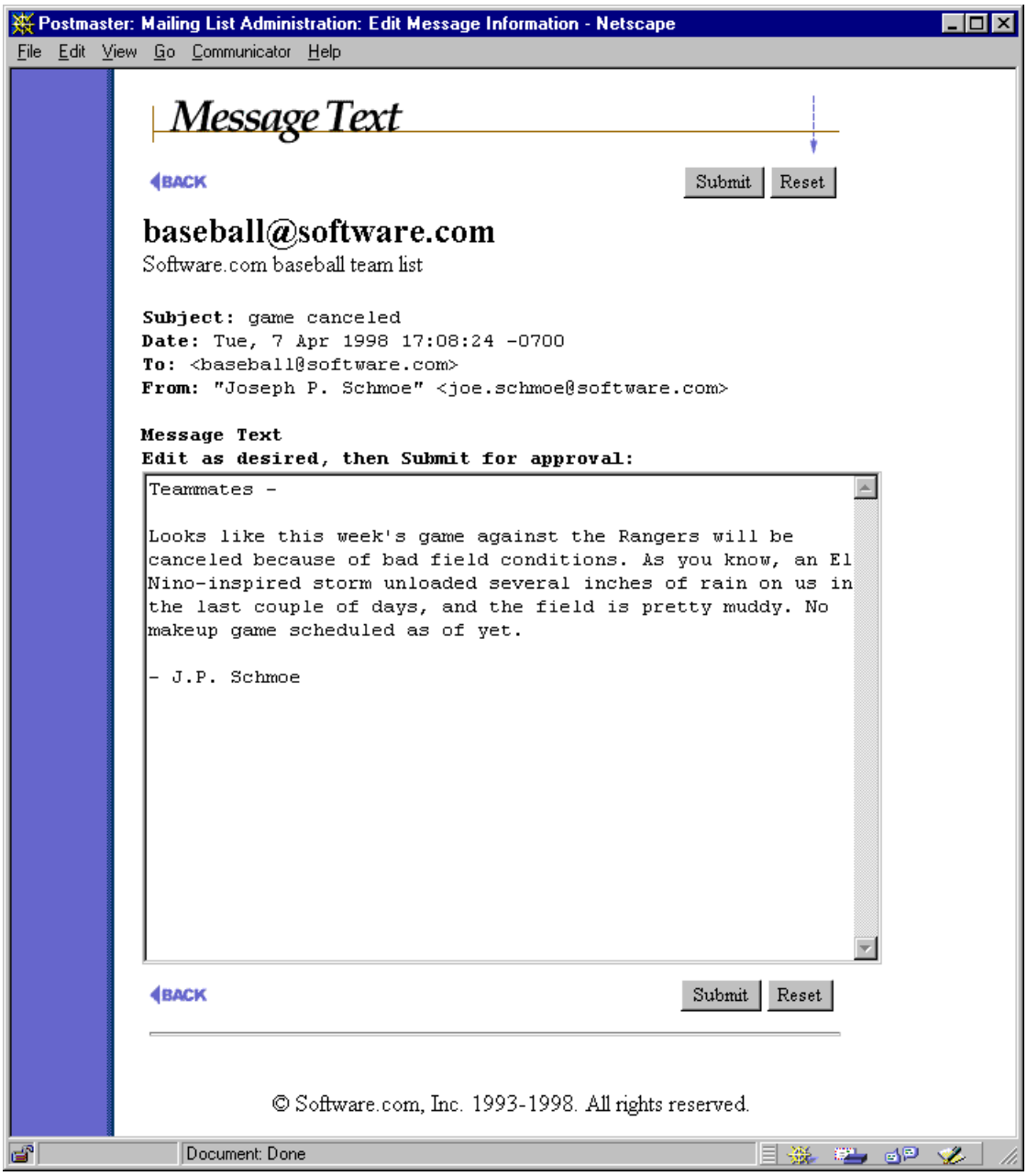

Figure 7-20: Message Text Form

To modify the message, simply edit the contents of the **Message Text** field and submit the form. Submitting changes to the message takes you back to the Moderated Message Form, which you can use to then approve or reject the modified message.

7.7 Locking a Mailing List

Like accounts, mailing lists can be locked to suspend activity. Locking a mailing list allows you to shut it down without permanently deleting it from your system. When a mailing list is locked, no postings will be accepted by the list, and the list owner will be prevented from modifying or accessing the list in any way.

To lock a mailing list, set the **Lock mailing list** option to **Yes** on the Mailing List Data Form (refer back to Figure 7-6) and submit the form. You can restore the list to normal operation by resetting this option to **No**.

7.8 Deleting a Mailing List

Deleting a mailing list is very similar to deleting an account, as described in Chapter 5. Mailing lists can be deleted by clicking on the **Delete List** button shown at the top of the Mailing List Data Form.

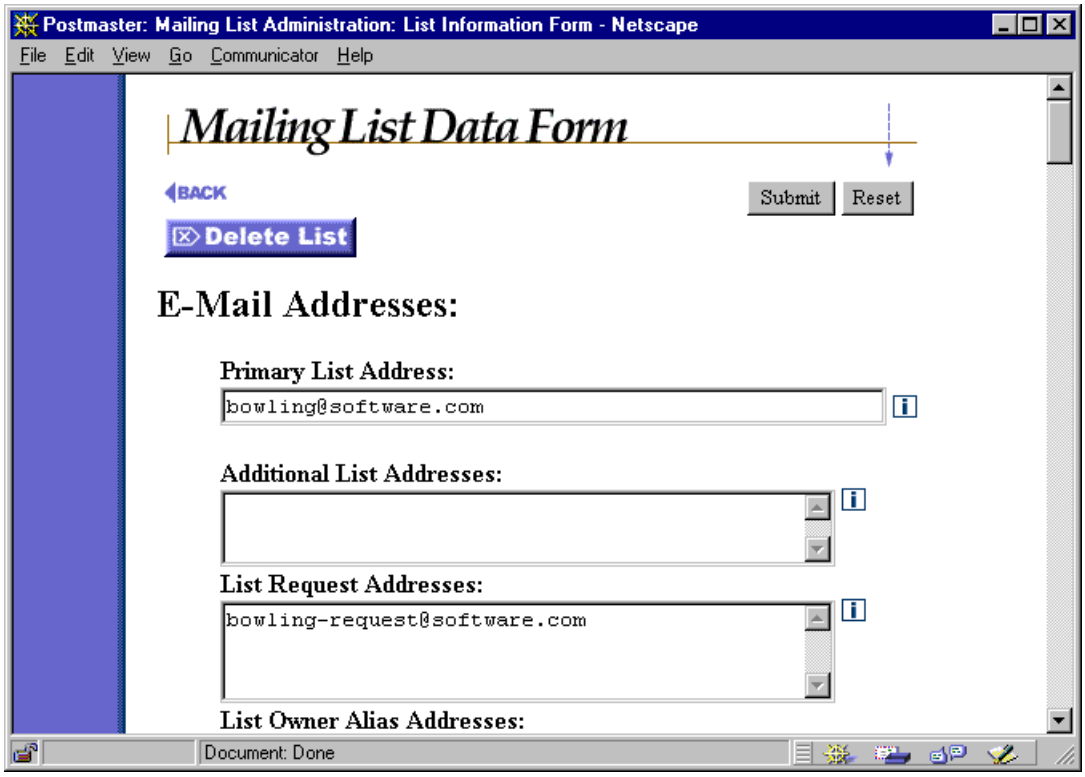

Figure 7-21: Mailing List Data Form (the top of it, at least)

After clicking this button, you will be asked to confirm the deletion before the list is permanently removed from the system.

Remember that subscribers will not receive the farewell message when the mailing list is deleted. This allows you to remove resource-draining mailing lists without generating a

stack of new messages to distribute. If you want to notify all of the list's subscribers that the mailing list will be deleted, you can first manually unsubscribe all users from the List of Subscribers Form (which causes each subscriber to receive the farewell message) and then delete the list.

7.9 The All-Mailboxes List

As discussed in Chapter 5, Post.Office includes a special mailing list that distributes postings to all accounts on your system that use the POP3 delivery method. The postmaster account is the owner of this mailing list, which has the posting address `All-Mailboxes@host.domain`.

Note again that this list distributes messages only to accounts that use the POP3 method of delivery. By default, your accounts that use only forwarding, Unix delivery, or program delivery will not receive messages sent to this list. To include these accounts in the All-Mailboxes list, manually add these accounts to its subscription list in the Subscribers Form.

Because of the security issues involved, this mailing list is locked upon installation, which prevents use of the All-Mailboxes list. In order to enable this feature, you must manually unlock this list (as described in Section 7.7). When the All-Mailboxes list is used by a site, it typically has very restrictive policies that allow only the Postmaster to post messages here. After all, you probably don't want just anybody to have access to this direct line to all of your users. However, you can configure this mailing list just as you do any other mailing list, so you can set whatever policies you deem appropriate.

7.10 What Your Users See

As described at the beginning of this chapter, the mailing list manager involves four different classes of Post.Office users, all of whom have different levels of access to different parts of the program. The mailing list manager interface for the Postmaster was described in the previous sections, but different interfaces exist for the three other user types. As the person responsible for running the mail system, you're probably curious about just what these interfaces look like.

This section takes a brief tour of the mailing list manager interfaces to local users, list owners, and remote users. For more information on these operations, refer to the *Post.Office User's Guide* and the *Post.Office List Owner's Guide*.

7.10.1 Local Users

The mailing list manager web interface available to local users is very similar to the end user account management interface. After logging in to Post.Office from the Authentication Information Form, users can click on the **Mailing Lists** menu button at the left to display the Mailing List Management menu.

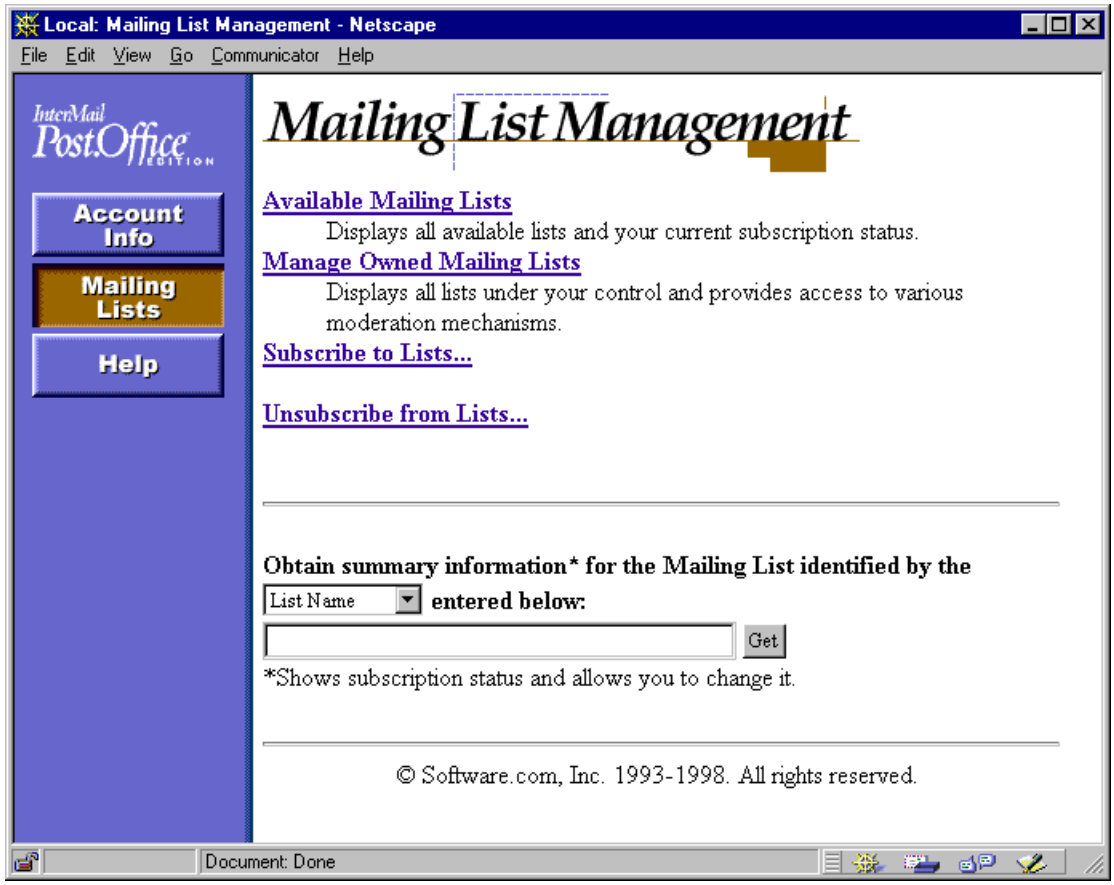

Figure 7-22: Mailing List Management menu

The menu contains four links – **Available Mailing Lists**, **Manage Owned Mailing Lists**, **Subscribe to Lists…**, and **Unsubscribe from Lists…** – as well as a text field and execution button. The text field allows users to get information about a specific mailing list without generating a list of every mailing list in the system.

To see which mailing lists are available to them, users click on the **Available Mailing Lists** link. This displays an alphabetical list of the mailing lists that have an open or moderated subscription policy for local users, along with a flag that marks the ones that they are already subscribed to.

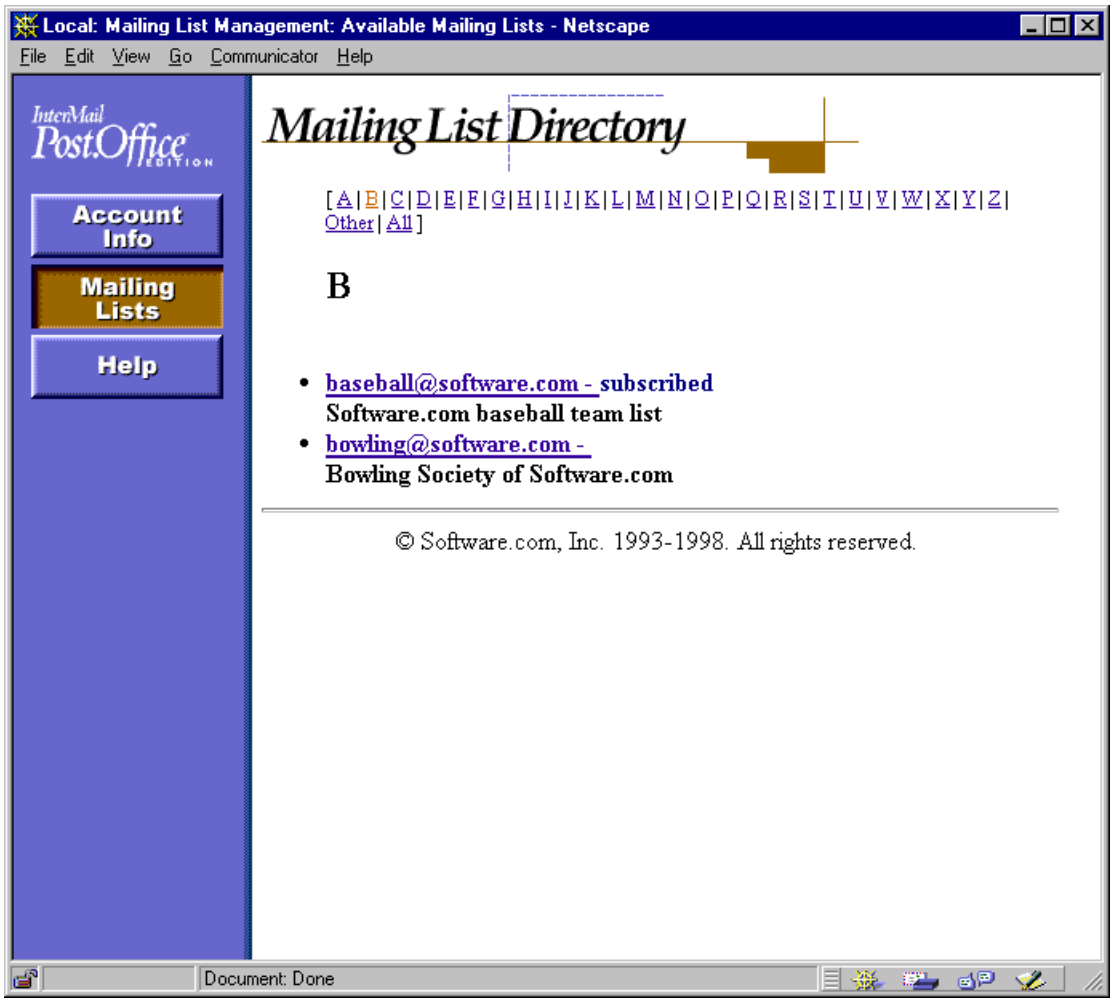

Figure 7-23: Mailing List Directory menu

For each mailing list, an address and a short description are displayed in this menu. If the user is currently subscribed to any of these mailing lists, they will be marked by the word **subscribed** to the right of the list address. Each list address is a link to a Mailing List Summary Form, which provides detailed information for the list and also lets users subscribe to (or unsubscribe from) the mailing list. This is the same form that is invoked by using the shortcut field on the local user's Mailing List Management menu (Figure 7-22).

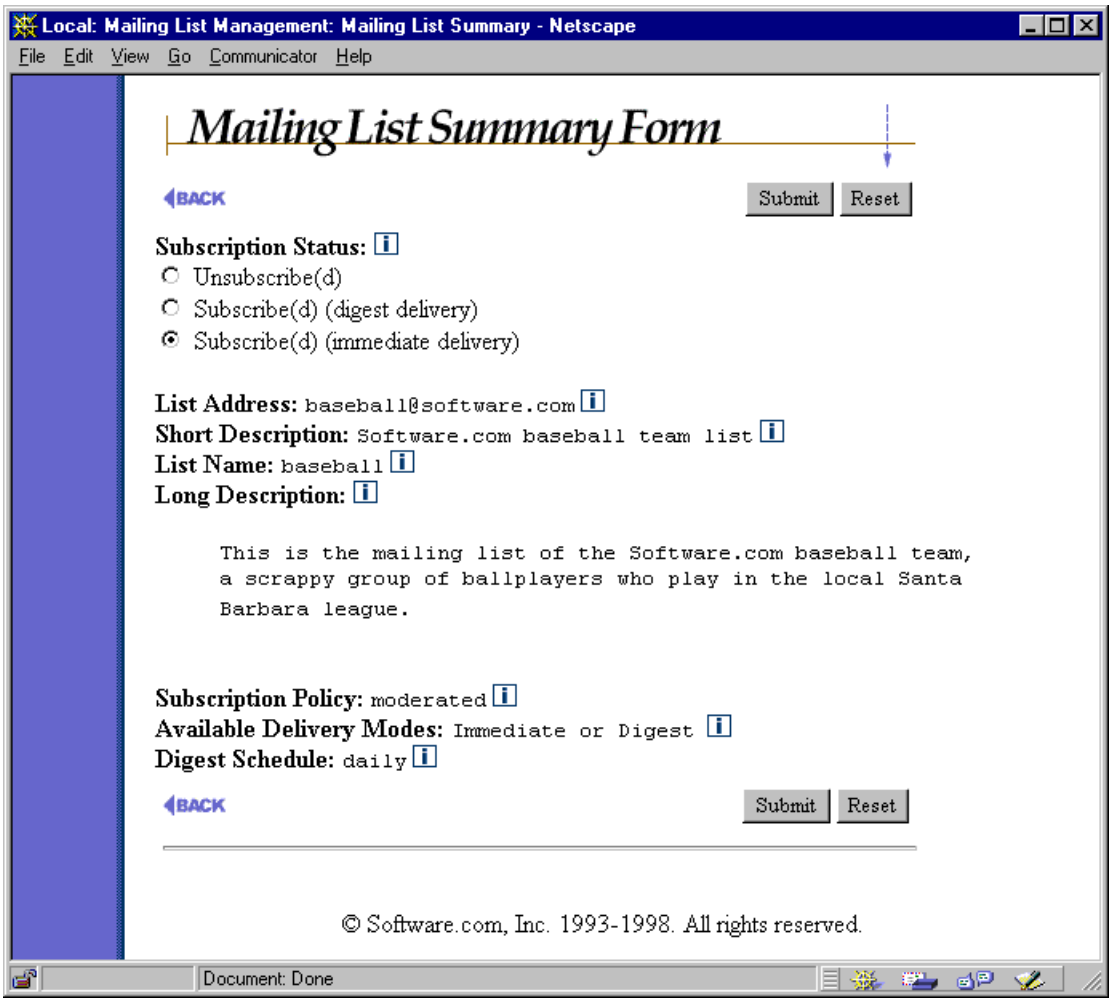

Figure 7-24: Mailing List Summary Form

In addition to the Mailing List Summary Form, local users have two other forms for requesting subscription or unsubscription for mailing lists: the Subscription Form, and the Unsubscription Form. These are the forms invoked from the **Subscribe to Lists…** and **Unsubscribe from Lists…** links on the Mailing List Management menu, and they allow users to submit requests for multiple mailing lists. These forms display only the mailing lists that apply to the current operation, so only mailing lists to which the user is already subscribed show up in the Unsubscription Form (and vice versa for the Subscription Form).

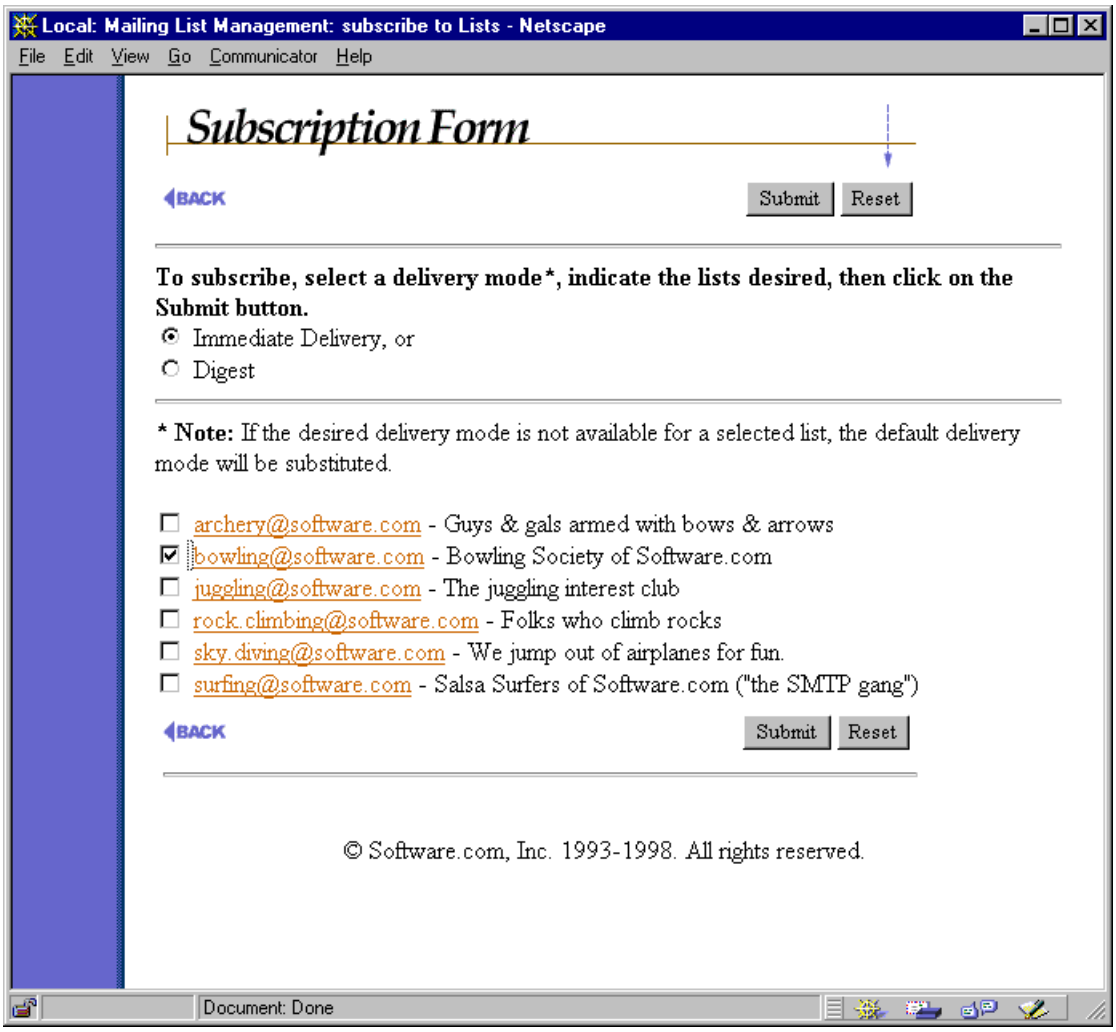

Figure 7-25: Subscription Form

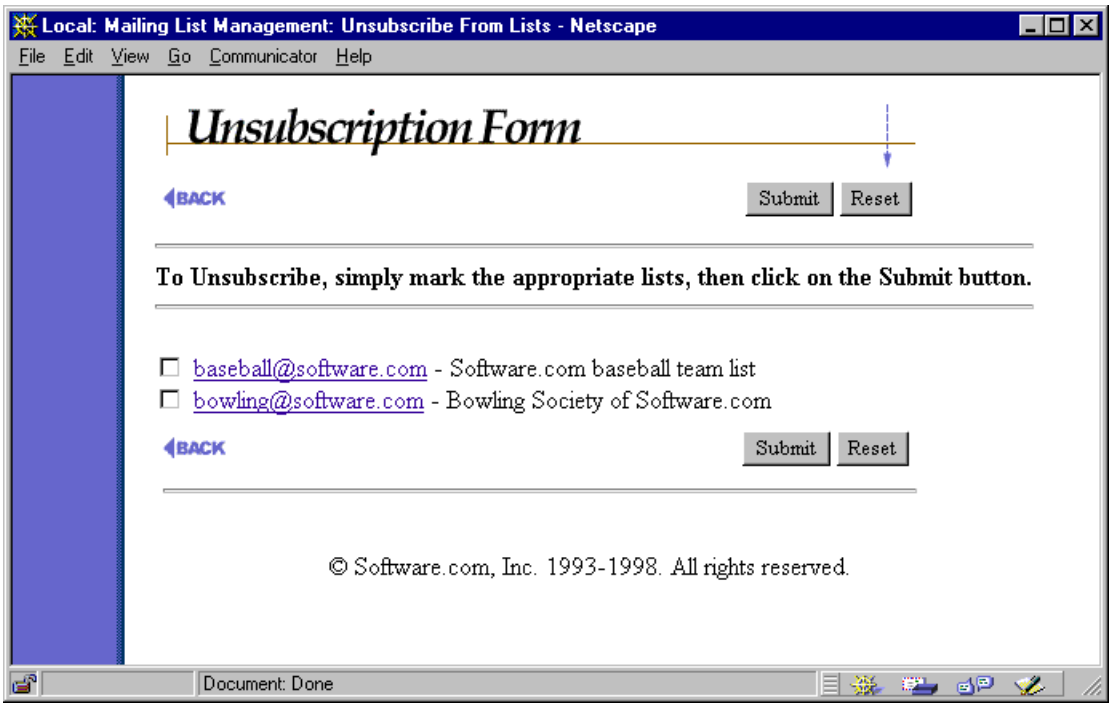

Figure 7-26: Unsubscription Form

Like on the Mailing List Directory menu, the address of each mailing lists shown on the Subscription Form and Unsubscription Form is a link to a Mailing List Summary Form, which the user can view to get more information on a particular mailing list.

7.10.2 List Owners

The web interface available to list owners is a combination of the local user and Postmaster list manager interfaces. As local users, they have access to the same forms for subscribing and unsubscribing from mailing lists that are available to all local users. However, list owners can also access forms for carrying out administrative tasks by clicking on the **Manage Owned Mailing Lists** link on the Mailing List Management menu. This displays the Owned Mailing Lists menu.

Figure 7-27: Owned Mailing Lists menu

As you can see, this menu is very similar to the Postmaster's List of Mailing Lists menu, but contains only the mailing lists owned by the user. Each list address is a link to a Mailing List Information Form, which is similar to the Postmaster's Mailing List Data Form and allows the list owner to modify most attributes of the list.[44] Three additional links for each mailing list allow list owners to access the same Subscriber List Form (Figure 7-15), Applicants to Moderate Form (Figure 7-17), and Messages to Moderate Form (Figure 7-18) available to the Postmaster. The tasks performed by the list owner are identical to the techniques described in Sections 7.5 and 7.6.

44 This form is nearly as long as the Postmaster's Mailing List Data Form, so we won't show it to you here. Refer to the *Post.Office List Owner's Guide* if you really want to see it.

7.10.3 Remote Users

A mailing list is referred to as “public” if its subscription policies allow users without e-mail accounts on your Post.Office server to subscribe to the mailing list (recall from Section 7.3.3 that there are separate subscription policies for users with and without e-mail accounts in Post.Office). These users – known as *remote users* – obviously cannot use the same interface to subscribe to mailing lists as local users, since they cannot log in to the system from the Authentication Information Form.

Nevertheless, remote users are still offered a web-based interface for subscribing to mailing lists hosted by the Post.Office. This interface can be accessed from the Authentication Information Form by clicking the **Mailing List Directory** button, which displays the Mailing List Directory menu for remote users.

Figure 7-28: Mailing List Directory menu (remote user version)

This form is very similar to the local user’s Mailing List Directory, and allows remote users to view the mailing lists in Post.Office.

Although users can get to this form through the Authentication Information Form, they can also simply go to it directly, since no login or authentication information is required (remember, this is the “public” area). This means that you can provide remote users with a URL that takes them specifically to a form for subscribing to your public mailing lists.

For all installations of Post.Office, this public mailing lists URL is at:[45]

`http://host.domain/guest/RemoteAvailableLists`

Users can click on an individual list address in the Mailing List Directory to access a Mailing List Summary Form for that list.

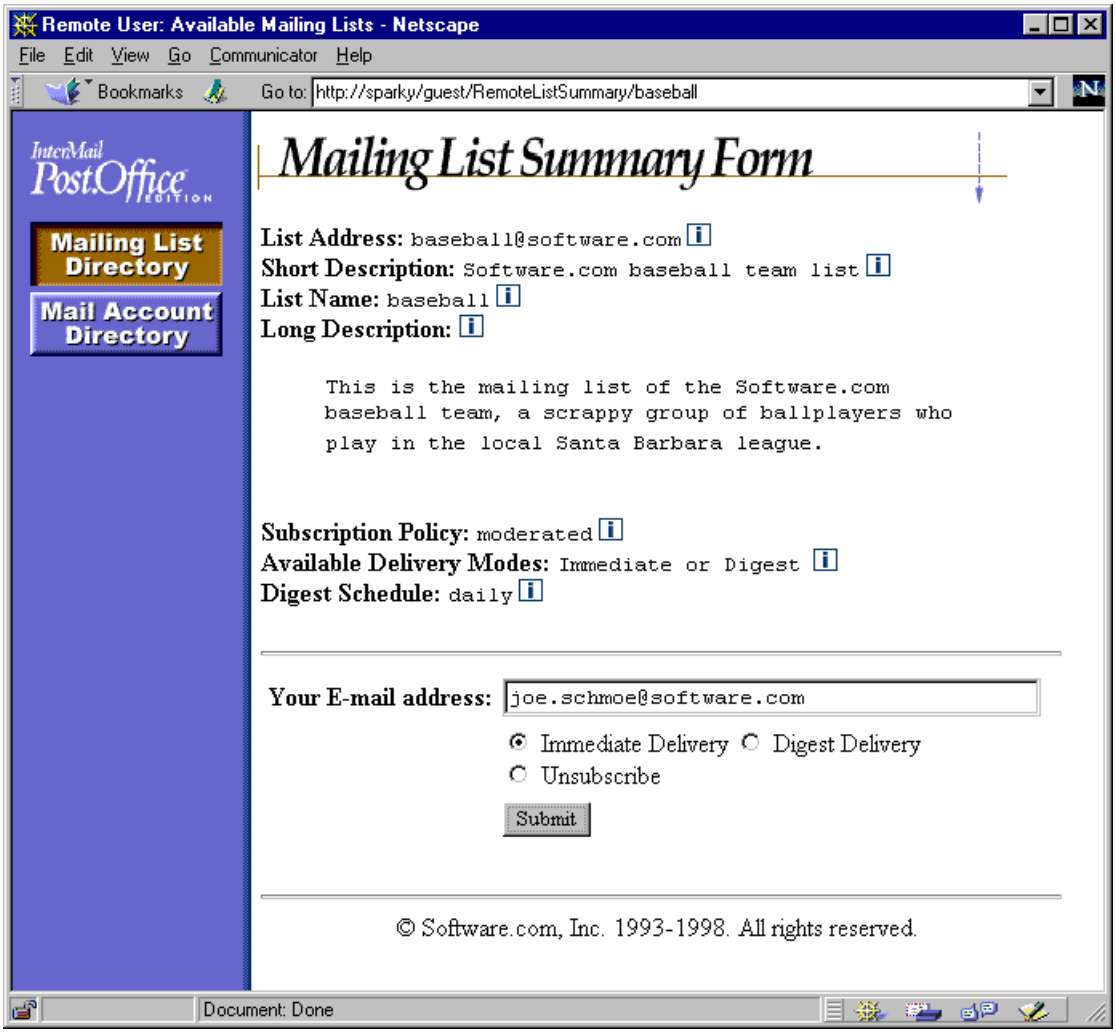

Figure 7-29: Mailing List Summary Form (remote user version)

Notice that this version of the List Summary Form includes a text field to allow the user to enter his or her e-mail address when requesting subscription. This information is required to process subscription requests by remote users because they did not provide an address and password in the Authentication Information Form.

45 This assumes that your Post.Office web server is running on port 80, the default. However, if your Post.Office server runs on a port other than port 80, you should include the server port number in the URL, as described in Chapter 3.

Like the public Mailing List Directory menu, users can directly access these public Mailing List Summary Forms instead of going through the menu. This form can be found for each public mailing list at:

`http://`*`host`*`.`*`domain`*`/guest/RemoteAvailableLists/`*`listname`*

Remember from the above section on local users that when displaying lists of mailing lists, Post.Office filters out the mailing lists that have a closed subscription policy. This rule also applies in the case of remote users: if a mailing list's remote user subscription policy is closed, this mailing list is hidden from all users who do not have Post.Office accounts on your system. So if you have a sensitive mailing list – like an internal employee list – you can ensure that it is safely hidden from the teeming masses by closing its subscription policy to remote users.

7.11 List Manager E-mail Interface

In addition to the web interface, Post.Office provides an e-mail interface for interacting with the mailing list manager. However, the mailing list manager e-mail interface supports only end user and minor list owner operations; the Postmaster's administrative mailing list manager tasks can be carried out *only* from the web interface.[46] So while this e-mail interface allows end users to request subscription to (and unsubscription from) mailing lists, and allows list owners to moderate subscription requests and messages, it does not allow the Postmaster to create, modify, or delete mailing lists.

Because you can't actually do any Postmaster-specific operations via the list manager's e-mail interface, it doesn't really apply to you. However, the rest of this section offers you an overview of using the e-mail interface to give you a better idea of what we're talking about here. For more information on executing specific end user or list owner operations via the e-mail interface, refer to the *Post.Office User's Guide* and *Post.Office List Owner's Guide,* respectively.

7.11.1 Submitting List Manager Requests

Like the popular Majordomo list manager program, the Post.Office e-mail interface accepts *requests* submitted to special e-mail accounts, and processes them accordingly. A request is simply an e-mail message that contains one of several keywords for performing certain mailing list operations, such as subscribing to a mailing list, getting a list of your current subscriptions, or getting descriptive information for a mailing list. For example, to request subscription to the list, a user can send an e-mail message that contains the word `subscribe`.

Because of the simplicity of this interface, it is sometimes easier to accomplish things this way than in the web interface.

46 This isn't entirely true, since command-line utilities are provided for executing many mailing list-related operations. Refer to Chapter 11 for more information.

Two Destinations for Request Messages

List manager requests messages can be addressed to one of two places. The first is the system's List Manager account, which is used when submitting commands for multiple mailing lists. The address of this account is

`list.manager@`*`host.domain`*[47]

The second request message destination is the request account of a specific mailing list, which is used for various support functions for the mailing list, such as sending welcome and farewell messages and processing e-mail commands. The address of the request account varies from list to list. The convention used by Post.Office is to append "-request" to the local portion of the primary list address to get the address of the request account. For example, for a mailing list with the primary address

```
cycling@software.com
```

the convention for the request account address would be

```
cycling-request@software.com
```

The only difference between sending a request to the List Manager account and sending the same request to the request account for a specific list is that you do not then need to include the name of the list for which you are submitting the request. The mailing list in question is assumed to be the one associated with the request account, which saves you the trouble of including an extra parameter with your request to identify the name of the mailing list.

Note: *This doesn't mean that the request account for a mailing list is limited to accepting commands only for its associated list. It isn't. As with the system-level List Manager account, you can send commands for any mailing list to any request account. The individual request accounts just simplify operations by assuming a default mailing list; you can always override the default by specifying a List Name with the command.*

Request Syntax

The syntax for request messages is fairly flexible. The subject line is always ignored, and may be left blank. Commands are entered one per line in the message body, with leading whitespace (spaces or tabs) ignored. Command keywords and any arguments must be separated by at least one space or tab.

47 With your server's hostname and domain, of course.

For example, to request mailing list information for the mailing list that has the List Name `surfing`, you would send a message like the following:

```
To:       list.manager@software.com
From:     john.doe@software.com
Subject:
-------------------------------------------------------
info surfing
```

Figure 7-30: Request for list information (sent to system account)

Assuming that you followed the convention for the request account address when creating the mailing list, the above request could have also been addressed as follows:

```
To:       surfing-request@software.com
From:     john.doe@software.com
Subject:
-------------------------------------------------------
info
```

Figure 7-31: Request for list information (sent to mailing list request account).

Notice that we no longer have to tell the system which list we're trying to get information on. Because this message is sent to a request account for a specific mailing list, the system assumes that this is the mailing list for which we're submitting the request.

Responses to Requests

Most e-mail commands receive a reply message from the system that returns the requested information or tells you the result of your request. For example, after the message shown in Figure 7-30 is submitted, the response similar to the following would be received:

```
To:       john.doe@software.com
From:     list.manager@software.com
Subject:  List Manager response
-------------------------------------------------------
>>>>info surfing
This is a mailing list for all the gnarley dudes and dudettes at
Software.com who enjoy riding good waves on the beaches of Santa
Barbara. We believe strongly in longboards, margaritas, and e-mail.
Surf's up!
```

Figure 7-32: Sample response to the "info" command

For subscription and unsubscription requests, the response message may also include information regarding verification or moderation. The response message may also indicate that you do not have the appropriate access to execute the command (for example, if you attempt to subscribe to a mailing list with a closed subscription policy).

Submitting Multiple Commands

More than one command can be submitted in the same message, with each command on its own line. In the following example, the sender is requesting subscription to two mailing lists, and requesting unsubscription from a third:

```
To:       list.manager@software.com
From:     john.doe@software.com
Subject:
-------------------------------------------------------
subscribe surfing
subscribe mountain_biking
unsubscribe yachting
```

Figure 7-33: Sample for multiple requests

There is no limit to the number of commands that can included in a single request message. If one of the commands in a request message fails because of a syntax error or access restrictions, the remaining commands are still processed.

The Trouble With Signatures

Many users have signatures automatically appended to all of their e-mail as it is sent from their mail client. Unfortunately, these signatures can wreak havoc on your ability to submit commands via the list manager e-mail interface, which will attempt to process all of the commands – that is, all of the text – in your request message.

Post.Office will screen out many types of signatures when processing e-mail commands, but you may still experience problems when submitting requests because of your signature. To prevent this, you should use the end command at the conclusion of the your other e-mail commands. This command instructs Post.Office that there are no more commands in this message, so it will not attempt to process any of the remaining text of your message, including your signature.

The following example shows the same requests as Figure 7-33, but with the end command to stop Post.Office from processing the signature.

```
To:       list.manager@software.com
From:     john.doe@software.com
Subject:
------------------------------------------------------------
subscribe surfing
subscribe mountain_biking
unsubscribe yachting
end

                        %%%%%%%
                       %% ~ ~ %%
                       (  @ @  )
********************oOOo*(_)*oOOo**********************
John Doe Jr., Ph.D.              (805)882-2470  x000
Software.com                     (805)882-2473  FAX
525 Anacapa St.                  (805)555-1076  Page
Santa Barbara, CA 93101          (805)555-1176  Cell
```

Figure 7-34: Using end to prevent signature errors

Because the `end` command signals the conclusion of the valid commands, Post.Office will not attempt to execute mailing list-related operations based on the contents of this signature. If the `end` command had not been included an error would have resulted, since

```
                        %%%%%%%
                       %% ~ ~ %%
                       (  @ @  )
********************oOOo*(_)*oOOo**********************
```

is not currently a supported e-mail command.

7.11.2 Available End User Commands

The following table lists the commands available to all users – both local and remote – from the mailing list manager e-mail interface. Parameters shown between [square brackets] are optional, while parameters shown in *italics* must be replaced by an appropriate value.

If the request message is sent to the Request Address for a specific mailing list, you do not have to specify the List Name as a command parameter. However, if the request message is sent to the system's general list management account (`list.manager@`*`host.domain`*), the *`listname`* parameter shown in the table below becomes a required parameter.[48]

Command	Additional Parameters	Description
`subscribe`	*`listname`* [*`address`*] [digest]	Requests subscription for the sender (or the specified address).
`unsubscribe`	*`listname`* [*`address`*]	Requests unsubscription for the sender (or the specified address).
`which`		Returns a list of the sender's current list subscriptions.
`who`	*`listname`*	Returns the subscriber list if sender has appropriate access.
`info`	*`listname`*	Returns the mailing list's long description.
`lists`		Requests a list of the available mailing lists.
`help`		Requests a list of the e-mail commands available to all users.
`end`		Marks the end of commands included in the request message, preventing the system from attempting to process text in the signature.

48 You must also include the List Name as a parameter if you are sending commands to a request account for a mailing list other than the mailing list for which you're submitting commands. Remember, if you send your commands to a mailing list's Request Address, the system will assume that this is the list you are trying to submit commands for (unless you tell it otherwise by specifying a List Name).

7.11.3 Available List Owner Commands

The following table lists the commands available for performing list owner activities via the e-mail interface. Parameters shown between [square brackets] are optional, while parameters shown in *italics* must be replaced by an appropriate value.

As with end user requests, if the request message is sent to the Request Address for a specific mailing list, you do not have to specify the List Name as a command parameter. However, if the request message is sent to the system's general list management account (list.manager@*host.domain*), the *listname* parameter shown in the table below becomes a required parameter.

Command	Additional Parameters	Description
`subscribe`	*`listname address`* `[digest]`	Requests subscription for the specified address; also used to approve moderated subscription requests.
`unsubscribe`	*`listname address`*	Requests unsubscription for the specified address; also used to approve moderated unsubscription requests.
`newinfo`	*`listname password`*	Changes the list long description.
`mkdigest`	*`listname password`*	Forces distribution of the digest.
`set password`	*`password`*	Sets the password for use with other commands.
`rejectuser`	*`listname address`*	Rejects moderated subscription requests.
`approvemail`	*`listname message#`*	Approves moderated messages.
`rejectmail`	*`listname message#`*	Rejects moderated messages.
`end`		Marks the end of the commands included in the request message. This prevents the system from attempting to process text in the message signature.

Again, consult the *Post.Office User's Guide* and *Post.Office List Owner's Guide* for information on using these e-mail commands to execute specific mailing list operations.

8

System Monitoring

This chapter is intended to assist you in the ongoing maintenance of your Post.Office mail system. The topics discussed in this chapter include:

- Receiving and responding to notifications of error conditions
- Processing messages that are awaiting delivery (known as queued mail)
- Accessing and cleaning up POP3 mailboxes
- Using log files to track system activities and performance

8.1 Error Conditions

Post.Office will generate an error message to the Postmaster on any occasion that it cannot carry out its tasks of sending, delivering, or returning mail. Most error messages will involve addresses that Post.Office is unable to process, either because they are not entered correctly or because they do not exist, but many conditions can produce such errors. It is the job of the Postmaster to determine the cause of the error and the appropriate response to it.

There are two kinds of error messages that are sent to the Postmaster: notification messages and action messages. Notification messages are simply informative messages that alert you to the problem so that you can take corrective action, while action messages are e-mail forms that require the Postmaster to resolve the error in some way. This means that while both types of messages indicate an error condition that should be investigated by the Postmaster, action messages require prompt attention – they indicate an error condition that Post.Office could not resolve itself, or which you specifically requested should be held for your intervention. Error conditions reported by action messages will exist until personally resolved by the Postmaster.

8.1.1 Types of Errors

The following error conditions will cause a notification or action message to be sent to the Postmaster:

- **Unknown User**. This error results when a message is sent to a user in your domain, but its destination address did not match any address in your Post.Office accounts database. This commonly happens when users mis-type an address or when an account has been deleted. Since typos are a serious epidemic among computer users, this is by far the most common error.

- **Unreturnable message**. This error occurs when a message needs to be returned to sender, but the **From:** address of the message is itself an unknown address. This can happen with auto-reply accounts, which – in accordance with Internet standards – technically do not have a return address. If an auto-reply message is returned, you will get this error. This can also happen if the DNS records of the sender's mail host are configured incorrectly.
- **Maximum MTA hops exceeded**. This one indicates that a message may be caught in a mail loop. If the number of mail servers that a message has passed through (that is, its number of "hops") exceeds the Postmaster-defined limit on MTA hops, Post.Office will hold the message and consider it undeliverable.
- **POP mailbox over-quota**. When a user is getting mail via POP3 delivery, you can set a limit on the size of their POP mailbox. If the amount of mail in the user's mailbox reaches this limit, any new messages for that account generate an error; the message is returned, and the sender and Postmaster are notified.
- **Security violations**. Because it does not run with root privileges, users cannot hack into the server system by going through Post.Office (as they can with other mail servers, such as sendmail). However, just because users can't do it, that doesn't mean that users won't *try* it. Post.Office looks for this type of tampering and reports it to the Postmaster.
- **Impending mail loop**. In the DNS, a single host can go by many names. This means that Post.Office may attempt to send a message to what appears to be a remote host, but which turns out to be its own host by another name. If Post.Office actually attempted to send such a message to itself, a mail loop would result; instead, the message is stopped in its tracks and the Postmaster is notified.
- **Insufficient permissions**. To deliver a message to a user's mailbox, Post.Office must have write access to the mailbox directory on the server file system. This access is set appropriately at installation time, but may later be accidentally changed by users or other applications in such a way as to prevent Post.Office from writing messages to the correct directories.

8.1.2 Setting Error Handling Options

Post.Office allows you to set a few options that affect the handling of error conditions. You can access the form for setting these options from the Status of Deferred Mail menu, which is itself displayed when you click the **Deferred Mail** menu button at the left of any menu screen. The Status of Deferred Mail menu looks like the following:

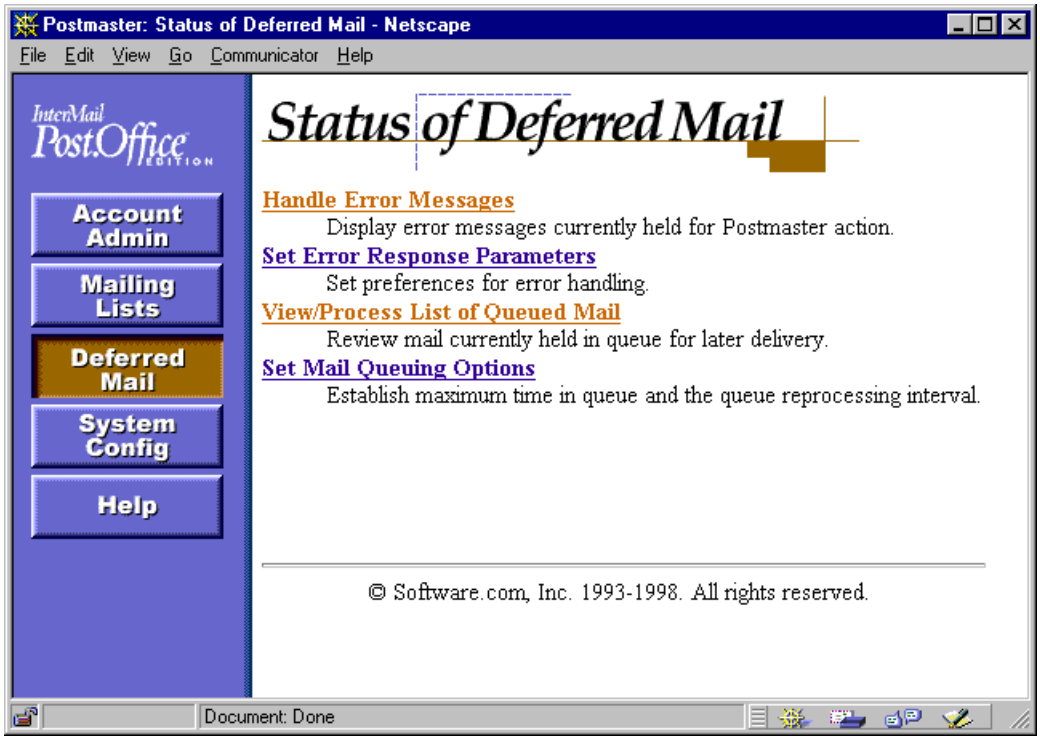

Figure 8-1: Status of Deferred Mail menu

Error Handling options are in the Error Response Parameters Form. This form was introduced in Chapter 4, and can be invoked from both the System Configuration menu and the Status of Deferred Mail menu. From the latter, this form is invoked by clicking on the **Set Error Response Parameters** link:

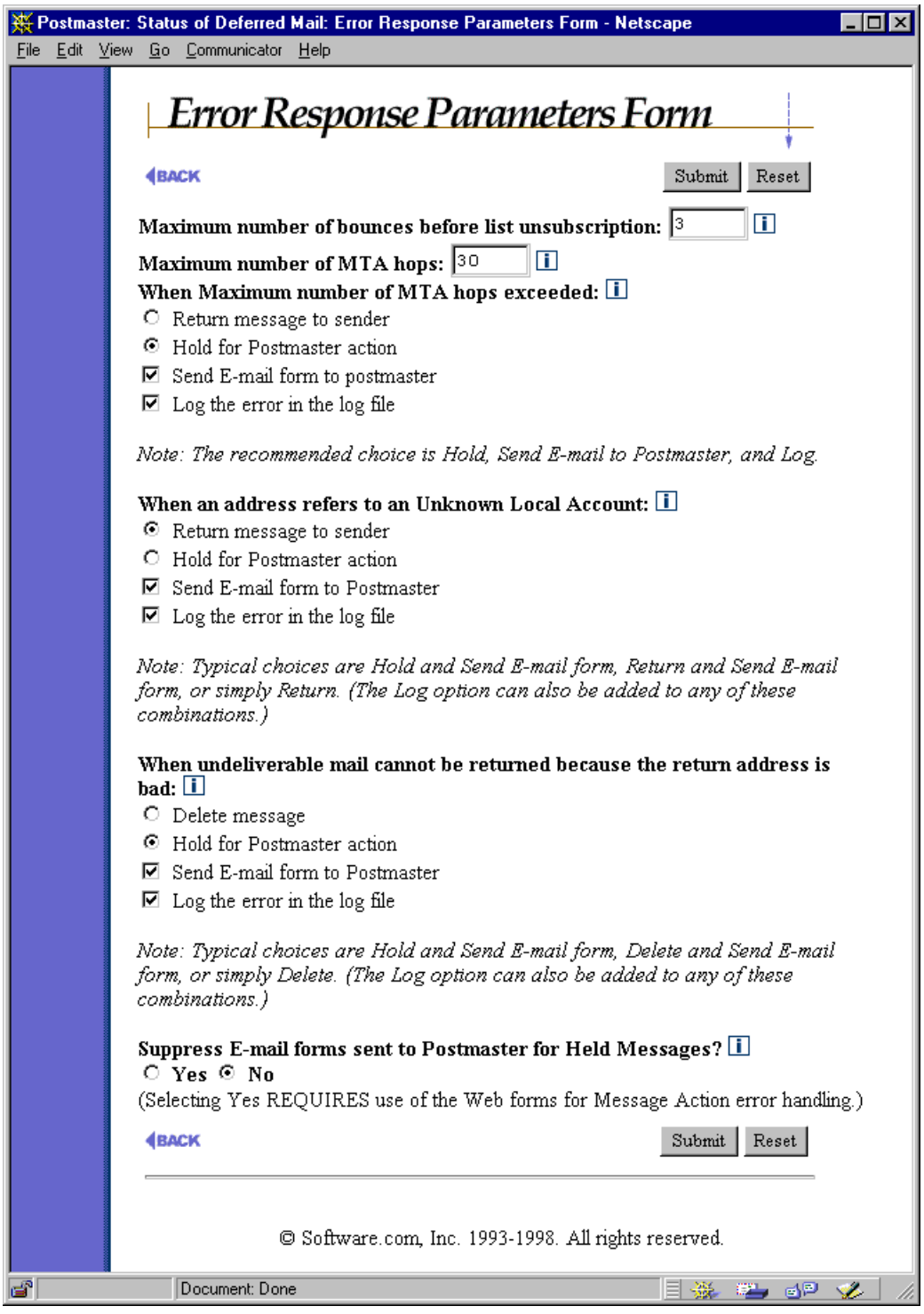

Figure 8-2: Error Response Parameters Form

This form was described in detail in Chapter 4. Since we're now concerned specifically with error messages, the following options are worth highlighting again:

- **When Maximum number of MTA hops exceeded**. This option defines the action taken by Post.Office when an incoming message exceeds the limit on MTA hops defined in the Maximum MTA Hops field. The sending of error messages to the Postmaster is controlled by the **Send E-mail form to Postmaster** check box. Since this error condition probably indicates a mail loop, requesting to return such a message would just continue such a loop, so the highly recommended selection for this error condition is **Hold for Postmaster action**.
- **When an address refers to an Unknown Local Account**. The option in this field that controls Postmaster notification is the check box field labeled **Send E-mail form to Postmaster**. When this check box is enabled, any Unknown User error will result in a message being sent to the Postmaster. This message may be either an action message or simply a notification, depending on whether the current selection for these errors is **Return message to sender** (notification) or **Hold for Postmaster action** (action message, naturally). Also, if the appropriate log option is enabled, the Unknown User condition will be recorded in the daily log file.
- **When undeliverable mail cannot be returned because the return address is bad**. This error means that Post.Office attempted to return a message to its sender (for example, in the case of a message addressed to an unknown local user), but could not because the return address of the original message does not exist. Since this error essentially indicates that the message can't be handled, the typical selection for this field is **Delete message**, which causes the message to be permanently deleted from Post.Office.
- **Suppress E-mail forms sent to Postmaster for Held Messages?** This option prevents action messages from being sent to the Postmaster. By default, the selection for this field is **No**, and the Postmaster is sent an action message whenever an error condition requires Postmaster intervention. Since errors that generate action messages can be handled in the web-based error handler, action forms are redundant if the Postmaster regularly logs in to the web interface to handle all errors. However, because such action messages are good reminders to the Postmaster that something requires their attention, enabling this option falls into the "available but not recommended" category.

8.1.3 Notifications

Many messages don't require any action on your part, and are sent simply to advise you that something happened. They typically warn you of an error condition, but are occasionally provided for your information only.

For example, if your Error Response Parameters are set to notify you for the Unknown User condition, Post.Office will let you know that someone tried to send a message to your domain to an address that didn't exist. In most cases, your reaction to this sort of error message will probably be something like "so what?" However, sometimes it's worth

paying attention to such messages. For instance, if you get a deluge of stuff for something like `help@your.domain`, it's a good bet that the people trying to get messages through to this address are customers or potential customers, and this would be a hint for you to set up an account for this address.

Another notification message might tell you that somebody tried to exploit a sendmail vulnerability to try to break into your system. While such a message is sent to you as a notification only, it is a good idea to pay attention to it – someone is probably trying to break into your network. This would at least require you to perk up your ears and check for any other signs of a security breach.

Notification messages are sent to the Postmaster from the Error Handler account, and can be distinguished from action messages by their subject ("Notification"). The following is a sample notification which alerts you to the most common type of error, the Unknown User condition:

```
The mail system on sparky.software.com encountered the following
error:

    The following destination addresses were unknown (please check
    the addresses and re-mail the message):


    SMTP <tipe-oh@software.com>

The original mail envelope addresses are:

    User-From: SMTP<jack.flash@software.com>
    Recipient: [<tipe-oh@software.com>]

------------------------------------------------------------
The message was submitted on Wed, 5 Mar 1997 12:27:30 -0800
by host [10.2.21.3] [10.2.21.3]

The original message header is below:
...
```

Figure 8-3: Unknown User notification

Again, notifications require no response on your part – they're just the way that your friendly neighborhood Post.Office keeps you up to date on its goings on.

8.1.4 Action Messages

Sometimes its not enough to simply notify the Postmaster of an error. Several error conditions, such as undeliverable or unreturnable messages, require some type of intervention by the Postmaster. Instead of simple notifications, these errors produce action messages; that is, they send e-mail forms to the Postmaster so that you can take appropriate action (get it? "action" messages). These messages are also known as Message Action Forms.

Note: *All errors reported to you by action messages can be handled via the web interface as well as by submitting Message Action Forms. See Section 8.1.5 for information on using the web-based error handler.*

Along with the Unknown User and Maximum MTA hops errors (which can produce action forms at you request), the following error conditions produce action forms:

- **Unreturnable message**
- **Impending mail loop**
- **Insufficient permissions**

See Section 8.1.1 for descriptions of these errors.

Like notifications, action messages are sent to the Postmaster from the Error Handler account. They can be distinguished from notifications by their subject: "Post.Office Message Action Form." The following is a sample Message Action Form for the Unknown User condition:

```
The mail system on sparky.software.com encountered the following
error:

    The following destination addresses were unknown (please check
    the addresses and re-mail the message):


    SMTP <jonn.doee@software.com>

Options for this mail message are:

                  Action: [] (Delete,Return,Resubmit)
    Postmaster-Password: [] (Required for any action)

The original mail envelope addresses are:

    User-From: SMTP<typo.man@megahuge.com>
    Recipient: [<jonn.doee@software.com>]

------------------------------------------------------------
The message was submitted on Tue, 4 Mar 1997 17:48:03 -0800
by host [10.2.101.32] [10.2.101.32]

The original message header is below:
...
```

Figure 8-4: Unknown User action form

It's important for you to know that action messages are *not* like notifications – they aren't just for information, they require you to *do* something. For instance, in the case of the

Unknown User condition, failing to respond to the action message means that the undeliverable message will sit around remaining undeliverable.[49]

Responding to action messages is similar to using other Post.Office e-mail forms: create a reply to the action message which includes the entire form, enter or modify a few pieces of information, and send it. Message Action Forms allow you to choose one of three actions for a problem message: Delete, Return, and Resubmit. To specify an action to be taken on a message, enter the appropriate action between the [square brackets] next to the `Action:` label.

Note: *If you're going to use Message Action Forms for handling errors, you must turn off the "Quoted Printable" option in your mail client (if it supports this option). Responding to and submitting e-mail forms may fail if this mail client option is used.*

The Return and Delete options are reasonably self-explanatory, but as for the Resubmit, you may wonder why you'd ever ask Post.Office to retry an operation that failed the first time. This certainly won't be of much help if the message or error condition stays as-is, but action forms also allow you to modify the destination address of the message's envelope. This means that you can "rewrite" the destination address of an undeliverable message (or the return address of unreturnable messages) before resubmitting it for delivery.

Note: *Only the envelope addresses can be rewritten in a Message Action Form – the To: and From: addresses in the message headers cannot be modified.*

49 Such undeliverable Unknown User messages are finally returned to sender after four days of sitting around collecting digital dust, meaning the sender spends the better part of a week thinking that his/her e-mail was successfully sent. If you've ever expressed frustration that the postal service took an extra day to deliver a letter or package to you, keep that in mind when you receive Unknown User action messages – nobody likes it when their "instant" electronic mail gets lost for four days.

Resubmitting a Message

The following is a sample reply to the Unknown User action form shown in Figure 8-4. Notice that this message includes a keyword for processing the message (in this case, `resubmit`), as well as the Postmaster password (required for submission) and a new destination address.[50]

```
> The mail system on sparky.software.com encountered the following
> error:
>
>     The following destination addresses were unknown (please check
>     the addresses and re-mail the message):
>
>
>     SMTP <jonn.doee@software.com>
>
> Options for this mail message are:
>
>                  Action: [resubmit] (Delete,Return,Resubmit)
>     Postmaster-Password: [$ecret] (Required for any action)
>
> The original mail envelope addresses are:
>
>     User-From: SMTP<typo.man@megahuge.com>
>     Recipient: [<john.doe@software.com>]
>
> ------------------------------------------------------------
> The message was submitted on Tue, 4 Mar 1997 17:48:03 -0800
> by host [10.2.101.32] [10.2.101.32]
>
> The original message header is below:
>
...
```

Figure 8-5: Response to Unknown User error: Resubmit

Note: *When correcting the destination address of the held message, be sure to correct the address on the* `Recipient:` *line, not the* `SMTP` *line at the top of the form. This is a common mistake.*

When this action message reply is sent to the Error Handler account, Post.Office will attempt to resend the message with the new envelope information. If the new address leaves the original message similarly undeliverable, you'll receive a new Message Action Form that says so.

50 Remember to modify the destination address on the "Recipient:" line, not the "SMTP" line at the top of the form.

Returning a Message

If instead of correcting the address and resubmitting the message you wanted to return the original message to its sender, your Message Action Form reply would look like this:

```
> The mail system on sparky.software.com encountered the following
> error:
>
>     The following destination addresses were unknown (please check
>     the addresses and re-mail the message):
>
>
>     SMTP <jonn.doee@software.com>
>
> Options for this mail message are:
>
>                   Action: [return] (Delete,Return,Resubmit)
>     Postmaster-Password: [$ecret] (Required for any action)
>
> The original mail envelope addresses are:
>
>     User-From: SMTP<typo.man@megahuge.com>
>     Recipient: [<jonn.doee@software.com>]
>
> ------------------------------------------------------------
> The message was submitted on Tue, 4 Mar 1997 17:48:03 -0800
> by host [10.2.101.32] [10.2.101.32]
>
> The original message header is below:
>
...
```

Figure 8-6: Response to Unknown User error: Return

Notice that in this form submission, the `Action:` field contains the word `return`. The original destination address (`jonn.doee@software.com`) hasn't been corrected, but since the message is being returned to sender, its destination address is irrelevant.

Deleting a Message

Finally, if you want to simply delete the message without attempting delivery or return, your Message Action Form would look just like the above return, but would contain the word `delete` in the `Action:` field.

Note: *Because deleting a message notifies neither the sender or intended recipient, it is recommended that you use the delete option only as a last resort when all other means have been exhausted. After all, you don't want your users to think that their mail is simply disappearing into a black hole.*

When Action Messages Go Unanswered

Failure to respond to an initial error message results in delivery of a second notice three days after the original notice. The second action message is labeled with the word RENOTIFICATION in the subject line. Pay particular attention to these messages, as they identify error conditions that remain unresolved despite previous notification. Failure to respond to a renotification does not produce any further warnings.

The importance of prompt response to error conditions cannot be overemphasized. It is one of the primary responsibilities of the Postmaster. Remember, when a message is held for error action, neither the sender nor the recipient is informed that the transaction remains incomplete. Also, messages that are both undeliverable and unreturnable will be left forever like cholesterol deposits clogging the arteries of your server file system, and nobody wants that.

8.1.5 Handling Errors Via the Web

The web-based interface for handling error messages is similar in many ways to the Message Action Form for handling errors via e-mail described in the previous section. However, this web interface has a number of advantages over these e-mail forms, most notably the ability to handle groups of messages in a single operation. The ability to view all error mail currently held by Post.Office in a single web form is another advantage.

The web form for dealing with error mail is the Error Message Handler Form. This form is invoked from the Status of Deferred Mail menu (Figure 8-1) by clicking the **Handle Error Messages** link.

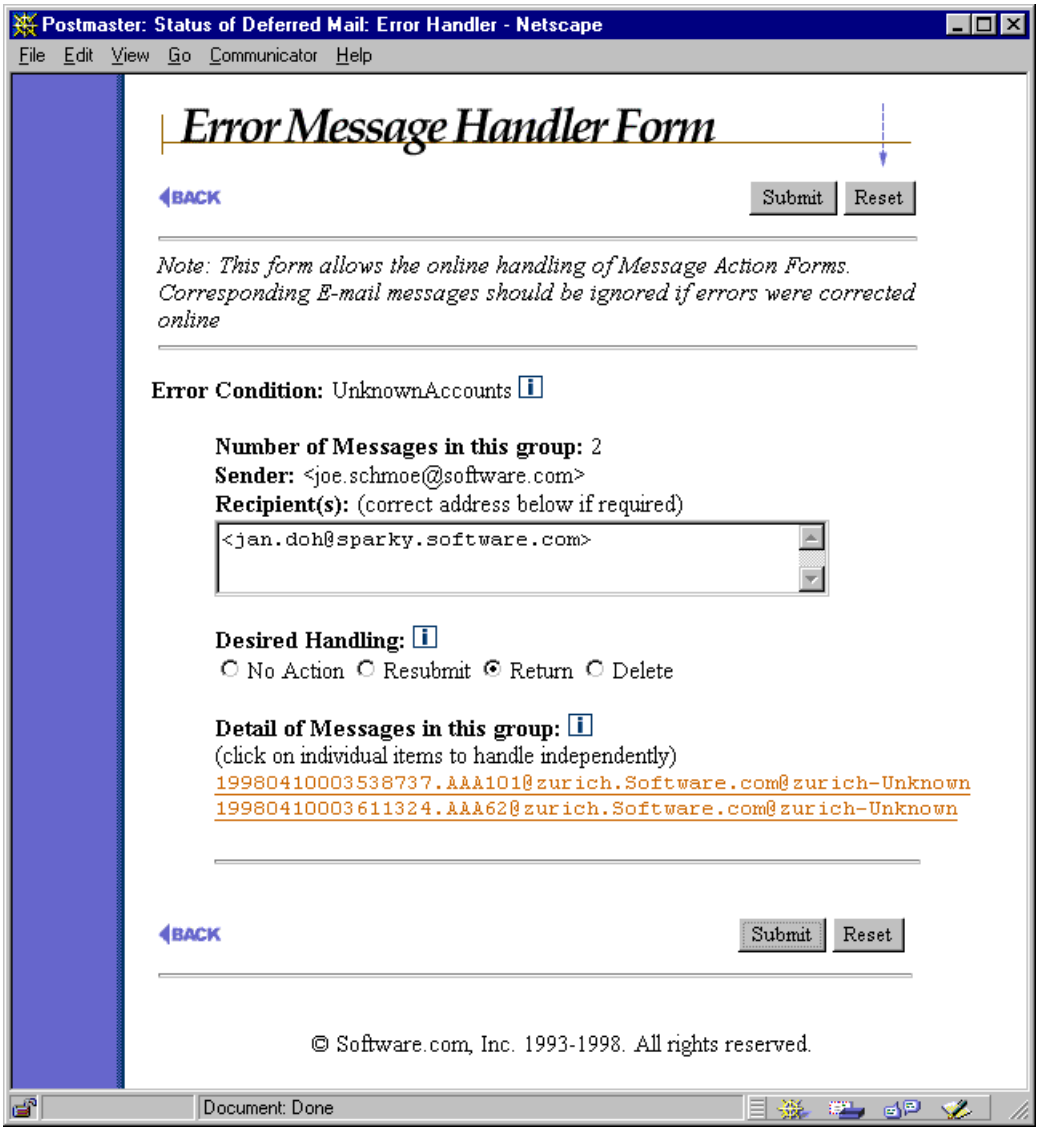

Figure 8-7: Error Message Handler Form

Although it bears little resemblance to the ASCII text Message Action Form, this web form contains exactly the same options. To handle an error message, correct the destination address in the **Recipient(s)** field (if desired), choose the appropriate radio button for the action from the **Desired Handling** field (Resubmit, Return, or Delete), and submit the form.

Note that in this example two error messages are grouped together in a single entry. Because many error messages may accumulate that have the same sender, recipient, and error condition (for example, one user who sends a stack of messages to an invalid address), grouping these messages on the Error Message Handler Form gives you a simple way to respond to similar error messages with one operation. You may apply a

single error handling option to all items in the group, or select each individual item for independent handling.

To view and/or handle an individual message, click on the message's identifier (the long name with all the numbers). This link is located under the heading **Detail of Messages in this group** at the bottom of the message group, and displays a Held Message Form, which looks like the following:

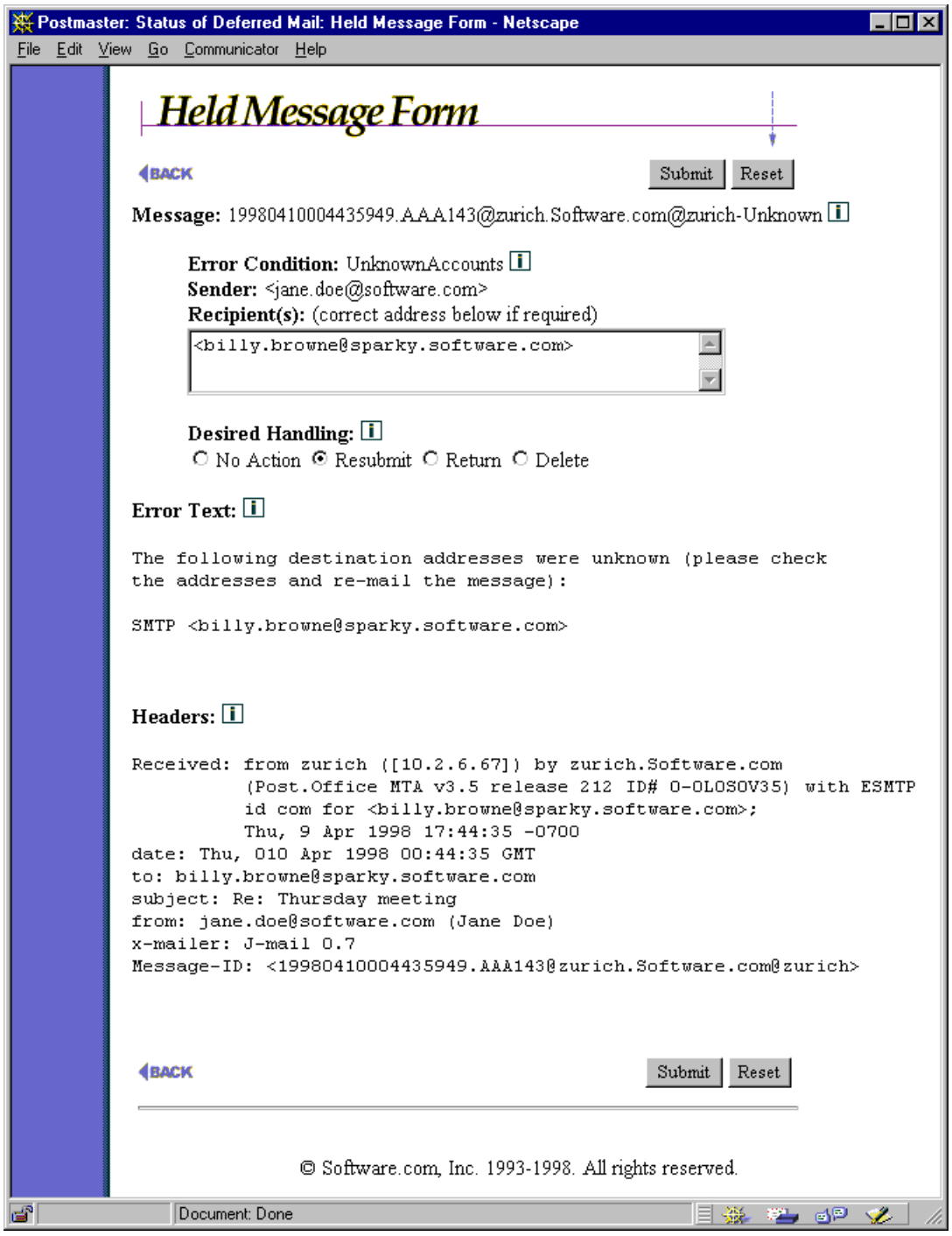

Figure 8-8: Held Message Form

This web form is very similar to the Message Action Form – it allows you to view a description of the error and the headers of the original message, modify the destination address, and select an action to be taken (Resubmit, Return, Delete). Like the Error Message Handler Form, you can execute your selected message handling action by submitting the form.

8.2 Queued Mail

During its daily routine, Post.Office will often find that it can't immediately send the mail that it has been asked to deliver to other mail servers.[51] However, provided that the destination mail host exists, Post.Office doesn't just return the mail; it assumes that the destination host's mail server is temporarily unreachable, so it "holds" the outgoing messages to that host and will try again later. Messages held in this manner are known as *queued mail*, since they sit in Post.Office's mail queue awaiting subsequent delivery.

Note: *Your intervention is not required for Post.Office to resend messages waiting in its mail queue. They will automatically be resent after a Postmaster-definable time interval, so you typically never have to worry about queued mail.*

8.2.1 When a Message Gets Queued

An outgoing message can be queued for one of many reasons. When displaying the list of queued mail, Post.Office also displays the reason why the message was not immediately delivered. The following are the possible reasons for a message being queued for later delivery:

- The **Always Defer Delivery to Remote Hosts** option is enabled. This option is available on the Mail Queuing Options Form, and is typically selected for sites with intermittent Internet access.
- Post.Office found the destination host, but was unable to establish an SMTP connection with it. Possible causes of this error include a network problem or an improper response.
- The destination server timed out. That is, Post.Office was talking to the sever, requested something, and never received a response.

51 A reasonably common occurrence, given the chaotic nature of the Internet and the slightly less than foolproof machinery that runs it.

- The destination server closed the SMTP connection, creating a deliberate break in the connection.
- An error occurred while looking up the mail exchange (MX) records of the destination domain's DNS server.
- Some other temporary server failure occurred.

Once a message is queued for one of the above reasons, all subsequent messages to that domain will be held until the next scheduled queue processing interval (typically one hour). That means, for example, that if you send a stack of 1,000 message to users in a particular domain, but that domain's mail server is temporarily off-line for some reason, Post.Office won't try to deliver all 1,000 – when the first message fails to be delivered, Post.Office will queue all 1,000 of them (as well as any subsequent mail for that domain) without attempting delivery, since it knows that this domain isn't ready to receive mail yet.[52]

Queued mail is organized into groups by destination host. When the queue processing interval has expired, Post.Office will attempt to deliver one from each group of waiting messages. If this attempt is still unsuccessful, the entire group remains queued; if successful – meaning the destination mail host is once again ready to receive message – Post.Office then delivers all of the messages that had been queued for that domain. You can force a delivery attempt before the queue process time expires (as described in Section 8.2.3), but manual intervention is not required.

The Importance of Checking the Mail Queue

The queuing of mail is generally harmless, and typically just indicates that the mail server hardware and software which other folks are using aren't quite the sturdy workhorses that yours are. However, depending on your configuration, the queuing of large numbers of messages could reveal a problem in your own network. For instance, in the case of a firewall system that simply pushes messages to an internal mail server system, if the mail queue of the firewall system has hundreds of messages queued up for the internal system then something is obviously amiss with your internal mail server.

For this reason, it can be a good idea to regularly check the list of queued mail (as described in Section 8.2.3). Depending on the amount of mail going through your system, any more than a handful of queued messages destined for a particular mail host could indicate a problem on the other end.

---

52 You can actually request Post.Office to always attempt delivery of messages, regardless of whether other messages are currently queued for the destination host. However, this falls firmly in the "not recommended" realm, because it can severely degrade server performance. See Section 8.2.2 for more information.

8.2.2 Setting Queuing Options

As noted above, Post.Office allows you to set options that affect the queuing of mail. Mail queuing options are set using the appropriately-named Mail Queuing Options Form, which is invoked from the Status of Deferred Mail menu (Figure 8-1) when you click on the **Set Mail Queuing Options** link.

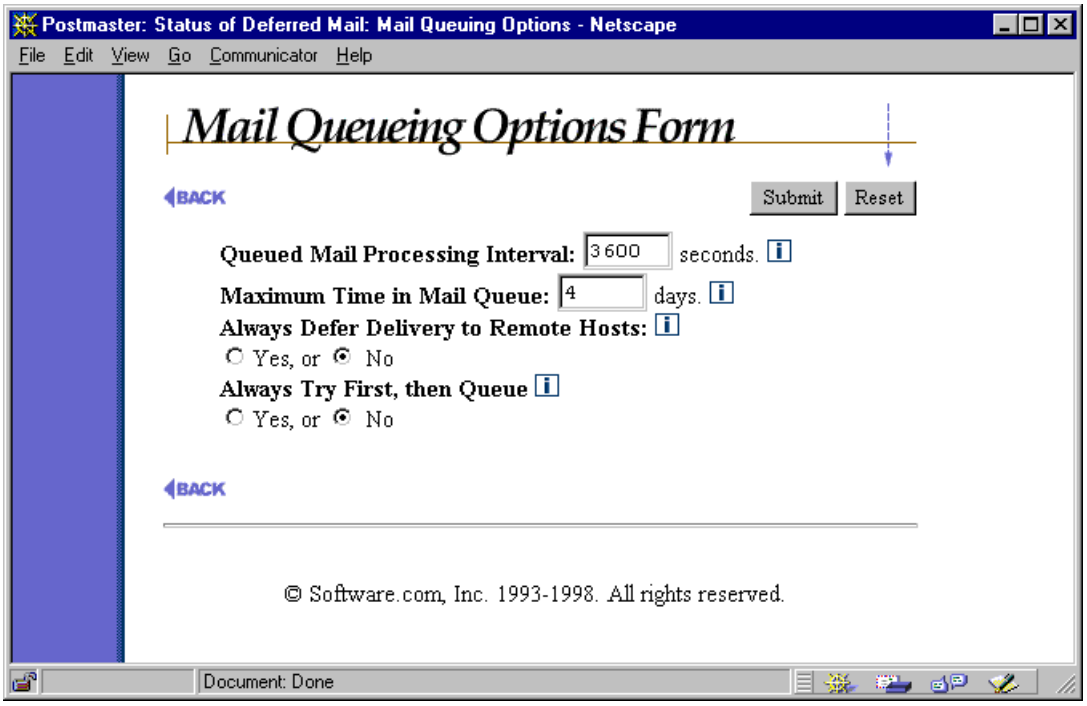

Figure 8-9: Mail Queuing Options Form

This form allows you to set the following mail queuing options:

- **Queued Mail Processing Interval**. As noted above, Messages that can not be delivered immediately are placed in a message queue for a later attempt. Just how much later that attempt is made is determined by the time you set in this field; when this number of seconds have elapsed, Post.Office will try again to deliver the message. Typical intervals are between 30 minutes (1800 seconds) and 2 hours (7200 seconds), with shorter intervals for speedier delivery, and longer intervals for better system performance.
- **Maximum Time in Mail Queue**. This parameter sets the maximum number of days that a message is allowed to sit in the queue for re-attempted delivery before it is finally returned. Internet standards recommend at least 4 or 5 days for this parameter, since messages that remain queued for that long probably aren't going anywhere. However, you may want to set a lower maximum time if you or your users have a more urgent need to know when mail has not been sent out.

- **Always Defer Delivery to Remote Hosts**. Normally when a message needs to be delivered to another machine over the Internet, Post.Office attempts to deliver it immediately and queues it only if there is a problem. However, enabling this option will cause Post.Office to instead queue all outgoing mail, and only attempt delivery when it processes the queue. You can specify how often the queue is processed by setting the Queued Mail Processing Interval.

  For sites with a full-time Internet connection, this option should be turned off because it causes delays in mail delivery. However, for sites that are not fully connected to the Internet (for instance, those that have a dialup PPP or SL/IP connection that is not always up), turning this on will prevent unnecessary attempts to deliver mail.

- **Always Try First, then Queue**. Enabling this option will cause Post.Office to attempt to deliver all mail immediately, even if other outgoing messages for the same host are already being held in the mail queue and awaiting the next scheduled re-processing interval. Use of this option is not recommended, since it bypasses that performance-friendly feature. This particular option is really only desirable on intranets which require the speediest possible mail delivery.

8.2.3 Viewing and Handling Queued Messages

Queued mail is generally not something that you need to worry about; once you set your system's mail queuing options as described above, you can just kick back and let Post.Office manage temporarily undeliverable mail. However, it can be a good idea to periodically check the list of queued mail to see if stacks of messages are piling up for another mail server. This can indicate a problem with the destination server, and if the machine is question is on your own network, then you should probably find out what's going on and correct the problem.

Post.Office allows you to view the list of currently queued messages in both the web and e-mail interfaces. In addition to just viewing the list, you can also specify that a group of messages should be processed (that is, Post.Office should attempt to immediately resend them) or expired (Post.Office should forget about delivery and simply return the messages).

The process option is useful for when you find the problem responsible for a stack of queuing messages and take corrective action; since you know that the destination server is back online, you can have that stack of mail delivered without waiting for the next queue processing interval. The expire option is useful when you find out that the destination server won't be online anytime soon; since it won't do Post.Office any good to try resending them, this allows you to return all undeliverable messages from the system so that no more valuable time and disk space is wasted on them.

The Web Form

The web interface form for handling queued mail is known as the List of Queued Mail Form. Like the Queuing Options Form, it is invoked from the Status of Deferred Mail menu (Figure 8-1). Click on the **View/Process List of Queued Mail** link to display the List of Queued Mail Form.

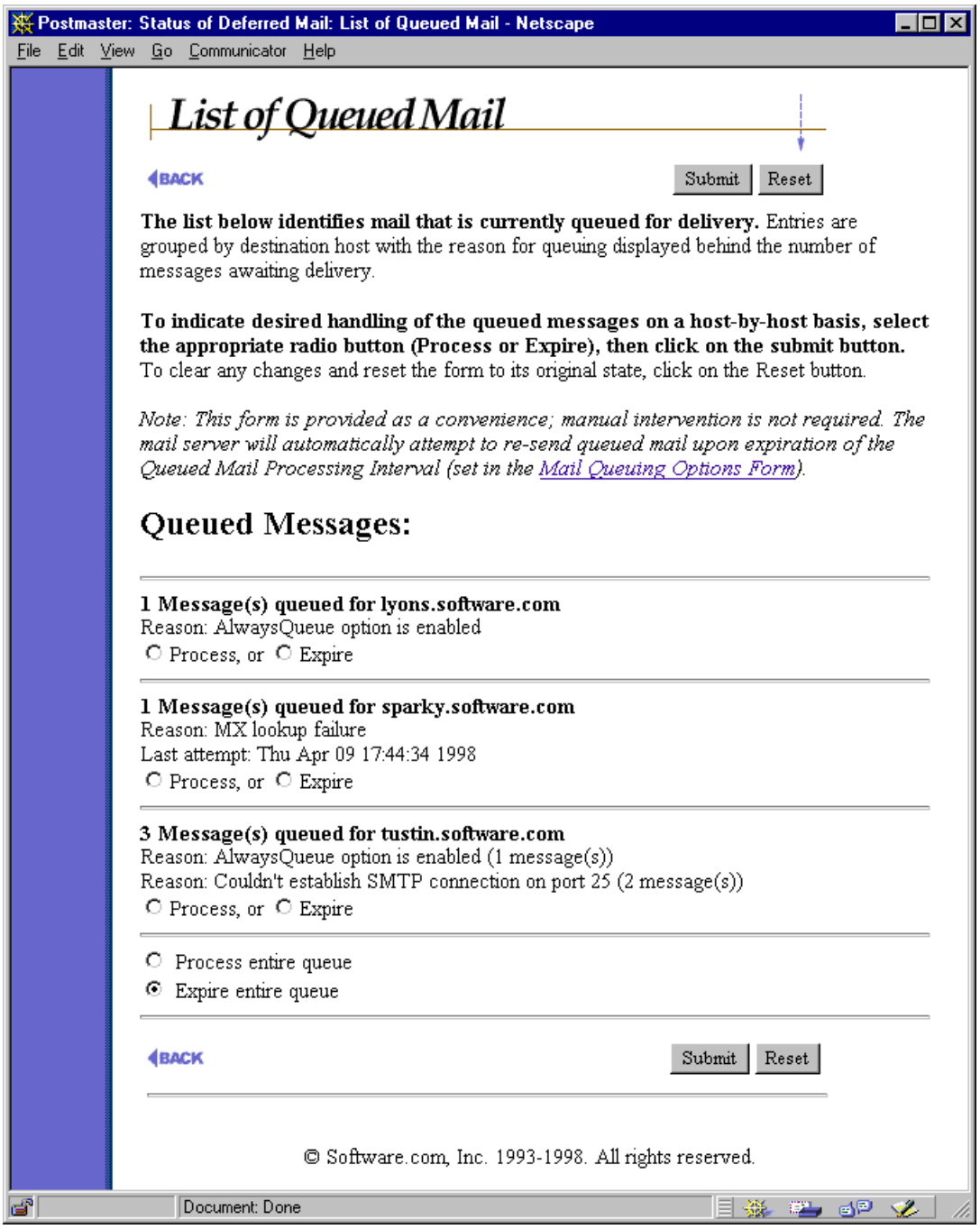

Figure 8-10: List of Queued Mail Form

All queued messages are grouped on this form by their destination host and/or domain. This means that a single entry on this form can represent multiple messages. Each entry includes the number of messages in the group, as well as the reason why the messages were not successfully delivered.

Each message group includes the radio buttons **Process** and **Expire**. To request immediate handling of the message group, select the appropriate radio button and submit the form. The Process option attempts to immediately deliver the messages, while the Expire option gives up on the whole group of messages and returns them to sender. Again, Post.Office will automatically attempt to resend at the end of the Queue Mail Processing Interval, and will automatically expire messages after the Maximum Time in Queue has been reached, so manual processing of the mail queue is unnecessary.

You can choose to process or expire all of the messages in the mail queue by selecting the **Process entire queue** or **Expire entire queue** options, respectively.

The E-mail Form

The Queued Mail e-mail form is nearly identical to the web-based List of Queued Mail Form shown above. It allows you to view the number of messages currently in the mail queue, the destination host to which they are supposed to be delivered, and the reason that Post.Office was unable to deliver the messages.

To request the Queued Mail Form, submit an e-mail message to your host's Configuration Manager address (`configuration@`*`host.domain`*) which contains the following keyword in the message body:

```
queue
```

The form returned to this command looks like the following:

```
The following is a list of Queued Mail on sparky.software.com

(Note: It may take several minutes for the queue to process or
expire for each host due to the time-out period for attempted
contact with unreachable hosts.  Please be patient when using this
form.  Additional information follows the Queued Mail list):

flakydomain.com: [] (process or expire) - 3 Message(s) queued.
  Reason:  Couldn't establish SMTP connection on port 25

some.other.com: [] (process or expire) - 1 Message(s) queued.
  Reason:  MX lookup failure

megahuge.com: [] (process or expire) - 189 Message(s) queued.
  Reason:  Server failed (MAIL)

...
```

Figure 8-11 Queued Mail Form

As with the web-based List of Queued Mail Form, you can choose to process or expire each group of mail in the queue. Create a reply message that includes the contents of the form, and enter the appropriate keyword (`process` or `expire`) in the [square brackets]

next to the destination domain. Send the form message to execute whatever mail queuing operations you specified, remembering as always to include the Postmaster password with the form.

Viewing the Mail Queue From the Command-Line (UNIX only)

In addition to the web and e-mail forms for viewing the mail queue, on UNIX platforms you can also use the `mailq` command to view this information. When this command is entered at the UNIX command prompt, the list of queued mail is displayed. In the following example, five messages await delivery:

```
% mailq
   Queued Messages    Destination Host
   ---------------    ----------------
            2          math.ucsb.edu
            3          megahuge.com
%
```

SMTP Queue Processing Requests (ETRN)

Yet another mail queue processing option is the `ETRN` command, which can be used to instruct remote mail hosts to attempt delivery of queued messages. Unlike `mailq` (described above), `ETRN` is an SMTP command; this means that it is executed by assorted mail programs during client connections to the Post.Office SMTP server. This is useful if you have a PPP or a SL/IP connection, or have a similarly intermittent connection to the Internet (or any other network you receive messages from).

Note: *ETRN is an open protocol standard, and is defined in RFC 1985. Post.Office also supports QSND, a propriety queue-processing command that was created before the existence of ETRN. QSND remains supported for backward compatibility.*

Although `ETRN` is designed to be used by connecting mail servers, you can also execute it manually to request queue processing. To do so, use the telnet utility to log in to port 25 of the Post.Office server system, and enter ETRN followed by "@" and the domain of the queue that should be processed. For example, the following telnet session requests delivery of the queued messages displayed in the `mailq` example above:


```
220 sparky.software.com ESMTP server (Post.Office v3.5 release
ID#0-0U1000L50S10000) ready Thu, 4 Dec 1997 12:42:38 -0800
HELO
250 sparky.software.com
ETRN @math.ucsb.edu
250 Ok
ETRN @megahuge.com
250 Ok
QUIT
```

8.3 Mailboxes

As discussed in Chapter 5, all e-mail accounts that use the POP3 delivery method have a mailbox on the server system. E-mail mailboxes are analogous to postal mailboxes; it is into this mailbox that messages are placed as they're delivered by Post.Office, and it is from this mailbox that messages are taken when requested by a mail client. Consequently, these mailboxes grow in size as e-mail is collected.

Because POP mailboxes consume disk storage space on the server, you should understand how they're stored, how to check their size, and how to clean them out if necessary.

8.3.1 How They're Stored

Mailboxes are simply directories on the server file system. All Post.Office mailboxes are stored in one of several directories in a central mailboxes directory. The location of the mailboxes directory is set at installation time, and can be viewed in the Licensing/Configuration Form. This form is invoked from the System Configuration menu by clicking on the **View Licensing/Configuration Information** link.

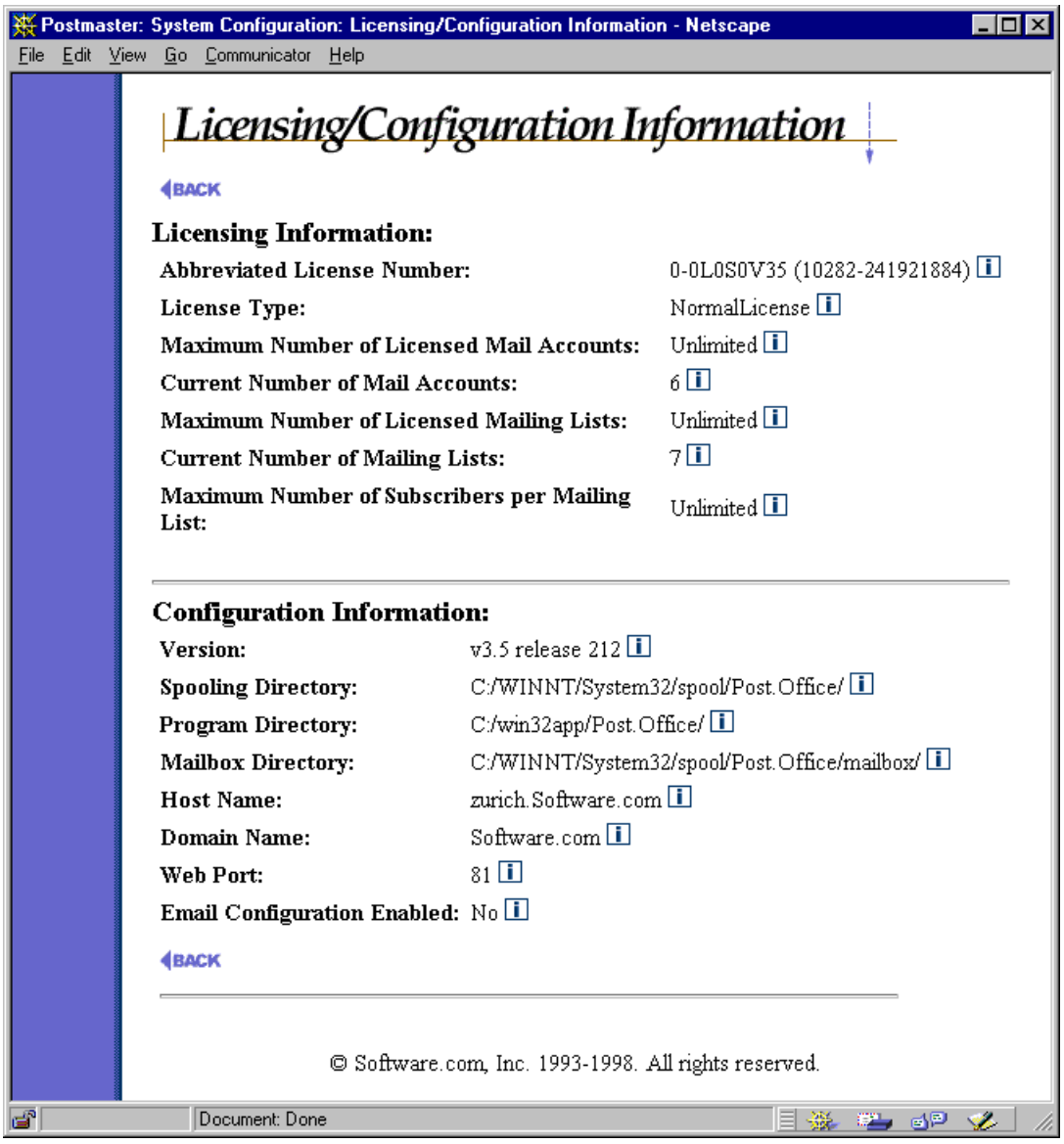

Figure 8-12 Licensing/Configuration Form

The location of the Post.Office Mailbox Directory is listed in the Configuration Information section of the form. Within this central mailbox directory are hundreds of numbered sub-directories (from 0 to 499) which contain individual user mailboxes. An individual mailbox directory has the same name as the account's unique identifier (UID), which is based on the Real Name specified for the account at time of creation. Both the UID and mailbox directory location of each account are shown on the Account Data Form (refer back to Chapter 5 for illustrations and descriptions of that particular form).

Note: *Viewing or modifying mailbox directories requires special login permissions. On UNIX platforms, you must be logged into the system as the Post.Office user. On NT, you must be logged in as the Post.Office user or a member of the administrator group.*

Within each mailbox is another directory named `in`; this is the directory that contains the actual message files. The `in` directory also contains a pair of files used by Post.Office to measure mailbox size and set a lock on the account. These files are named (respectively):

```
__size__
__size_lock__
```

The message files themselves are given unique names which include their date and time of receipt, and which look something like the following:

```
19970227080711159.AAI86@sparky.software.com
```

When you look in a typical mailbox, you'll see both the two size files and probably a few messages. The following is a directory listing for a mailbox on NT 4.0.

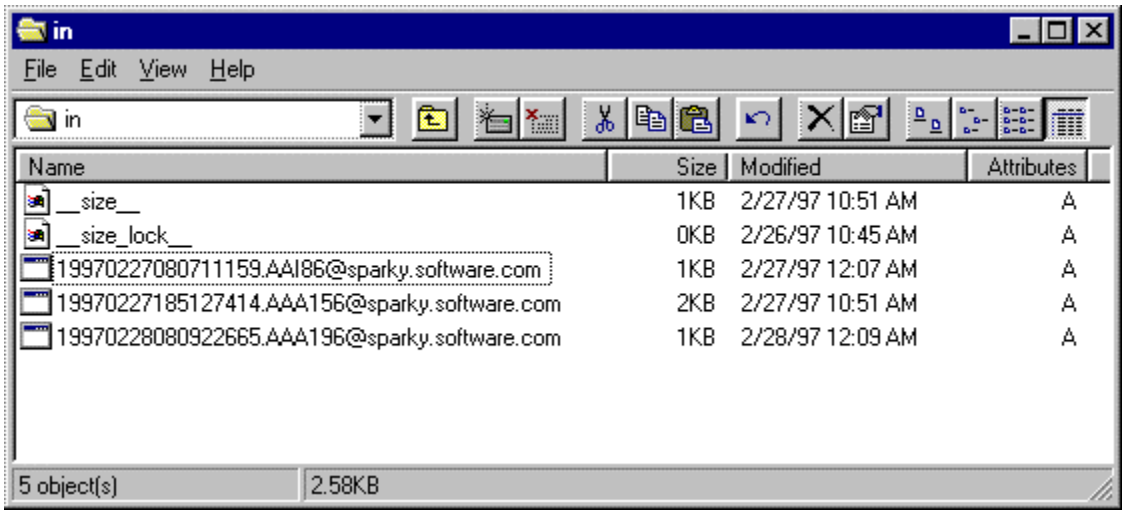

Figure 8-13 Contents of a typical mailbox

Although going through mailbox directories is *highly* discouraged, there are a few conditions in which it can become necessary or desirable for the Postmaster to do so. See Section 8.3.3 for more information.

8.3.2 Checking Mailbox Size

Because mailboxes consume storage space on your server system, it's certainly in your interests to keep track of their growth. As discussed in Chapter 5, you can set a limit for the maximum size of each mailbox (recall that when an account reaches its mailbox limit, any new messages sent to that account are returned to sender and the Postmaster is notified). But if you don't set account limits, or if you're concerned about the amount of storage being taken up by particular users, you can easily check the size of each user's POP mailbox.

Mailbox sizes can be most easily checked in the List of Accounts menu. This menu displays the Real Name and primary address of each account. By clicking on the **Show Quota** link in the General Accounts section of the menu, you can display mailbox usage information for each account; this provides a convenient way for you to check up on who's using up your disk space.

Recall that this menu is invoked from the Account Administration menu by clicking on the **List of Accounts** link, and looks like the following:

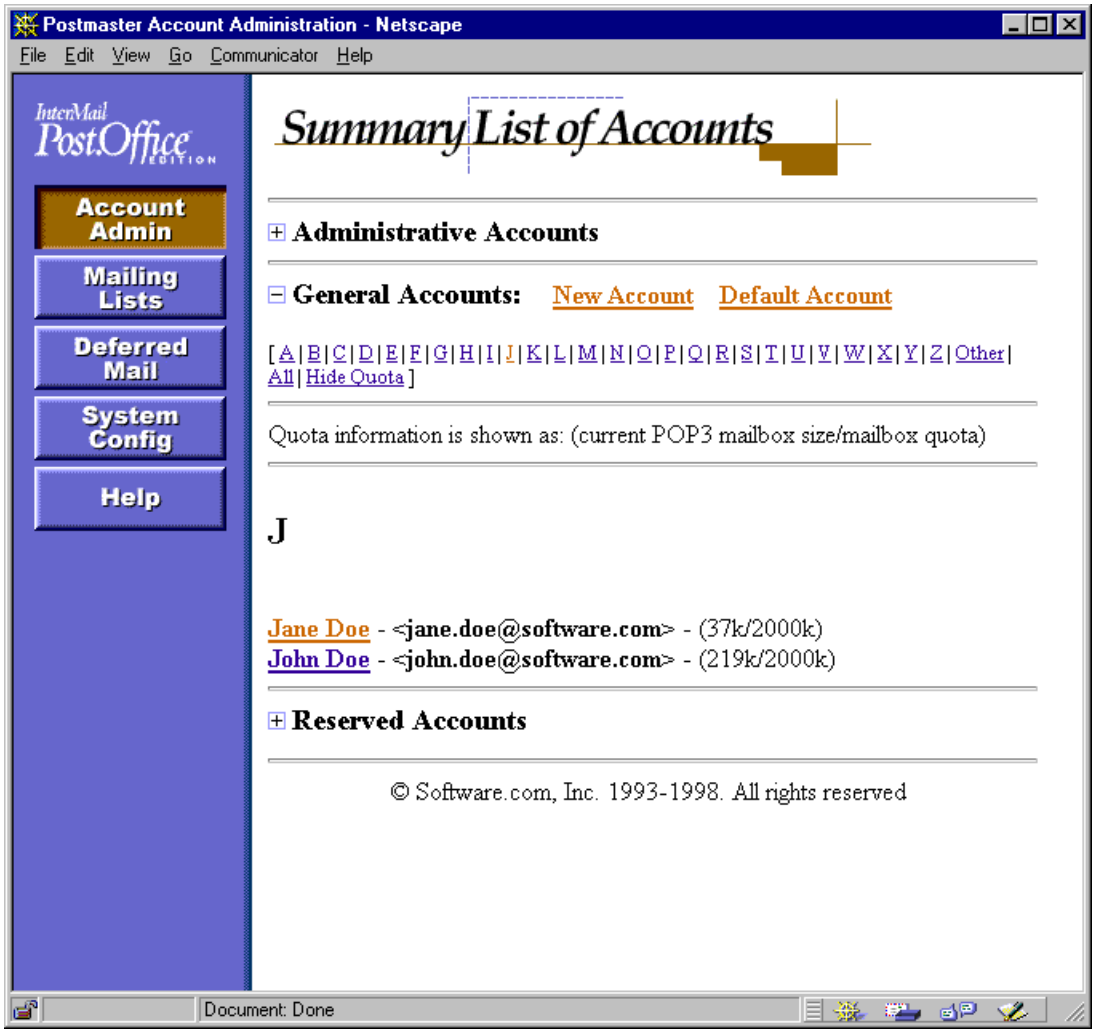

Figure 8-14 List of Accounts menu

If you only want to look at the mailbox usage numbers of a particular account, you can get this information from its Account Data Form.

Things to look for when viewing mailbox usage are accounts that are at or near their limit, and accounts that are taking up an unusual amount of storage space (especially those with no mailbox limit). Depending on what you find, you may decide that the mailbox limits of certain accounts should be raised or reduced. Remember, the idea is to provide your users with enough space to actually use their e-mail account, but not so much that your users can collectively max-out your storage capacity.

Hint: *It is generally a good idea to tell your users to not use the option provided by their mail clients to keep their messages on the server, since always using this option will fill up their POP3 mailboxes sooner or later. Alternatively, you can tell them to clean out all of their old messages after one or two weeks (or whatever time interval suits your storage capacity and personality).*

8.3.3 Cleaning Them Out

Poking around the files in mailbox directories on the server file system is a rare operation that strays pretty far into "not recommended" territory. However, under extreme circumstances, you may need to do this, so you should know how.

Most of the conditions that can require you to directly access a mailbox are all variations of the same problem: an account receives a message that the user's mail client can't retrieve. This can be caused by messages that are exceptionally large (especially for users downloading mail through a modem), or which contain special formatting (MIME encoding, etc.) that the user's mail client can't handle; a message can even become corrupted or have its file permissions changed in such a way as to prevent delivery. Such a message can shut down mail retrieval for that account, since no subsequent messages can be retrieved until the first one is removed.

Post.Office has no trouble handling messages of enormous size, so if you want to casually toss 10 Mb messages around your mail system, you certainly can. But many mail clients will experience problems when it comes time to receive such mega-messages, especially if the user is downloading their mail via a modem. So while enormous messages are not a problem for Post.Office, they can certainly pose a problem for your users.

Solving the problem of the oversized message is simple: go to the user's mailbox and delete the offending message (or at least move it out of the way). This is a little more difficult than it sounds, but the procedure breaks down like this:

1. Get the location of the user's mailbox directory, as shown on the Account Data Form.
2. On the server file system, go to that mailbox directory (remember that this operation requires that you have appropriate user privileges).
3. Find the message of unusual size – this will be the message that is causing the problem. If you find more than one, you might as well delete them all, since each will cause the same problem.
4. Delete the message(s) using normal file system commands.

In the following example, an account has somehow received a 25 Mb message (this scenario can be prevented by setting the limit for maximum message size described in Chapter 4, but for this example, let's assume that no limit exists). Since the user of this account is using a 14.4 kb modem to retrieve her mail, it would take her literally all day (and night) to get this message. When she discovers the problem, she contacts the Postmaster for help. The Postmaster looks up the mailbox directory location on the user's Account Data Form, and gets a listing of the files in this directory:

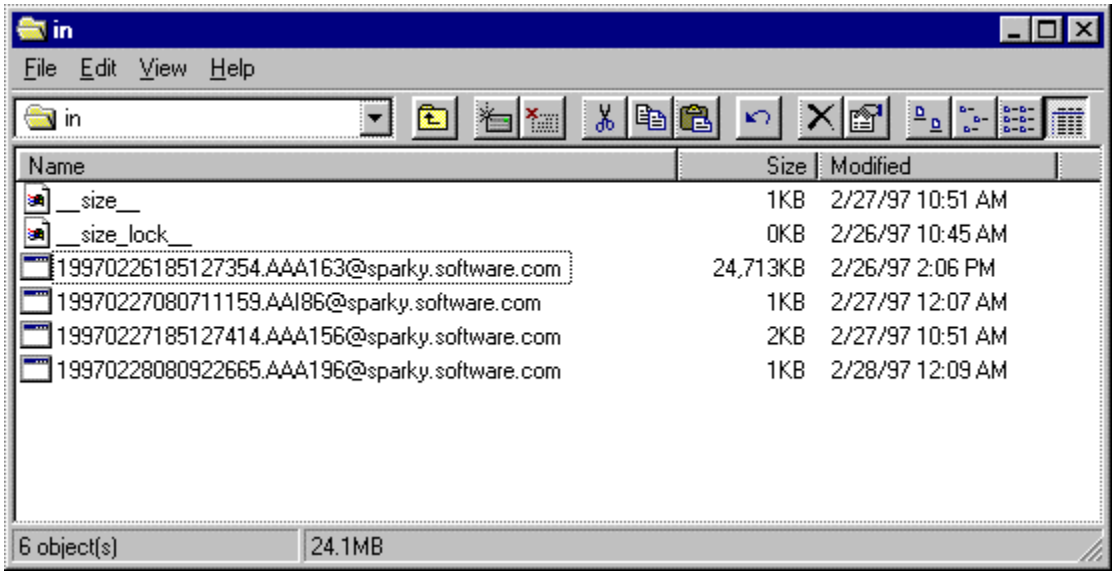

Figure 8-15 Listing of mailbox directory (NT 4.0)

It doesn't take the Postmaster very long to spot the problem – a gargantuan mega-message is blocking delivery of the other messages waiting for this user. By deleting this file, the Postmaster removes the roadblock and restores normal mail delivery.

8.4 Logging Information

This section discusses the details of the information recorded in the Post.Office logs. A *log* is a file that contains a running record of operations performed by Post.Office. Depending on how you have set your logging options, your daily logs can record every message arrival, every user request for POP3 delivery, and every mailing list distribution. These log messages are extremely valuable when debugging problems, since you can view the exact steps Post.Office went through when it encountered whatever problem you run into. Because many logging options include the time required to complete a transaction, logs can also be used to measure mail system performance.

All Post.Office log information for a specific day is kept in a single file and that file is named in the form of `post.office-####.log`, where #### represents the month and day. For example, the log file for April 15 has the name

```
post.office-0415.log
```


8.4.1 Setting Logging Options

The location and contents of log files can be set by the Postmaster in the Logging Options Form. This form is invoked from the System Configuration menu by clicking on the **Set Logging Options** link, and looks like this:

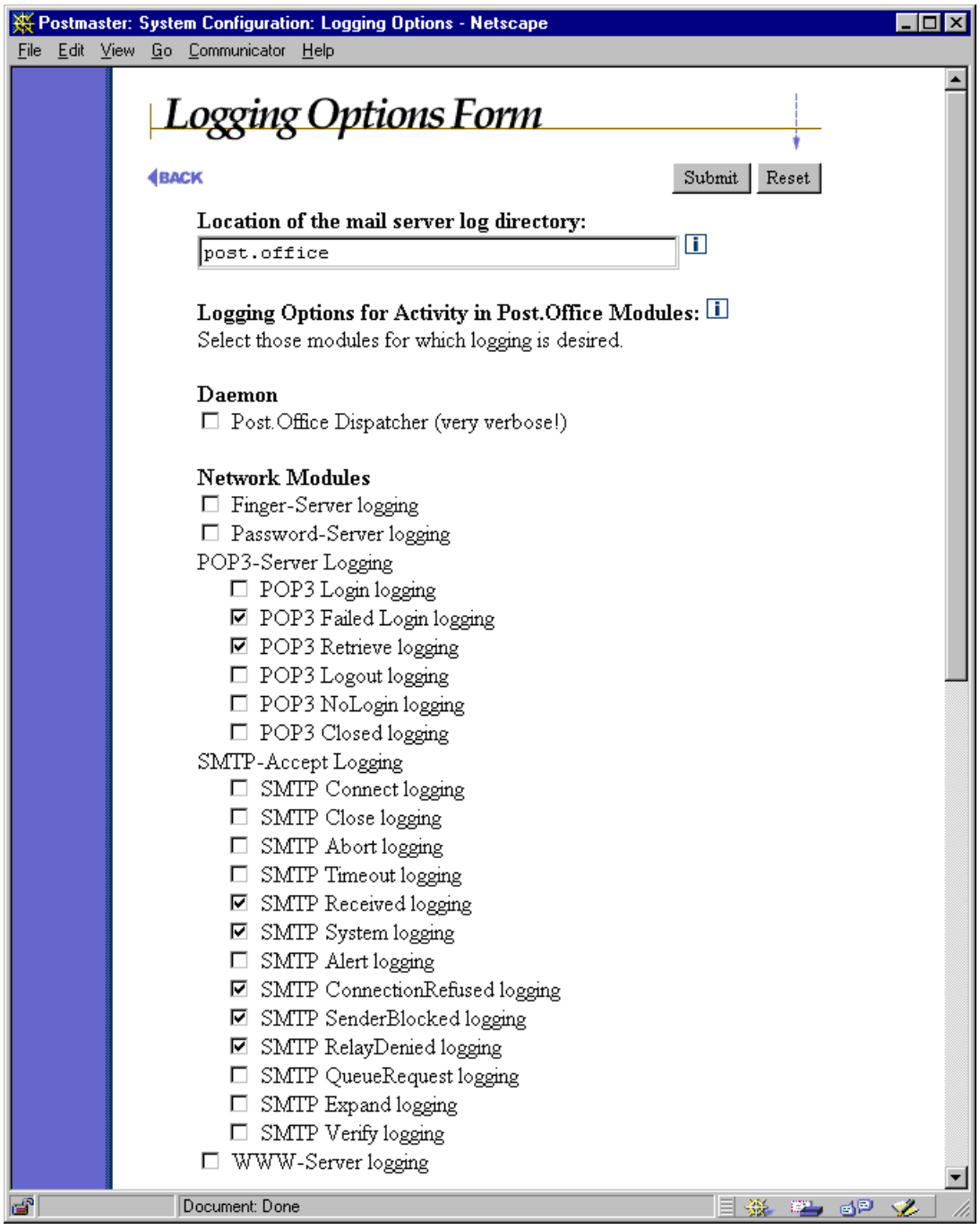

Figure 8-16 Logging Options Form (part 1 of 2)

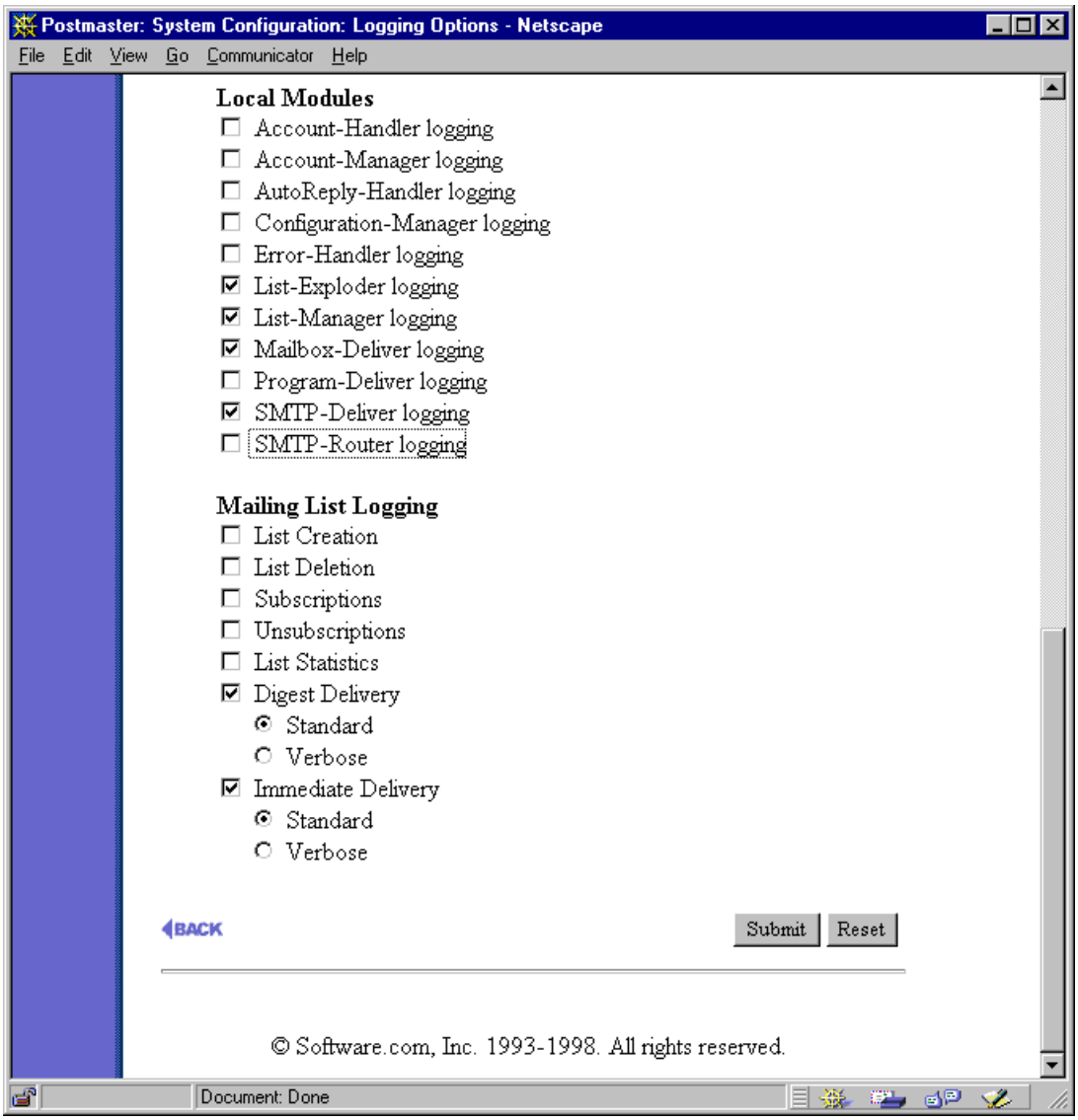

Figure 8-17 Logging Options Form (part 2 of 2)

By default, the log files are kept in the `post.office/log` directory, and we highly recommend that you leave it as the default. However, you can override the default location by specifying the full path of the directory where you want the logs to go in the text field labeled **Location of the mail server log directory**. Be sure that the permissions of the log directory you choose allow the Post.Office user to access the log files.

Besides logging directory location, the Logging Options Form allows you to determine exactly which Post.Office activities will be recorded in the logs. These activities fall into four general categories: Daemon, Network, Local, and Mailing List. The following section offers details on the individual logging options.

To turn on logging for one or more activities, enable the check box next to each of the desired logging options (and disable the check box next to unwanted options to turn them off). Changes take effect immediately after submitting changes to this form, so today's log file will begin logging whatever new options you requested.

8.4.2 Log File contents

Each module that writes an entry in the log file uses a format that is machine readable and can be used for automatic processing. Each entry consists of the current date and time, the module that recorded the information, and the module specific information. The date and time are given in the format YYYYMMDDhhmmss-OGMT: year, month, day, hour (00-23), minute, second, and offset from GMT time.

The following is a sample log snippet:

```
19970226001320-0800:SMTP-Accept:Connect:[10.2.21.3]

19970226001321-0800:SMTP-Accept:Received:[10.2.21.3]
:19970226081320963.AAA133@fido.software.com:6703:0
:<zack.taylor@megahuge.com>:<john.doe@software.com>

19970226001321-0800:SMTP-Accept:Close:[10.2.21.3]:1:1:6569

19970226005947-0800:Mailbox-Deliver
:19970226085946739.AAA176@sparky.Software.com:John_Doe
```

This may look cryptic, but it's actually quite simple. Four events are logged here:

1. On February 26, 1997 (`19970226`), at 12:13 a.m. PST (`001320-0800`), a mail client (or server) connected to the Post.Office SMTP server to give it mail (`SMTP-Accept:Connect`). The IP address of the connecting host is provided (`10.2.21.3`).
2. The message was successfully transmitted (`SMTP-Accept:Received`). Among other things, the sender of the message (`zack.taylor@megahuge.com`) and recipient (`john.doe@software.com`) are given.
3. The connecting program has no more mail to give at this time, so it closes its connection to Post.Office (`SMTP-Accept:Close`).
4. The message is delivered to the recipient's mailbox (`Mailbox-Deliver`). The name of the mailbox is given (`John_Doe`), as is the name of the actual message file that is written to that mailbox – that's the long string that starts with the date (`19970226`) and ends with the hostname and domain of the mail host (`sparky.software.com`).

See? Pretty straightforward, once you know what you're looking for.

Incidentally, long lines in the log file are not broken into more than one line as shown in this and other examples – this line wrapping is done here for presentation purposes only. There is one case where a single log entry can span multiple lines in the log file, and that is when the error handler module records the contents of a message's control file. In this case, each line of the multi-line entry is indented with a tab character, so an automatic log file parser should be able to easily detect them.

8.4.3 Available Logging Options

The following sections contain information on the various Post.Office logging options that are available. Practically everything that Post.Office does can be recorded to a log file. However, you probably won't record *everything*, since most of this information is not useful on a day-to-day basis; depending on the amount of activity on your server, it also can create enormous log files and degrade server performance. But many logging options are useful for calculating server performance, and the others are nice to have it available in case you need to do some troubleshooting.

Note that in the format description for each logging entry, values shown between <angle brackets> are values that are replaced with actual data. Except where noted, these values do not literally appear between these brackets in the log files.

Post.Office Dispatcher

This is the mother of all logging options – it records practically every movement of binary bits into, out of, or even close to Post.Office. It is incredibly verbose and creates large, unwieldy log files that you probably won't enjoy reading through. This option should be used only for very low-level troubleshooting.

Finger-Server

The finger server records the connecting host's IP address (enclosed in [square brackets]), along with the name that requested. The format of this log entry is:

```
<date-time>:Finger-Server:[<client-IP>]:<name>
```

For example:

```
19951020123456-0800:Finger-Server:[234.56.78.90]:frank
```

Password-Server

This entry records transactions on port 106 related to a password feature used by the Eudora mail client.

POP3-Server

There are six different levels of logging for the Post.Office POP server. The **Login**, **Logout**, and **Retrieve** options can be used to determine connection time and resource usage, as well as the frequency with which users are contacting the server. The other three POP3 log options – **Closed**, **NoLogin**, and **Failed Login** – are useful for debugging when a user can't log in, or for detecting users trying to gain unauthorized access.

Login. This option records client connections to the Post.Office POP3 server. It includes the IP address of the connecting system (enclosed in [square brackets]), and the username of the account being accessed. The format of this log entry is:

```
<date-time>:POP3-Server:Login:[<client-IP>]:<pop-login>
```

For example:

```
19971022182344-0800:POP3-Server:Login:[123.45.67.89]:mike
```

Failed Login. This option records client connections to the Post.Office POP3 server which were unsuccessful. A POP login typically fails because the specified POP login name does not exist, or the specified password is incorrect. This log entry includes the IP address of the connecting system (enclosed in [square brackets]), the username given by the client, and the reason that the login failed.

The format of this log entry is:

```
<date-time>:POP3-Server:FailedLogin:[<client-IP>]:
      <pop-login>:<error>
```

For example:

```
19971022182324-0800:POP3-Server:FailedLogin:[13.45.6.9]:
      mike:BadPassword
19971022182347-800:POP3-Server:FailedLogin:[13.45.6.2]:
      jdoe:UnknownUser
```

There are three possible error conditions found in a `FailedLogin` log entry: `UnknownUser` (the POP login name does not match that of an existing account), `BadPassword` (the given password is not correct), and `AccessDenied` (the login data was correct, but the connecting client is not within the account's General Access Restrictions).[53]

Retrieve. This option records a log entry for each message downloaded by a user from the POP server. It includes the IP address of the connecting system, the username of the account being accessed, the size of the message (in bytes), the time required to retrieve the message (in seconds), and the return address of the sender (enclosed in <angle brackets>) as defined in the message's `Return-Path` header. The format of this log entry is:

```
<date-time>:POP3-Server:Retrieve:<pop-login>:<bytes>:<seconds>:
      <sender>
```

For example:

```
19971022182345-800:POP3-Server:Retrieve:mike:46390:12:
      <joe@foo.bar.com>
```

53 Technically, there's a fourth possible POP login error: `SystemFailure`. This error implies that Post.Office was unable to read account information from its database. However, since this should *never* occur, you won't be seeing any `SystemFailure` errors in your log files.

Logout. This option records normal closures of POP client connections. It includes the IP address of the connecting system (enclosed in [square brackets]), the username of the account being accessed, and the total connection time of the session (in seconds). The format of this log entry is:

```
<date-time>:POP3-Server:Logout:[<client-IP>]:<pop-login>:<seconds>
```

For example:

```
19971022182344-0800:POP3-Server:Logout:mike:42
```

NoLogin. This entry indicates that a POP connection was closed normally, but that no login occurred during the session. This could indicate that unauthorized users are attempting (unsuccessfully) to access mail on your system. It includes the IP address of the connecting system (enclosed in [square brackets]), and the total connection time of the session (in seconds). The format of this log entry is:

```
<date-time>:POP3-Server:NoLogin:[<client-IP>]:<seconds>
```

For example:

```
19971117182935-0800:POP3-Server:NoLogin:[10.3.83.19]:20
```

Closed. Similar to the POP3 Logout logging option, this option records unexpected closures of POP client connections. It includes the IP address of the connecting system (enclosed in [square brackets]), the username of the account being accessed, and the total connection time of the session (in seconds). The format of this log entry is:

```
<date-time>:POP3-Server:Closed:[<client-IP>]:<pop-login>:<seconds>
```

For example:

```
19971117183600-0800:POP3-Server:Closed:[10.3.83.19]:mike:17
```

SMTP-Accept

There are no less than 13 different levels of logging for the SMTP-Accept module, which is the Post.Office component responsible for receiving incoming messages. These SMTP-Accept logging options are the most important method of determining the mail system load, as they can record information on every message handled by Post.Office – their recipients, their size, the amount of time required to process them, and the return address of the sender. Also included among the SMTP-Accept logging options are entries that record information about messages rejected because of relay restrictions or mail blocking rules, as well as entries which indicate possible tampering.

Connect. This option records SMTP connections, from e-mail clients as well as other mail servers. It includes the IP address of the connecting system (enclosed in [square brackets]). The format of this log entry is:

```
<date-time>:SMTP-Accept:Connect:[<client-IP>]
```

For example:

```
19970226001320-0800:SMTP-Accept:Connect:[10.2.21.3]
```


Close. This option records the closing of SMTP client connections. It includes the IP address of the connecting system (enclosed in [square brackets]), the total connection time (in seconds), the number of messages sent during the connection, and the total number of bytes sent. The format of this log entry is:

```
<date-time>:SMTP-Accept:Close:[<client-IP>]:<seconds>:
      <#-messages>:<bytes>
```

For example:

```
19970226001321-0800:SMTP-Accept:Close:[10.2.21.3]:1:2:6569
```

Abort. This option is identical to the **Close** option above, but indicates that the client aborted its connection. For example:

```
19971118001632-0800:SMTP-Accept:Abort:[10.2.85.88]:11:0:6
```

Timeout. This option is identical to the **Close** option above, but indicates that the client connection timed out. For example:

```
19971118001632-0800:SMTP-Accept:Timeout:[10.2.85.88]:602:0:6
```

Receive. This option records the receipt of individual messages. Included in this log entry are the following: IP address of the connecting system (enclosed in [square brackets]), the unique identifier of the message, the size of the message (in bytes), the number of seconds required to accept the message, the sender's return address (enclosed in <angle brackets>), and a comma separated list of recipient addresses (each enclosed in <angle brackets>). The format of this log entry is:

```
<date-time>:SMTP-Accept:Received:[<client-ip>]:
      <message-id>:<bytes>:<seconds>:
      <sender>:<recipient>,<recipient>,...>
```

For example:

```
19970226001321-0800:SMTP-Accept:Received:[10.2.21.3]:
      19970226081320963.AAA133@fido.software.com:6703:0:
      <zack.taylor@megahuge.com>:<john.doe@software.com>,
      <jane.doe@software.com>
```

System. This option logs system failures that resulted in the inability to receive a message.

Alert. This option logs security-related warnings, such as when an SMTP client issues commands – such as `WIZ` or `DEBUG` – which are intended to compromise server security. This option also records the sending of an exceptional number of invalid SMTP commands, which can indicate an attempt to hack into the mail system. Included in this log entry are the IP address of the connecting system (enclosed in [square brackets]), and the potential security violation. The format of this log entry is:

```
<date-time>:SMTP-Accept:Alert:[<client-ip>]:
      <possible-security-risk>
```

For example:

```
19971118004203-0800:SMTP-Accept:Alert:[10.2.85.88]:
      Client issued ''WIZ''

19971118003812-0800:SMTP-Accept:Alert:[10.2.85.88]:
      Client issued too many bad commands
```


ConnectionRefused. This option logs network connections which are denied because the client's IP address matches a blocked IP address listed on the Mail Blocking Options Form. Included in this log entry is the IP address of the blocked system (enclosed in [square brackets]). The format of this log entry is:

```
<date-time>:SMTP-Accept:ConnectionRefused:[<client-ip>]
```

For example:

```
19970425164342-0700:SMTP-Accept:ConnectionRefused:[12.45.6.78]
```

SenderBlocked. This option logs the blocking of messages because the sender address matched a blacklisted pattern. Included with this log entry are the IP address of the client system (enclosed in [square brackets]), the sender address, and the number of failed recipients. The format of this log entry is:

```
<date-time>:SMTP-Accept:SenderBlocked:[<client-ip>]:
      <sender>:<#-recipients>
```

For example:

```
19970425164317-0700:SMTP-Accept:SenderBlocked:[10.3.91.11]:
      <incredible-offer@junkmailer.com>:1000
```

RelayDenied. This option logs attempted mail relaying that was denied because of settings in the SMTP Relay Restrictions Form. Included in this log entry are the IP address of the client system (enclosed in [square brackets]), the sender's return address, and the number of failed recipients. The format of this log entry is:

```
<date-time>:SMTP-Accept:RelayDenied:[<client-ip>]:
      <sender>:<#-recipients>
```

For example:

```
19971118005245-0800:SMTP-Accept:RelayDenied:[10.2.85.88]:
      <sir.spamalot@junkmailer.com>:30000
```

QueueRequest. This option records client usage of the ETRN or QSND commands, which requests processing of the mail queue. Included in this entry are the IP address of the client system (enclosed in [square brackets]), and the remote mail domain for which queue processing was requested. The format of this log entry is:

```
<date-time>:SMTP-Accept:QueueRequest:[<client-ip>]:<domain>
```

For example:

```
19971118005638-0800:SMTP-Accept:QueueRequest:[10.2.1.8]:software.com
```

Expand. This option records client usage of the `EXPN` command, which returns the primary address of an account given a valid address for that account. Included in this entry are the IP address of the client system (enclosed in [square brackets]), and the e-mail address specified with the request (enclosed in <angle brackets>). The format of this log entry is:

```
<date-time>:SMTP-Accept:Expand:[<client-ip>]:<address>
```

For example:

```
19971118011036-0800:SMTP-Accept:Expand:[10.2.1.8]:<jdoe@software.com>
```


Verify. This option records client usage of the VRFY command, which is used to verify the existence of an account given an e-mail address for the account. Included in this entry are the IP address of the client system (enclosed in [square brackets]), and the e-mail address specified with the request (enclosed in <angle brackets>). The format of this log entry is:

```
<date-time>:SMTP-Accept:Verify:[<client-ip>]:<address>
```

For example:

```
19971118011036-0800:SMTP-Accept:Verify:[10.2.5.8]:<jd@software.com>
```

WWW-Server

WWW-Server log entries record events that occur in the Post.Office web interface, such as users logging in to this interface and submitting forms. Included in these log entries is the IP address (enclosed in [square brackets]) of the client system which is accessing the web interface, as well as the type of transaction performed. The format of this log entry is:

```
<date-time>:WWW-Server:[<client-ip>]:<transaction>
```

For example:

```
19971118005122-0800:WWW-Server:[10.2.85.88]:GET / HTTP/1.0
      19971118005131-0800:WWW-Server:[10.2.85.88]:POST
      /Authentication HTTP/1.0
```

Account-Handler

This option records messages handled for a particular local account. Included in this log entry is the unique identifier of the message handled.

```
<date-time>:Account-Handler:<message-id>
```

For example:

```
19971118003322-0800:Account-Handler:
      19971118083321560.AAA176@zurich.Software.com
```

Account-Manager

This logging option records the use of the Account-Manager account to request and submit account-related e-mail forms. Included in this log entry is the unique identifier of the message handled.

AutoReply-Handler

This option records the use of the auto-reply feature, and creates a log entry each time an auto-reply message is sent. It includes the unique identifier of the received message that will be automatically replied to.

```
<date-time>:AutoReply-Handler:<message-id>
```

For example:

```
19971118013145-0800:AutoReply-Handler:
      19971118093141843.AAA220@fido
```

Configuration-Manager

This logging option records the use of the Configuration-Manager account to request and submit system-related e-mail forms. Included in this log entry is the unique identifier of the message handled.

Error-Handler

The Error-Handler records several events associated with errors in mail handling. The type of log entries created by this module include the sending of e-mail Message Action Forms to the Postmaster, receipt of these forms with Postmaster instructions, and the arrival of undeliverable messages. Note that the headers of error messages that are in error

The following examples indicate the types of Error-Handler logging. This first example indicates that the Error-Handler account received an e-mail form from the Postmaster:

```
19971118014014-0800:Error-Handler:19971118094012227.AAA177@fido
```

The following log entry is the result of invalid information provided in the message whose arrival was recorded in the above example. The reason for the error is given, which in this case is a missing or incorrect Postmaster password:

```
19971118014014-0800:Error-Handler:Error:
      Authentication Failed for message:
      (19971118094012227.AAA177@fido) Reason: Invalid Password.
```


The next example is a log entry that records the arrival of a message to Post.Office which could not be delivered for one reason or another. The reason for the error is given, as are the complete headers of the message are provided.

```
19971118015018-0800:Error-Handler:19971118095015865.AAA98@fido-
Unknown
      Function: Error-Handler
      Control-Type: Mail
      Priority: 2
      Submitted-Date: Tue, 18 Nov 1997 01:50:16 -0800
      MIME-Encoding: 7BIT
      Host-From: [10.2.85.88] [10.2.85.88]
      User-From: SMTP<scottm@sparky.software.com>
      Message-Size: 1127
      MTA-Hops: 0
      Channel-To: SMTP <john.deo@software.com>
      Error: SMTP-Router:UnknownAccounts (WriteUnknownAcct)
      Error-Text: SMTP <john.deo@software.com>
      Trace: SMTP-Accept
      Trace: SMTP-Router
```

Note: *As shown in the above example, headers and other information about undeliverable messages are indented with a tab character. In fact, these are the only log entries that do not begin with a date/time stamp. This allows an automatic log parser to easily detect this information.*

List-Exploder

This log entry indicates that Post.Office has extracted the list of individual recipients for a message addressed to a mailing list. Included in this log entry is the unique identifier of the message that was worked on by the List-Exploder (that is, the message that was addressed to a mailing list). For example:

```
19971118173823-0800:List-Exploder:19971119013820651.AAA64@sparky
```

Note: *Another event logged by the List-Exploder module is the distribution of messages to all subscribers using the immediate mode of delivery; this log entry is controlled separately by the Immediate Delivery logging option described below.*

List-Manager

The option records information about messages which are addressed to the List.Manager account, or to the request handler accounts associated with mailing lists. These are typically messages which contain list manager e-mail commands, or are bounce messages returned by other mail servers. Included in this log entry is the unique identifier of the message that was received. For example:

```
19971118182616-0800:List-Manager:19971119022609687.AAA141@sparky
```


Mailbox-Deliver

This option records the delivery of a message to POP3 mailboxes. Included in this log entry is the unique identifier of the message that was delivered, and a comma-separated list of recipient mailboxes. The format of this log entry is:

```
<date-time>:Mailbox-Deliver:<message-id>:<mailbox>, ...
```

For example:

```
19971118173824-0800:Mailbox-Deliver:
      19971119013823645.AAA220@spoarky.com:Jane_Doe,John_Doe
```

Note: *Recall from Section 8.3.1 that mailbox names are taken from each account's unique identifier (UID).*

Program-Deliver

Similar to the Mailbox-Deliver option above, this option records the delivery of a message to a program. Included in this log entry is the unique identifier of the message that was delivered, and a comma-separated list of recipients (specified by account UID). The format of this log entry is:

```
<date-time>:Program-Deliver:<message-id>:<account-id>, ...
```

For example:

```
19971118173849-0800:Program-Deliver:19971119013823645.AAA220@z.com:
      Jane_Doe,John_Doe
```

Unix-Deliver (UNIX platforms only)

On Unix platforms, the Unix-Deliver logging options records the delivery of messages to the Unix mail facility. Included in this log entry is the unique identifier of the message that was delivered, and a comma-separated list of recipients (specified by account UID). The format of this log entry is:

```
<date-time>:Unix-Deliver:<message-id>:<account-id>, ...
```

For example:

```
19971118173849-0800:Unix-Deliver:19971119013823645.AAA220@baz.com:
      Jane_Doe,John_Doe
```


SMTP-Deliver

This option logs activities of the SMTP-Deliver module, which is responsible for sending outgoing messages to other mail servers.

The most common SMTP-Deliver log entry is one that records the attempted delivery of a message. Included in this log entry are the unique identifier of the message, the action taken for the message (Delivered or Deferred), the size of the message (in bytes), the host name of the destination mail server, the return address of the message, and a comma-separated list of recipient addresses (each enclosed in <angle brackets>). The format of this log entry is:

```
<date-time>:SMTP-Deliver:<message-id>:
      <action>:<bytes>:<hostname>:<sender>:
      <recipient>,<recipient>,...>
```

For example:

```
19970716011333-0700:SMTP-Deliver:19970716081331455.AAA111@fido:
      Delivered:2019:mail.accordance.com:<skippy@software.com>:
      <joe@accordance.com>

19970716012012-0700:SMTP-Deliver:19970715005131664.AAA513@sparky:
      Deferred:616:maczieg.com:<jdoe@software.com>:
      <scott@maczieg.com>, <chris@maczieg.com>
```

Other SMTP-Deliver log entries are warnings that indicate that connections to other servers timed out, or were unsuccessful because of problem with the domain's DNS records. For example:

```
19970716011847-0700:SMTP-Deliver:Warning:
      MX lookup for foo.com timed out

19970716011853-0700:SMTP-Deliver:Warning:
      MX lookup for bar.com returned no records

19970716131627-0700:SMTP-Deliver:Warning:
      Timed out waiting for SMTP greeting from: 207.177.177.11
```

SMTP-Router

This logging option records the activities of the SMTP-Router, the module responsible for reading the headers of incoming messages and determining how they should be handled (delivered to a mailbox, sent to a remote mail server, etc.). Included in this log entry is the unique identifier of the message that was handled.

```
19951020130722-0800:SMTP-Router:19951020200722.AAA1234@foo.com
```


List Creation

This option controls logging for a specific WWW-Server event: the creation of a new mailing list. This log entry includes the List Name of the new mailing list, and is in the format:

```
<date-time>:WWW-Server:List-Created:<listname>
```

For example:

```
19970306183857-0800:WWW-Server:List-Created:surfing
```

List Deletion

Like the List Creation option above, this option controls logging for a specific WWW-Server event: in this case, the deletion of an existing mailing list. This log entry includes the List Name of the deleted mailing list, and is in the format:

```
<date-time>:WWW-Server:List-Deleted:<listname>
```

For example:

```
19970306183836-0800:WWW-Server:List-Deleted:elvis_fans
```

Subscriptions

This option controls logging for user subscriptions to mailing lists. This may appear as an event logged from two different Post.Office modules: the List-Manager (subscription requests submitted via e-mail), and the WWW-Server (subscriptions via the web interface). Both types of entries include the List Name of the mailing list, the address of the subscriber (enclosed in <angle brackets>), and the delivery mode (digest or immediate) requested.

The format of these entries is as follows:

```
<date-time>:List-Manager:User-Subscribed:<listname>:
      <subscriber>:<mode>

<date-time>:WWW-Server:User-Subscribed:<listname>:
      <subscriber>:<mode>
```

For example:

```
19970306184040-0800:List-Manager:User-Subscribed:surfing:
      <john.doe@software.com>:digest

19970710124120-0700:WWW-Server:User-Subscribed:archery:
      <jane.doe@software.com>:immediate
```


Unsubscriptions

This option is similar to the Subscriptions option above, and logs user unsubscriptions from mailing lists. This may appear as an event logged from two different Post.Office modules: the List-Manager (unsubscription requests submitted via e-mail), and the WWW-Server (unsubscriptions via the web interface). Both types of entries include the List Name of the mailing list, the address of the subscriber (enclosed in <angle brackets>), and the delivery mode (digest or immediate) that the user was subscribed in.

The formats of these entries is as follows:

```
<date-time>:List-Manager:User-Subscribed:<listname>:
      <subscriber>:<mode>

<date-time>:WWW-Server:User-Subscribed:<listname>:
      <subscriber>:<mode>
```

For example:

```
19970306184040-0800:List-Manager:User-Unubscribed:surfing:
      <john.doe@software.com>:digest

19970710124120-0700:WWW-Server:User-Unsubscribed:archery:
      <jane.doe@software.com>:immediate
```

List Statistics

This entry indicates the distribution of a mailing list's nightly statistics message to the list owner(s). It includes the List Name of the mailing list, the number of messages submitted to the mailing list that day, the total size (in bytes) of those messages, the number of subscribers, and the mailing list's digest schedule. The format of this log entry is:

```
<date-time>:List-Scheduler:List-Statistics:<listname>:
      <#-messages>:<bytes>:<#-subscribers>:<digest-schedule>
```

For example:

```
19970307000046-0800:List-Scheduler:List-Statistics:league:
      13:37:12:daily 5 pm
```

Digest Delivery

This option records the distribution of a mailing list's digest, which is handled by the List-Scheduler module. Included in this entry is the List Name of the mailing list, the unique identifier of the message (enclosed in <angle brackets>), and the size (in bytes) of the digest message. When the **Standard** mode is selected, the number of subscribers is also shown, whereas the **Verbose** mode includes a complete list of message recipients (each enclosed in <angle brackets>) instead of only the number.

The format the **Standard** log entry is:

```
<date-time>:List-Scheduler:List-Activity-Digest:<listname>:
      <message-id>:<bytes>:<#-recipients>
```

For example:

```
19970306170300-0800:List-Scheduler:List-Activity-Digest:surfing:
      <19970307010258708.AAA95@software.com>:24:39
```

The format the **Verbose** log entry is:

```
<date-time>:List-Scheduler:List-Activity-Digest-Verbose:
      <listname>:<message-id>:<bytes>:
      <recipient>,<recipient>,...>
```

For example:

```
19970306170300-0800:List-Scheduler:List-Activity-Digest-Verbose:
      surfing:<19970307010258708.AAA95@software.com>:24:
      <joe.schmoe@software.com>,<john.doe@software.com>
```

Immediate Delivery

This option is similar to the Digest Delivery option above, and records the distribution of a message to all mailing list subscribers using the immediate mode of delivery. Unlike the Digest Delivery option, this log entry is recorded by the List-Exploder module, but is otherwise nearly identical.

Included in this entry is the List Name of the mailing list, the unique identifier of the message (enclosed in <angle brackets>), and the size (in bytes) of the message. When the **Standard** mode is selected, the number of subscribers is also shown, whereas the **Verbose** mode includes a complete list of message recipients (each enclosed in <angle brackets>) instead of only the number.

The format the **Standard** log entry is:

```
<date-time>:List-Exploder:List-Activity-Immediate:<listname>:
      <message-id>:<bytes>:<#-recipients>
```

For example:

```
19970306183651-0800:List-Exploder:List-Activity-Immediate:surfing:
      <19970307024224950.AAA364@zurich>:2:39
```

The format the **Verbose** log entry is:

```
<date-time>:List-Exploder:List-Activity-Immediate-Verbose:
      <listname>:<message-id>:<bytes>:<recipient>,<recipient>,...>
```

For example:

```
19970306183651-0800:List-Exploder:List-Activity-Immediate-Verbose:
      surfing:<19970307024224950.AAA364@zurich>:2:
      <joe.schmoe@software.com>,<john.doe@software.com>,...
```


8.4.4 Cleaning Out Log Files

A new Post.Office log file is created every day at midnight, with yesterday's log file left intact for your records. The size of each log file varies depending on the number of logging options you have selected and the amount of mail traffic on your system. Over time, maintaining an excessive number of log files can noticeably impact your server storage space. It is therefore recommended that you go to the log directory periodically and delete log files that you consider too old to be useful.

9

Backup and Restore Instructions

This chapter is intended to assist you in backing up and restoring the files that make up Post.Office mail system. The topics discussed in this chapter include:

- Backing up and restoring Post.Office on NT platforms with the help of the poperms utility
- Backing up and restoring Post.Office on UNIX platforms

9.1 Backing Up the Mail System

How many times have you been told to back up your system? Well, you're about to be told again. The importance of this safety precaution cannot be overemphasized. True, Post.Office runs flawlessly most of the time, but that occasional hardware failure still needs to be considered.

The frequency with which you back up your system is determined by the specific characteristics of your installation (the amount of mail processed per day, the potential importance of a single message, the time required to execute a backup, etc.). You should review the standard practices established at your site for guidance in determining how often to back up your mail server.

When backing up the Post.Office files we advise that you do the following:

- Observe our naming conventions for ease of reference. This is not required, but it will make your life easier should you ever need to use the restoration instructions. (Those instructions assume use of our recommended names.)
- Store all backup files on another host.
- Store all files relating to a single backup in the same directory.

9.1.1 The Post.Office Permission Setting Tool (poperms)

Poperms is a utility which allows you to modify NT permissions in order to facilitate the backup, restoration, and relocation of Post.Office. It is included with the Post.Office package for version 3.5. You will find the poperms tool referenced in multiple instruction sets. Each time the tool is referenced the proper parameters will be specified.

The program runs from the command line using the following syntax:

POPERMS [-r] [-p] [-o] [-l] [-m] [-f] [-a] [-e] [-u | user_name]

The poperms variables are defined in the table below.

Variable	Meaning
`-r`	Set registry permissions
`-p`	Set Post.Office Program directory permissions
`-o`	Set Post.Office Spool directory permissions
`-l`	Set Post.Office Log directory permissions
`-m`	Set Post.Office Mailbox directory permissions
`-f`	Set all Post.Office File system directory permissions
`-a`	Set ALL Post.Office File AND Registry permissions
`-e`	give full permissions to the group Everyone in addition to the user specified
`-u`	use the Post.Office user that the service logs in as
`user_name`	name of user to give permissions to

To execute a command successfully you must supply at least one parameter from the first group, and *only* one from the second group. The program will then set the requested files and Registry entries to Full Control for the user specified (and *additionally* allow Full Control to the group Everyone if that option is selected).

Note: *The most common use of this tool in correcting permission problems is to reset the permissions on all Post.Office files and Registry entries to the proper permissions for the Post.Office user (the one identified at time of installation). The command that follows accomplishes that goal.*

```
poperms  -a  -u
```

Security Feature: You must be a member of the administrator group to run this program, and the utility must be placed in the Post.Office executables directory. The utility is placed in that directory by default when Post.Office 3.5 is installed.

9.1.2 Post.Office Full System Backup for NT

Safe software practices dictate that you back up your mail server on a regular basis. Follow the steps outlined below to back up your entire Post.Office installation (executables, account and configuration information, and mailboxes and their contents).

Note: *Restoration instructions (should you need them) are provided later in this chapter, but they assume proper completion of the documented backup procedure.*

1. Make sure that you are logged in as the Local Administrator of your host. Logging in as the Domain Administrator is not sufficient unless the machine you are using is a Primary Domain Controller.
2. Stop the Post.Office service using the Control Panel's Services applet (select the item named post.office-MTA and click Stop).
3. Open the permissions to "everyone" by using the following command:

   ```
   poperms -a -e -u
   ```

4. Launch the Registry Editor (REGEDT32.EXE) and make backup copies of the required Registry keys.
   - Locate the `HKEY_LOCAL_MACHINE\SOFTWARE\Software.com\Post.Office` key, select it, pull down the Registry menu, and choose the Save Key command. Name the backup file `Post.Officev3.5RegKeyBackup`.
   - Locate the following key: `HKEY_LOCAL_MACHINE\SOFTWARE\Software.com\Post.Office\MTA-Accounts` select it, pull down the Registry menu, and choose the Save Key command. Name the backup file `MTAv3.5RegKeyBackup`.
5. Back up the required file system information. (The exact location of the required directories can be found via the Post.Office Control Panel applet. When the Post.Office applet window appears, look at the entries at the bottom of the window.)
   - Save the Spooling Directory containing configuration and account information under the name `Post.Officev3.5configBackup`.
   - Save the Program Directory containing Post.Office executables under the name `Post.Officev3.5execsBackup`.
   - Save the Mailbox Directory containing your mailbox information under the name `Post.Officev3.5mailboxBackup`.
6. Close the permissions back to the installation defaults by using poperms:

   ```
   poperms -a -u
   ```

7. Start Post.Office Stop the Post.Office service using the Control Panel's Services applet (select the item named post.office-MTA and click Start).

9.1.3 Post.Office Full System Backup for UNIX

Safe software practices dictate that you back up your mail server on a regular basis. Follow the steps outlined below to back up your entire Post.Office installation. Restoration instructions (should you need them) are provided later in this chapter, but they assume proper completion of the documented backup procedure.

Solaris Backup Instructions:

These instructions assume that you have installed Post.Office in the default locations which are /opt/post.office (program executables) and /var/spool/post.office (the spooling directories). If you selected other locations you will need to adjust the instructions accordingly. To determine the current locations of your Post.Office, Program, and Mailbox directories list the contents of the `/etc/post.office.conf` file.

1. Log on as root.
2. Shutdown Post.Office by typing:

   ```
   /opt/post.office/post.office  shutdown
   ```

3. Back up the following items making sure to maintain permissions and links:
   - /var/spool/post.office (the Post.Office account and configuration information)
   - /var/spool/mailbox (the mailbox information)
   - /etc/post.office.conf (your configuration file)

Backup Instructions for All Other UNIX Platforms

These instructions assume that you installed Post.Office in the default locations which are /usr/local/post.office (program executables) and /var/spool/post.office (the spooling directories). If you selected other locations you will need to adjust the instructions accordingly. To determine the current locations of your Post.Office, Program, and Mailbox directories list the contents of the /etc/post.office.conf file.

1. Log on as root.
2. Shutdown Post.Office by typing:

   ```
   /usr/local/post.office/post.office  shutdown
   ```

3. Back up the following items making sure to maintain permissions and links:
   - /usr/local/post.office (the Post.Office executables)
   - /var/spool/post.office (the Post.Office account and configuration information)
   - /var/spool/mailbox (the mailbox information)
   - /etc/post.office.conf (your configuration file)

9.2 Restoring the Mail System

Restoring your mail server is a relatively painless process, provided you took the precaution of making regular backups. The instructions that follow assume the existence of the required backup files.

9.2.1 Restoring the Mail System on Windows NT

If you backed up your Post.Office system as instructed (see Section 9.1.2), you should have copies of the following items.

Registry Information

- a backup copy of the Post.Office Registry key (the `Post.Officev3.5RegKeyBackup` file)
- a backup copy of the MTA-Accounts Registry key (the `MTAv3.5RegKeyBackup` file)

File System Information

- a backup copy of the Spooling Directory (`Post.Officev3.5configBackup`)
- a backup copy of the Program Directory (`Post.Officev3.5execsBackup`)
- a backup copy of the Mailbox Directory (`Post.Officev3.5mailboxBackup`)

The instructions that follow will guide you through the restoration process. They assume that you named the backup files as suggested. If you selected other names for those files, you will need to adjust the instructions accordingly.

1. Make sure that you are logged in as the Local Administrator of your host. Logging in as the Domain Administrator is not sufficient unless the machine you are using is a Primary Domain Controller.
2. Once you have re-installed NT, you must re-install Post.Office on the machine by:
   - Running Setup.exe
   - Accessing Post.Office as “Postmaster” to complete the Installation Wrap-up Form
3. After Post.Office has been re-installed, stop the Post.Office service using the Control Panel’s Services applet (select the item named post.office-MTA and click Stop).
4. Launch the Registry Editor (`REGEDT32.EXE`) and restore the backup copy of the Post.Office Registry key over the current copy.
   - Locate the `HKEY_LOCAL_MACHINE\SOFTWARE\Software.com\Post.Office` key, select it, pull down the Registry menu, and choose the Restore command. Specify `Post.Officev3.5RegKeyBackup` as the key from which to restore.

5. Restore the required file system information. (The exact location of the current directories can be found via the Post.Office Control Panel applet. When the Post.Office applet window appears, look at the entries at the bottom of the window.)

   - Restore the backup copy of the Spooling Directory (`Post.Officev3.5configBackup`) by copying it on top of the current copy. This will restore your configuration and account information.
   - Restore the backup copy of the Mailbox Directory (`Post.Officev3.5mailboxBackup`) by copying on top of the current copy. This will restore your all mailbox information.

6. Run poperms to reset the permissions back to the default installation permissions:

   ```
   poperms  -a  -u
   ```

7. Restart Post.Office Stop the Post.Office service using the Control Panel's Services applet (select the item named post.office-MTA and click Start).

9.2.2 Restoring the Mail System on UNIX

The restoration instructions guide you in replacing the current copies of the Program, Spooling, and Mailbox directories with the versions you stored after your last backup. To determine the current locations of your Program, Spooling, and Mailbox directories list the contents of the /etc/post.office.conf file.

Restoration Instructions for Solaris

If you backed up your Post.Office system as instructed (see Section 9.1.3), you should have copies of the following items:

- your Post.Office account and configuration information
- your mailbox information
- your configuration file

Assuming you took that precaution, please:

1. If Post.Office is running , shut it down by typing:

   ```
   </opt>/Post.Office  shutdown
   ```

2. Restore the two file system directories (containing the Post.Office account and configuration information, and your mailbox information), as well as the configuration file from your backup, on top of the current versions. Remember to maintain permissions and links.

   Note: If you run into ownership or permission problems, it's possible that executing a chown or `chgrp` on the copied files and assigning ownership to your Post.Office user (mta) and group (mta) will be enough. The required permissions for UNIX are listed in the Post.Office FAQ for your review (http://wwww.software.com).

3. Remove the 3.5 executables with pkgrm:

   ```
   pkgrm  SCOM-MTA
   ```

4. Download the 3.5 version of Post.Office for Solaris from our web site to a temporary directory.

5. Uncompress the file you downloaded and expand the resulting archive file to create the Post.Office package (SCOM-MTA) by typing:

   ```
   cd /var/tmp/PO35
   uncompress
   tar  xvpf  packagename.tar
   ```

6. From the location in which you stored it, install the 3.5 package by typing:

   ```
   pkgadd -d . SCOM-MTA
   ```

   Caution: The upgrade program will ask if you wish to change configuration information. You should answer “no”.

7. If you are currently using Program Delivery you will need to re-enable that feature by typing:

   ```
   chmod u+s /opt/post.office/local/Program-Deliver
   rm /opt/post.office/trusted/NO-PROGRAM-DELIVERIES
   ```

   Caution: There are security issues associated with the use of Program Delivery. Please read Chapter 6 of the Post.Office manual to ensure you understand those issues before enabling this feature.

8. Run the Post.Office configuration program: /opt/post.office/Setup.

Restoration Instructions for All Other UNIX Platforms

If you backed up your Post.Office system as instructed (see Section 9.1.3), you should have copies of the following items:

- the Post.Office executables
- your Post.Office account and configuration information
- your mailbox information
- your configuration file

Assuming you took that precaution, please:

1. If Post.Office is running , shut it down by typing:

   ```
   </usr>/local/Post.Office  shutdown
   ```

2. Restore the three file system directories (containing the Post.Office executables, the Post.Office account and configuration information, and your mailbox information), as well as the configuration file from your backup, on top of the current versions. Remember to maintain permissions and links.

3. Start the Post.Office server:

   ```
   </usr>/local/Post.Office  startup
   ```


Hint: *You might run into permission problems if you didn't backup or restore with the original permissions. The required permissions for UNIX are listed in the Post.Office FAQ for your review (http://www.software.com). Once your mail server is restored and the permissions properly set, you will need to re-start the Post.Office server.*

10

Troubleshooting

The best time to review recommended troubleshooting techniques is before you have a problem. With that in mind we've developed the following sections to provide you with the background information necessary to handle exceptional situations with confidence. We've even included an overview of our favorite troubleshooting tools.

10.1 The Post.Office FAQ

For a more extensive list of questions and answers check the Post.Office FAQ. The FAQ is an open document with answers to frequently asked questions about Post.Office and Software.com. It's available on our web site at `http://www.software.com`. We continuously update and revise the FAQ to reflect new questions from our customers and discuss new features available in Post.Office.

You can obtain the FAQ via the web or ftp.

- To view or retrieve the FAQ using the World Wide Web, simply point your browser at `http://www.software.com`. From the main menu, follow the appropriate links to the FAQ. You can either read it on your browser or download the file.
- To obtain the FAQ via ftp, log in to the ftp server, `ftp.software.com`, as user *ftp* or *anonymous*. Provide your e-mail address as the password.

We encourage you to review the FAQ every once in a while. That way you'll be able to find answers to your questions before they come up, and gain insight as to how other folks put Post.Office to good use.

10.2 How Mail is Routed through Post.Office

The first step in troubleshooting your mail server is understanding what's happening behind the scenes.

Post.Office is always listening for incoming mail on the standard port for SMTP transactions (port 25). This incoming mail can be from:

- a local (or remote) mail client software sending mail out
- another mail server on the local network
- or, a remote mail server out on the Internet.

Once mail is received by Post.Office it must determine to whom it should be delivered. To make that decision Post.Office relies on the addressing information that appears on the electronic envelope "containing" the message. The addresses on the envelope are referred to by different names than those on the message header and may, in fact, contain different information. The "Mail From:" and "Rcpt To:" addresses on the envelope are analogous to the "To:" and "From:" addresses on the message header, but it's important to note that Post.Office relies on the information provided in the former and not the latter.

10.2.1 Standard Flow of Mail Through the Server

The steps below provide a summary of the delivery process. Each step is explained in detail in the sections that follow. The concepts are illustrated in Figures 10-1 and 10-2.

1. Check the sender address against the Mail Blocking rules; reject the message if needed.
2. Check for SMTP address compliance; perform address completion as required.
3. Check for Incoming Domain Re-writing rules for the destination domain; perform domain re-writing as required.
4. Check the SMTP Relay Restrictions to see if the sender is allowed to send mail to the recipient; reject the message if needed.
5. Check to see if From: Address Re-writing should be performed.
6. Check the channel aliases for instructions on external re-routing.
7. Check for delivery to a local address (mailing list or mail account).
8. Check local mail domains to determine if message falls within the scope of this mail server's authority.
9. Check the Mail Routing Table for additional re-routing instructions.
10. Check domain name server definitions (i.e. MX and A records) or the local host file.

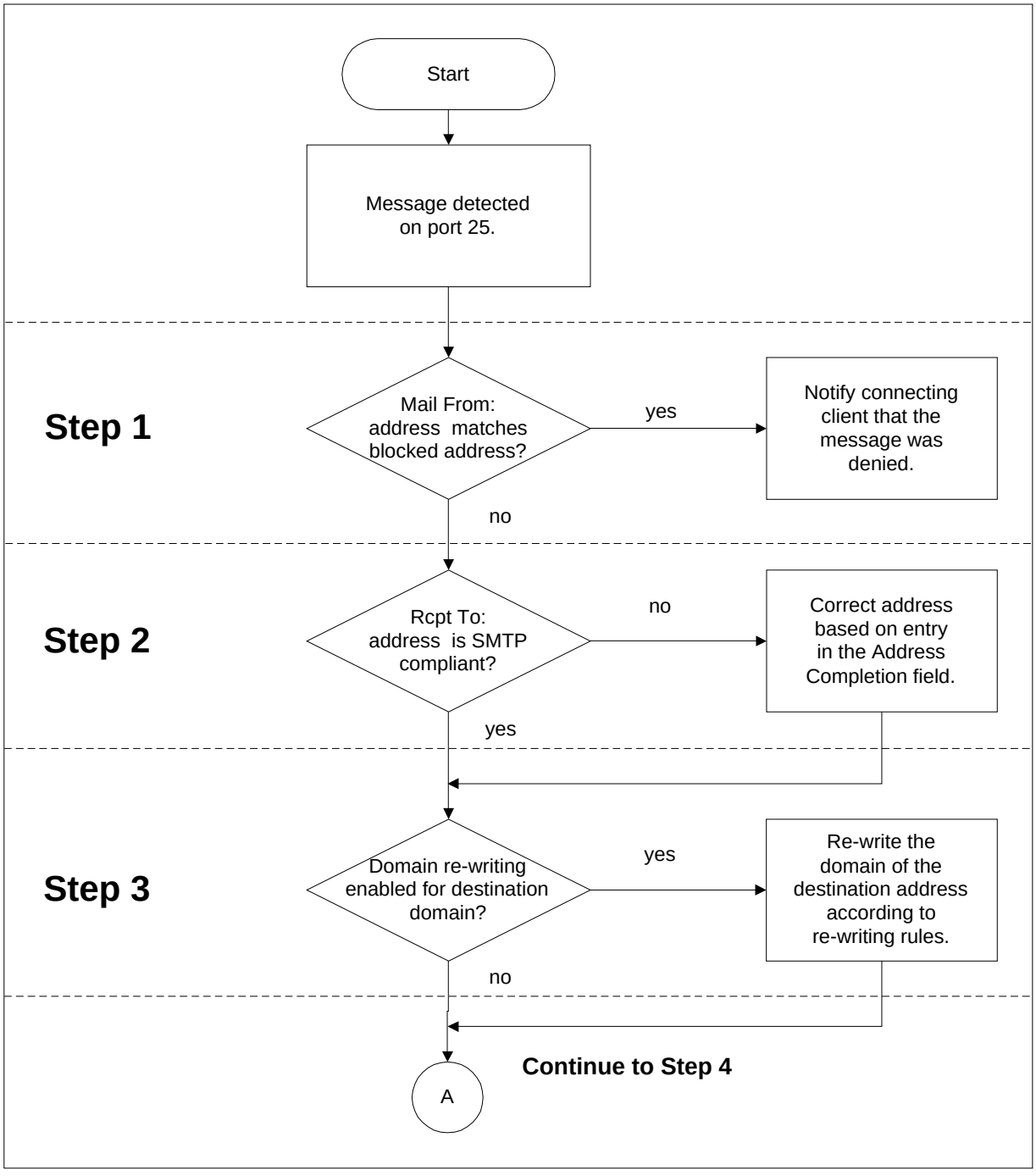

Figure 10-1: Initial operations performed on all mail received by the Post.Office server

Step 1: Check the sender address against the Mail Blocking rules; reject the message if needed

If you are using the Mail Blocking features, Post.Office will check the sender address of each incoming message against the list of addresses, domains, and usernames specified on the Mail Blocking Form. If the "Mail From:" address is covered under any of these blocking rules, the sending client is notified that the message will not be accepted. Note that Post.Office does this immediately after it receives the sender address in the "Mail From:" SMTP command; it will not even read in the headers and body of rejected messages.

Note: *If you are blocking mail by IP addresses, and a blacklisted system attempts to give mail to your server, Post.Office will drop the connection before any mail information is sent. In other words, such systems never even get as far as Step 1 here; the connection is dropped as soon as it is made.*

Step 2: Check for SMTP address compliance; perform address completion as required

In order to deliver mail, Post.Office requires standard, fully qualified SMTP addresses in both the "Mail From:" and "Rcpt To:" fields on the message envelope. If the original delivery address conforms to SMTP standards (i.e., it is in the format xxx@yyy.zzz or some other variation that includes an @ sign and at least one dot to the right of the symbol) the server proceeds immediately to Step 3. However, if Post.Office receives a message envelope from a mail client or server that does not have a standard SMTP address (i.e. joe), it will attempt to make the address compliant. The logic for address completion is as follows:

- If the original address does not include an @ sign, assume it represents a user name (without host or domain). Check the configuration database for an entry in the Address Completion field. If an address completion entry exists, add an @ sign to the original address, follow it with the address completion entry, and proceed to Step 3. If the Address Completion field is blank, look up the host.domain name of the machine running Post.Office, add an @ sign to the original address, follow it with the host.domain name, and proceed to Step 3.
- If the original address includes an @ sign, but no dots to the right of that symbol, assume it represents a user/host combination. Find the domain name specified at the time of installation, append the domain name to the right of the host name and proceed to Step 3.

Step 3: Check for Incoming Domain Rewriting rules for the destination domain; perform domain re-writing as required

Post.Office next checks the domain portion of each “Rcpt To:” address to determine if it should be rewritten. If an entry exists for a domain in the Incoming Domain Rewriting table, then all destination addresses that include the domain will be automatically rewritten to the new domain value before moving on to Step 4.

For example, if you have defined an Incoming Domain Rewriting rule to rewrite the domain accordance.com to software.com, then all messages sent to

```
john.doe@accordance.com
```

will have their envelope destination address rewritten in this step to

```
john.doe@software.com
```

If domain rewriting is not required, Post.Office simply continues on to Step 4 without making any changes to the message.

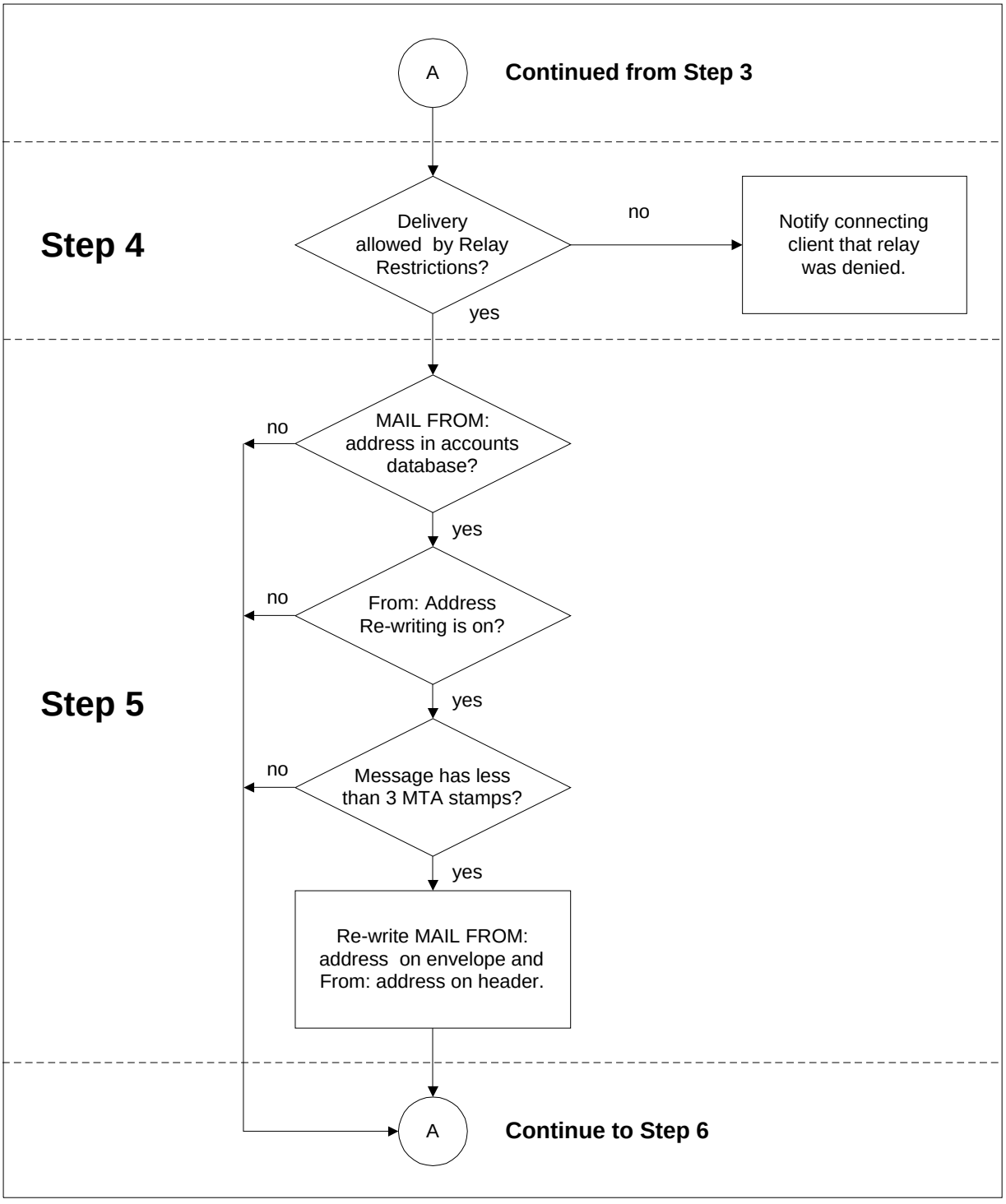

Figure 10-2: The various steps in determining whether a return address will be re-written.

Step 4: Check the SMTP Relay Restrictions to see if the sender is allowed to send mail to the recipient; reject the message if needed

Next, Post.Office checks its SMTP Relay Restrictions to see if relay is being restricted. If no relay restrictions exist, the message continues on to Step 5.

If relay is restricted, Post.Office checks the IP address of the connecting client, as well as the "Mail From:" address, against the values specified in the SMTP Relay Restrictions Form. If these rules do not restrict mail from this sender, then the message proceeds to Step 5; otherwise, it is considered restricted.

If the message is found to be restricted, Post.Office then checks the Relay Restrictions delivery rules, which define the domains which are allowed to receive restricted relay mail. If delivery to a recipient is allowed by these rules (for instance, if it is addressed to a local user), then the message continues on to Step 5. If delivery is not allowed, Post.Office notifies the connecting client that the relay attempt was rejected.

Step 5: Determine if From-Address Rewriting Should be Performed

Post.Office will check to determine if the message header's From: address should be re-written. It will only re-write the address if Post.Office is the first or second mail server to have received the message. Otherwise, we consider ourselves too "far" from the user to perform From Address Rewriting. (Post.Office determines how far a message has traveled by counting the "received by" lines in the envelope header.)

To determine if the "Rcpt To:" address should be re-written, Post.Office looks for an exact match between the "From" address on the envelope and any of the Internet addresses listed in the accounts database. If no match is found (or the message is from too far away), the system proceeds to Step 6. If an exact match is found between the "Rcpt To:" address on the envelope and an entry in the Internet Addresses field of any locally defined account, the From Address Rewrite Style for that account is noted. If the entry in the From Address Rewrite Style field is **none**, the address is left as originally written. If a From Address Rewrite style is indicated, the server retrieves the Primary Internet Address for the account, formats the address in the style indicated, and uses the re-formatted address to replace the original "Rcpt To:" address on the envelope, and the "From:" address on the message header.

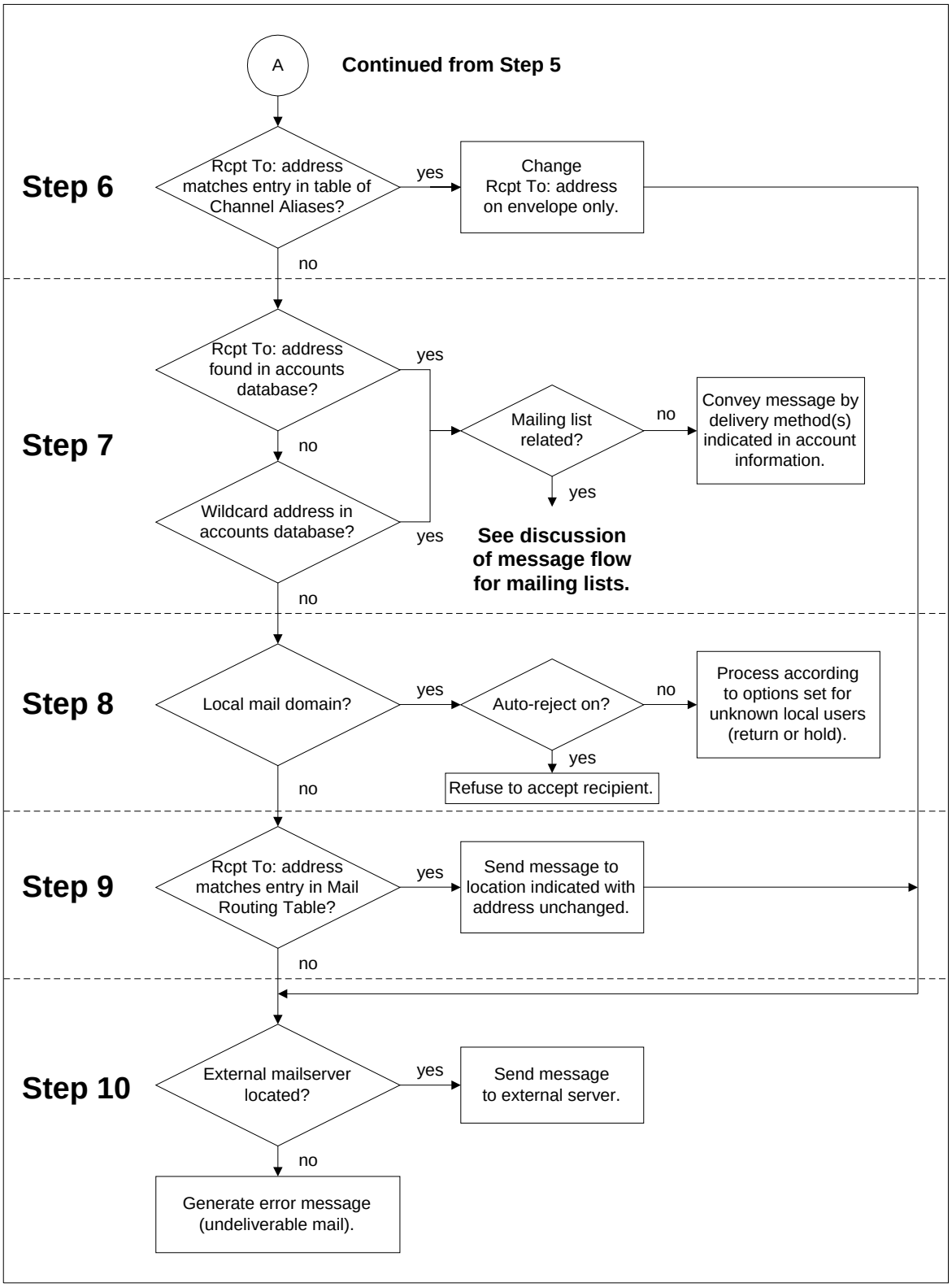

Figure 10-3: The operations performed by Post.Office when attempting to deliver a message (continued from Figure 10-1)

Step 6: Check SMTP Channel Alias Table

Next Post.Office checks the "Rcpt To:" address on the envelope against the entries in the SMTP Channel Alias Table to determine if the message should be immediately re-routed to another account address on another mail server host.

If the "Rcpt To:" address on the envelope matches an Internet address listed in the left section of an alias entry, Post.Office replaces that address with the address in the right section and skips ahead to Step 10.

If no match is found, Post.Office patiently proceeds to Step 7.

Step 7: Check for Local Delivery

If the "Rcpt To:" address on the envelope is not found in the SMTP Channel Alias Table, Post.Office will try to determine if the mail should be delivered locally. Local delivery is determined by matching the "Rcpt To:" address on the envelope with an Internet address in the Post.Office accounts database. If an exact match is found among the entries in the Internet Addresses field of any account, Post.Office will deliver the message to that account as specified (i.e. POP3 Delivery, Forwarding, Local Deliver, and/or Program Delivery for UNIX Systems). If a match is found among the Internet addresses for a local mailing list, the message will be sent on to the appropriate list management module for handling. (See the next section of this chapter for a more complete discussion of list mail handling.)

If the address on the envelope does not match the address for any local mail account or local mailing list, the system proceeds to Step 8.

Step 8: Check Local Mail Domain

If the "Rcpt To:" address on the envelope is not found in a Post.Office account definition, Post.Office will check to see if it has the authority to say that it is an "unknown user or account for this domain." It does this by comparing the domain in the "Rcpt To:" address to the list of local mail domains maintained in the configuration database (entries made via the Local Mail Domain field in the Post.Office System Configuration Form). If a match is found, it indicates that Post.Office is the sole mail server for this domain and any addresses not found locally are considered unknown. An error message will be generated saying that the message recipient is an unknown account. This message will be sent by Post.Office to the Postmaster (always) and to the originator (if defined to do so in the Error Response Parameters web form).

Note: *The system automatically assumes that the host.domain name of the machine running Post.Office is included among the local mail domains.*

Step 9: Check Mail Routing Table

If the domain in the "Rcpt To:" address on the envelope is not found among the entries in the Local Mail Domain field (there may be more than one), Post.Office will assume another mail server is responsible for this domain.

To determine if mail for that domain should be routed to another specific mail server host, Post.Office checks the entries in its Mail Routing Table (MRT). If the domain in the envelope delivery address matches the domain listed to the left of the colon in any MRT entry, Post.Office re-directs the message to the machine identified on the right side of the colon.

Note: *The entries in the Mail Routing Table are order sensitive. If the "Rcpt To:" address matches more than one entry, it will be processed according to the instructions set by the first matching entry. Remember this if you want to send all mail that isn't delivered locally to a firewall mail server; you would want that to be your last entry.*

For example, to route mail addressed to the domain `msmail.com` to the SMTP gateway for that domain, the required entry would be:

```
msmail.com:[tcp/ip_address_of_msmail_gateway]
```

And to ensure that mail to all hosts and subdomains was similarly routed, this additional entry should be made:

```
*.msmail.com:[tcp/ip_address_of_msmail_gateway]
```

To route all mail to a firewall mail server, entries like the ones below are required:

```
*:[tcp/ip_address_of_firewall]
*.*:[tcp/ip_address_of_firewall]
```

If the entry to the right of the colon specifies a TCP/IP address, Post.Office will deliver the mail to that host without assistance. If the route is defined as a host or domain name, Post.Office will need to proceed to Step 10 locate the IP address for that external host.

Similarly, if no match is found, Post.Office proceeds to Step 10.

Note: *Unlike the processing for channel aliases, handling via the Mail Routing Table does not result in the re-writing of envelope address information. The mail server to which the message is routed must be able to understand the address as is.*

Step 10: Determine Location of External Host

If the domain name in the message envelope's "Rcpt To:" address is not listed in the Mail Routing Table, Post.Office assumes it should deliver the mail message externally to the mail server defined for that domain in the Domain Name System (DNS).

Post.Office will ask its host machine to find the TCP/IP address of the mail server for the specified domain. The host will first ask for the Mail Exchange Record (MX record) defined for that domain and, if found, will return the TCP/IP address of that mail server. If no MX record is found, the host will ask for the Address Record (A record) for that domain and, if found, will return the TCP/IP address of that host. If no MX or A record are found for the domain, the host will check its local host file to see the domain name is listed there. Assuming one of these requests comes back with a TCP/IP address, Post.Office will deliver the message to that address.

If no TCP/IP address can be found for that domain, the message will be considered undeliverable. An error message will be generated within Post.Office and sent to the originator of the message, unless the return address is bad, in which case the Postmaster will be notified.

Exceptions

Two new mail server options introduced in Post.Office 3.5 can affect the routing of mail within your system. The use of either requires an understanding of the concept of local mail domains.

Local mail domains are identified via entries on the Mail Routing Options web form. They are mail domains over which you are claiming complete authority. As a result, you are allowed to decide the fate of all messages sent to any address within that domain.

Automatic refusal of mail for unknown local users, causes a change in Step 8. If the option to "Verify recipients within Local Mail Domains before accepting mail?" is turned on, the server checks for the existence of a local mail account *before* accepting delivery of a message. If no local account is found, delivery is refused. This pre-qualifying step saves time by eliminating the need for processing bounced mail through your server. It also provides the sender with an immediate opportunity to correct the unrecognized address.

Note: *Response to a refusal of this type is under control of the individual mail client or mail server attempting to send the message. Most NT clients offer clear explanations that are easily understood, but be forewarned that sendmail (a commonly used UNIX client) places the offending message in a dead letter file. For this reason it is advised that the option be used with caution. In general, it is not recommended that this feature be used in a UNIX environment.*

Wildcard@Domain addressing causes further complications by allowing the Postmaster to specify an account for receipt of messages sent to an unknown address within a local mail domain.

When a wildcard address is included in a mail account, that account will receive all mail addressed to the designated domain unless an exact match for the message's Rcpt To: address is found among the addresses specified for a local mail account. This process effectively short circuits the mail routing hierarchy.

The system proceeds through steps 1-3 and then seeks a match for the delivery address among the standard Internet addresses in the accounts database. If a match is found, the message will be delivered to the appropriate local account. If a match is not found, the system will look for an address in the form `*@domain`. If a match is found, the message will be delivered to the account containing the wildcard address.

Great care should be taken in using this feature as it offers the potential for an abuse of privacy. Mail intended for another recipient may end up in the wildcard account as a result of a spelling error. As a result, the person reviewing messages delivered to the wildcard account may have access to mail that was intended to be confidential. If your intention is to avoid the handling of mail addressed to Unknown Users, a better choice would be to avoid using wildcard addressing and instead set the error response option for Unknown Local Users (found on the Error Response Parameters web form) to Return to Sender. This results in the automatic return of all mail sent to an unknown address within your local mail domain.

10.2.2 Handling of Mailing Lists Messages

Mailing list correspondence is sent to one of the following three addresses.

- Submissions for posting are sent to the list address.
- Requests (for subscription, subscriber information, and the like) are sent to the list's request address.
- Mail intended for the list owner is sent to the list owner alias address.

The handling of such mail varies based on the address to which it was addressed. When Post.Office searches the accounts database and finds a match for the "Rcpt To:" address of a message among the information for a mailing list, it notes the type of address for which the match was found and processes the message appropriately. Detailed descriptions of the processing for each address type appear in the sections that follow.

Delivery of Mail to a List Address

Messages are sent to a list address with the intention of having them forwarded to all subscribers. If the list is defined without restriction, this is exactly what will happen. However, before posting can occur the following checks are made:

Is the message a request that was sent to the wrong address (i.e., does it appear to contain a mailing list request that should have been sent to the list request address)?

If a request keyword (like `subscribe`) is sent to a list address and the Detect Requests option is turned on, the message will be returned to sender with an error message stating that: "`A request was posted to the list`." No further processing will occur.

Is the user a subscriber?

Post.Office checks the sender's address against each subscriber address for the list. If a match is found, the sender is considered a subscriber, and the submission is handled according to the list's posting policy for subscribers.

If the posting policy is "Open," the message is immediately posted to the list. If the posting policy is "Moderated," the message is held pending list owner approval. If the policy is "Closed With Notification," the message is returned to the sender with an error stating the list is closed, *and* a message is sent to the list owner advising them of the rejection. If the policy is "Closed Without Notification," the sender still receives an error message, but the list owner is not notified.

If the sender of the message is *not* a subscriber to the list, Post.Office proceeds to the next question.

What is the posting policy for remote users?

Next, Post.Office checks the list's posting policy for remote users. As in the previous example, if the posting policy is "Open," the message is immediately posted. If the posting policy is "Moderated," the message is held for approval. If the policy is "Closed With Notification," it is returned to the sender with an error message, and the list owner is notified. If the policy is "Closed Without Notification," the sender still receives an error message, but the list owner is not notified.

What happens to messages requiring list owner approval?

As noted above, if the list's posting policy is "Moderated," list owner approval is required before posting can occur. If the owner approves the message, it is immediately posted to the list. If the list owner rejects the message, it is discarded (without notice to the sender).

Delivery of Mail to a List Request Address

As the name suggests, messages sent to a list request address are requesting something – information, access, etc. To handle request messages, the system asks and answers the following questions:

Is this a valid request?

Requests must be formatted is a specific manner in order to be recognized and accepted by Post.Office. If the formatting is incorrect, the message will be returned to the sender with an error noting the reason for rejection. On the other hand, if the formatting is valid Post.Office proceeds to the next step in the process. See Chapter 7 for example of the proper request message format.

Is this a request for list information?

Post.Office recognizes the keyword `info` as a request for mailing list information. Access to mailing list information is unrestricted, so the system responds immediately with the content of the mailing list's long description.

Is this a request for subscriber information?

Post.Office recognizes the keyword `who` as a request for access to the list of subscribers. Access to this information is controlled by entries in the Subscriber List Access field. If the address from which the request was received meets the criteria established by those restrictions, the subscription list is provided by return mail. If the address fails to meet the required criteria, the request is denied and the sender receives an error message stating: "You don't have access to the subscriber list."

Is this a request for subscription to the list?

The keyword `subscribe` is recognized as a request for subscription. Subscription may or may not be subject to verification. If the verification option is turned on, a request for verification is returned to the subscribing address. If no response is received, the original request is ignored. If verification is confirmed (by return mail), the system continues processing the subscription request.

Once past the verification hurdle, the next step is to ascertain the identity of the individual requesting subscription – are they or are they not a local user (someone with a mail account on the server). Post.Office compares the address from which the request was received to the addresses in its accounts database. If a match is found, the request is from a local user. If not, it's from a remote user.

Post.Office then checks the subscription policy for the appropriate user type. If the subscription policy is "Open," the user is subscribed to the list and notified by e-mail. If the subscription policy is "Moderated," the request is held pending list owner approval. If the policy is "Closed With Notification," the request is denied and the sender is advised via e-mail. If the policy is "Closed Without Notification," the sender still receives a rejection message, but the list owner is not notified.

Is this a request for unsubscription from the list?

The keyword `unsubscribe` indicates a desire to cancel list subscription. Unsubscription may or may not be subject to verification. If the verification option is turned on, a request for verification is returned to the requesting address. If no response is received, the original request is ignored. If verification is confirmed (by return mail), the system continues processing the unsubscription request.

Unsubscription may also be moderated. If the moderation option is turned on, the request is held subject to list owner approval.

What happens to messages requiring list owner approval?

Requests requiring moderation are reviewed by the list owner. If the owner approves the request, it is granted immediately. If the list owner rejects the request, it is denied.

Delivery of Mail to a List Owner Alias Address

People who wish to correspond with the owner of a list send mail to the list owner alias address. This address protects the privacy of the list owner by providing them with a means of anonymity. It also allows easy, behind-the-scenes transfers of list ownership.

Messages to the list owner alias address are immediately forwarded to the mail account of the responsible party, then handled according to the delivery method specified there.

The Influence of List Limits

All mailing list correspondence is subject to the global limits for a list. If any of these limits are exceeded, the message will be rejected, and returned to the sender with notification of the offending condition.

Not all lists have established limits, but possible limits involve:

- Maximum number of subscribers
- Maximum kilobytes per message
- Maximum messages submitted per day
- Maximum total kilobytes submitted per day

Note: *The limits imposed for maximum messages per day and maximum kilobytes per day apply to all messages submitted to the list, regardless of the number approved for posting.*

10.3 Error Messages

Routine error handling (misaddressed messages, list mail that bounced, etc.) is covered in the discussion of system monitoring found in Chapter 8. However, before you turn there, you may want to review the error message you received carefully. It should be self-explanatory. Certain messages are only notices, and require no further action on your part, though it is generally good practice to keep track of errors since they are occasionally symptoms of a deeper malaise of your mail system.

More cryptic error codes are defined in the Post.Office FAQ which is available via the World Wide Web at http://www.software.com.

10.4 Internal Mail Handling

Each message accepted by the Post.Office mail server is immediately divided into three files: Header, Body, and Control. The Header and Body files (which contain the message's header information and body text) are stored together in a temporary location, while the Control file circulates through the system's modules for processing. Once the appropriate delivery action has been determined, the Header and Body are reunited in a single message file that is deposited in the user's mailbox, forwarded to a program, or sent to another mail server for external delivery.

Usually this operation happens so quickly that the temporary storage phase is transparent. Occasionally, however, mail is held for interaction with a list owner or Postmaster. In such cases, it can be useful to know where the mail is stored in the interim.

Header, Body and Control File Storage

In all cases, Header and Body files are stored together until internal processing is complete. They are held in the messages directory which is located within the Post.Office Spooling directory (`Spooling directory/messages`).

Note: *The location of the Spooling directory is under user control and is established when the Post.Office software is first installed. Check the Licensing/Configuration web form for the exact location of your mail server's Spooling directory.*

Control files are stored in various directories depending on the reason mail is being held. The list below identifies the reasons mail may be held and indicates the location of the Control file in each case.

- Messages queued for servers that are temporarily unavailable.

  `Spooling directory/deferred/SMTP-Deliver/`*`unavailablehostname`*

- Messages with errors held for Postmaster action.

  `Spooling directory/deferred/Error-Handler`

- Messages that are queued pending program delivery.

  `Spooling directory/deferred/Program-Deliver`

- Messages submitted to a list that require list owner approval before posting.

  `Spooling directory/deferred/List-Exploder/approval/LUID`

- Messages being collected in a digest for later delivery.

  `Spooling directory/deferred/List-Scheduler/LUID`

As a rule, any Control file which remains in the `deferred` directory for more than a week should be subject to review. Do not, however, delete Control files capriciously. All three message files (Control, Header, and Body) must be handled as a group. Deleting one or two files from a set can interfere with proper handling of that message.

Note: *The three files for a single message are easily identified because each file name consists of a common message number followed by text which identifies the item uniquely (as Control, Header, or Body).*

10.5 Troubleshooting Tools and Techniques

This section provides an overview of the software commonly used by the Technical Support staff of Software.com in the course of researching possible problems with Post.Office. The functionality of this software is not covered in its entirety but simply to the extent that it is most useful when helping our customers successfully install, operate, and troubleshoot Post.Office.

10.5.1 Telnet

Telnet is a tool which allows the user access, usually with proper permissions, to a remote computer. This can be quite useful when diagnosing problems. SMTP servers and POP servers may be accessed across the Internet (without knowing a password) because of the way the SMTP and POP3 ports operate. These ports (25 and 110 respectively) are always listening for queries from remote computers. Telnet can be used to communicate with Post.Office through these ports and actually ask it some simple questions. Telnet is frequently used to learn the following:

- Are all the Post.Office servers running?
- Does a certain Internet address exist on the machine in question?

What follows is a simple telnet session which will illustrate these operations.

Sample NT Telnet Session to SMTP Port

The first test:

1. Run Telnet and choose Connect->Remote System
2. Type the host.domain of the computer you wish to connect with in the “Host Name:” field.
3. Specify the port you wish to connect with (Use “25” for SMTP and “110” for POP3).
4. Click on the “Connect” button and wait for your host to respond.

If Post.Office is running on the remote machine it will respond with something like the following:

```
“220 fido.software.com ESMTP server (post.office v3.5 evaluation
license) ready Tue, 9 Apr 1997 19:16:33 -040”
```

Receipt of the above tells you that Post.Office is running.

Now that you know that Post.Office is working, you can determine if a specific Internet address exists for an account. In the following example we will look for an address `jake@software.com`.

1. Type `expn jake@software.com`
2. Hit return.

If the account exists you will see something like the following:

```
250 jake@software.com (Jake the dog)
```

If the account does not exist, Post.Office will report the following:

```
550 Unknown address: <jake@software.com>
```


Sample UNIX Telnet Session to SMTP Port

To begin, from the command line, type "telnet" followed by a space and then the host.domain of the computer you wish to connect to followed by a space and then the port number and hit return.

Here's a sample Telnet session:

```
>telnet sparky.software.com 25
Trying 198.17.234.116 ...
Connected to sparky.software.com.
Escape character is '^]'.
220 sparky.software.com ESMTP server (post.office v3.5 evaluation
license) ready Mon, 15 Apr 1997 15:47:54 +0100
```

This information tells you that a Post.Office MTA is alive and listening at port 25. You can also telnet to other ports to determine their status.

Now that you know that Post.Office is working, you could then determine if a specific Internet address exists for an account. In the following example we will look for an address `jake@software.com`.

1. Type `expn jake@software.com`
2. Hit return.

If the account exists you will see something like the following:

```
250 jake@software.com (Jake the dog)
```

If the account does not exist, Post.Office will report the following:

```
550 Unknown address: <jake@software.com>
```

Information such as that gathered with these simple Telnet operations can help determine if help is needed in starting the Post.Office server, or if the problem is simply an issue of mail accounts that were set up incorrectly.

10.5.2 Nslookup

Nslookup is a utility that allows you to request information from DNS nameservers. This software tool usually comes packaged with Name Servers or with other DNS software. With nslookup you can discover whether mail is capable of being delivered to any host or virtual domain on the Internet. With this information you will be prepared to advise you on how to configure your DNS records in order to successfully use Post.Office.

Nslookup can be started by bringing up a DOS or UNIX window and typing “nslookup” at the prompt. From this point, there are a number of ways that nslookup can be used. Some of these different “functions” are summarized here. Each of these functions, with the exception of “server” can be initiated by typing “set q=*” or “set type=*” at the nslookup prompt.

- q=ns
- q=mx
- q=a
- q=soa
- q=ptr
- server

q=ns

This function tells nslookup that you want information about name servers that support a specific domain. For example, if you wanted to know which name servers support the top level domain “com” you would type the following at the nslookup prompt (note that there is a “.” after the “com”. This must be added in order to prevent nslookup from adding your domain to the end of “com” which would result in a search for “com.software.com”, which would be no fun…)

(The nslookup prompt appears as a “>“ in this example.)

```
>set q=ns
>com.
>com     nameserver = H.ROOT-SERVERS.NET
 com     nameserver = B.ROOT-SERVERS.NET
 com     nameserver = C.ROOT-SERVERS.NET
 com     nameserver = D.ROOT-SERVERS.NET
 com     nameserver = E.ROOT-SERVERS.NET
 com     nameserver = I.ROOT-SERVERS.NET
 com     nameserver = F.ROOT-SERVERS.NET
 com     nameserver = G.ROOT-SERVERS.NET
 com     nameserver = A.ROOT-SERVERS.NET
 H.ROOT-SERVERS.NET      internet address = 128.63.2.53
 B.ROOT-SERVERS.NET      internet address = 128.9.0.107
 C.ROOT-SERVERS.NET      internet address = 192.33.4.12
 D.ROOT-SERVERS.NET      internet address = 128.8.10.90
 E.ROOT-SERVERS.NET      internet address = 192.203.230.10
 I.ROOT-SERVERS.NET       internet address = 192.36.148.17
 F.ROOT-SERVERS.NET      internet address = 192.5.5.241
 G.ROOT-SERVERS.NET      internet address = 192.112.36.4
 A.ROOT-SERVERS.NET      internet address = 198.41.0.4
```

The list that nslookup provides is that of all of the name servers responsible for the domain “com” followed by their associated “A” records which match the name to its IP address.

q=mx

In order for mail to be delivered on the Internet, a mechanism must exist to convert the host.domain of a mail server machine (sparky.software.com) to its appropriate IP number. This can be done in two places, either with the "hosts" file on the machine sending mail or with a referenced DNS name server. A query for MX records can be very helpful in determining if mail can be delivered to a mail server. Here's how such an operation might look:

```
>set q=mx
>software.com
>software.com    preference = 20, mail exchanger = rex.software.com
 software.com    preference = 30, mail exchanger = smtp1.cerf.net
 software.com    nameserver = rex.software.com
 software.com    nameserver = allman.cabrillo.com
 software.com    nameserver = noc.cerf.net
 software.com    nameserver = rover.software.com
 rex.software.com       internet address = 198.17.234.33
 smtp1.cerf.net  internet address = 192.102.249.30
 allman.cabrillo.com     internet address = 206.29.8.1
 noc.cerf.net    internet address = 192.153.156.22
 rover.software.com   internet address = 198.17.234.110
>
```

The information this query returns tells us a number of things such as mail sent to `user@software.com` will first be delivered to a machine called `rex.software.com` which has an IP address of "198.17.234.33". If this attempt fails (maybe "rex" is not currently available), then a second attempt at delivery will be made to the backup mail server called "smtp1.cerf.net" with an IP address of "192.102.249.30". You can tell which machine is the "backup" by the numbers (in this case "20" and "30") found next to the words "mail exchanger". The protocol dictates that the first delivery attempt should be made to the machine with the lowest "preference" number (in this case "20"). In addition to this mail delivery information, there is a list of the name servers which are "Authoritative" for the domain (authoritative means they are responsible for maintaining the DNS records for the queried domain). There is also a list of "A" records for these name servers.

q=a

You and I like names, probably because they have meaning for us. With this in mind, the present day Internet is set up such that computers can be referenced with names which are usually easy to remember and work with (such as the computer `rex.software.com`). Computers, however, would regard this (if they could regard) as not too efficient and a sure sign of just how feeble minded their creators are. For this reason, there is another side of the Internet which allows for a number, called an "IP" (Internet Protocol) address, to represent a computer's name. Computers like this a whole lot, and when a user sends mail to his/her friend, the computers in between quickly convert the host.domain into an IP address. This they use amongst themselves until it's time to deliver the mail to the recipient human at which point they begrudgingly convert the IP number back into the user friendly host.domain. The DNS record which makes the correlation between the host.domain and its IP number is the "A" record (which stands for "address").

We can use NS lookup to ask for the "A" record of a certain host.domain or domain by setting "q=a". The following is an example of how this is done.

```
>set q=a
>rex.software.com
 Name:  rex.software.com
 Address:  198.17.234.33
>
```

This information, being so crucial to mail delivery, can help determine if a certain host.domain or domain is resolvable via DNS.

q=soa

If, in the course of your Internet administration, you find it necessary to either learn more about a certain name server or perhaps you even wish to contact that name server's administrator, you can use nslookup with "q=soa" to get the goods. Here's an example.

(By the way, "sao" means "Source of Authority)

```
>set q=soa
>software.com
 software.com
 origin = rover.software.com
 mail addr = bindmaster.software.com
 serial = 9604090
 refresh = 43200 (12 hours)
 retry   = 7200 (2 hours)
 expire  = 1209600 (14 days)
 minimum ttl = 43200 (12 hours)
 software.com    nameserver = rover.software.com
 software.com    nameserver = rex.software.com
 software.com    nameserver = allman.cabrillo.com
 software.com    nameserver = noc.cerf.net
 rover.software.com   internet address = 198.17.234.110
 rex.software.com       internet address = 198.17.234.33
 allman.cabrillo.com     internet address = 206.29.8.1
 noc.cerf.net    internet address = 192.153.156.22
>
```

As you can see, this query really delivers. Among the items you may already recognize are some new bits of information that are quite useful.

- **Origin**. The name server from which this information was retrieved.
- **Mail Addr**. If you put a "@" in place of the first period in the listed address you then have the email address for the system administrator of that name server. For example, "bindmaster.software.com" becomes "bindmaster@software.com".
- **Serial**. this number, when changed, tells the secondary name server (when it queries the primary) that it is time to refresh its records. This is necessary if the administrator makes any changes to the DNS records and wishes to have the backup DNS update with these changes.
- **Refresh**. this tells the secondary name server how often (in seconds) it should query the primary for record changes (basically, checking to see if the serial number has changed).

- **Retry**. If the secondary fails to contact the primary at the interval specified in the "refresh" setting, the "retry" setting tells it how long it should wait before it should try to contact the primary again.
- **Expire**. If the secondary fails to contact the primary at the interval specified in the "refresh" setting, and continues to fail to connect during subsequent attempts, the "expire" setting tells it how long it should continue giving out information to querying name servers and resolvers before it is to stop giving out information.
- **Minimum ttl**. "Minimum ttl" stands for "Minimum Time to Live" and refers to how long querying name servers should keep the information given out by this name server. This is important when making DNS changes as it will take at least as long as the time specified in this setting before other name servers with cached information from this name server will once again make a fresh query to this name server for DNS information. Therefore, if your customer changes this setting he/she cannot expect mail to operate under the new parameters until this machine's ttl has expired on all outstanding name servers.

q=ptr

Sometimes it is necessary to see if an IP number has a corresponding host.domain. When this needs to be done, the "q=ptr" is a handy function. Here's an example. (PTR is short for "Pointer")

```
>set q=ptr
>198.17.234.110
110.234.17.198.in-addr.arpa     name = rover.software
234.17.198.IN-ADDR.ARPA nameserver = rover.software.com
234.17.198.IN-ADDR.ARPA nameserver = rex.software.com
234.17.198.IN-ADDR.ARPA nameserver = noc.cerf.net
234.17.198.IN-ADDR.ARPA nameserver = allman.cabrillo.com
rover.software.com   internet address = 198.17.234.11
rex.software.com       internet address = 198.17.234.33
noc.cerf.net    internet address = 192.153.156.22
allman.cabrillo.com     internet address = 206.29.8.1
>
```

Such a DNS query is called a "reverse lookup". You may note that the first set of IP numbers in the results are backwards and seem to end with "IN-ADDR.ARPA". This is another convention of the DNS realm that is vital for proper name resolution.

Hint: *Such information is helpful to us when assisting Post.Office customers. Most frequently, we would perform such a search when a customer complains that he/she is unable to access Post.Office via the web. Post.Office uses reverse lookup when it attempts to verify that a connecting web client's host is within any specified "Access Domains". If the host.domain cannot be referenced in a reverse lookup it will fail to qualify for access.*

server

DNS queries begin when a computer's resolver asks the name server it has listed as its primary contact a question. It is possible to change this initial contact when using nslookup by specifying this with the "server" function. Here's an example.

```
> server rover.software.com
 Default Server:  rover.software.com
 Address:  198.17.234.110
>
```

You may not find yourself doing this much, but it's a good trick to have up your sleeve.

10.5.3 Ping

Ping is a utility which allows you to determine if a remote computer is up and running. This is useful if you wish to isolate a problem and you suspect that the remote computer may not be responding to network queries. Although the information received from a Ping query may not be very rich in content, it can be conclusive.

```
C:\>ping 198.17.234.116
Pinging sparky.software.com [198.17.234.116] with 32 bytes of data:
Reply from 198.17.234.116: bytes=32 time<10ms TTL=32
Reply from 198.17.234.116: bytes=32 time<10ms TTL=32
Reply from 198.17.234.116: bytes=32 time<10ms TTL=32
Reply from 198.17.234.116: bytes=32 time<10ms TTL=32
C:\>
```

As you can see from the example. The computer from which the ping query was issued delivered four separate packets each with 32 bytes of data. If these packets are received by the pinged host, it then issues a response which is what appears in the example.

Each "reply" displayed on the screen gives us a short profile of the packet which was sent and subsequently returned. This information includes the following:

- **IP#**. This is the address of the queried host.
- **Bytes**. This tells you the size (in bytes) of the packet sent.
- **Time**. This is the time it took for the individual packet which was sent to reach and then return from the queried host. If the round-trip query took less than 10 milliseconds the exact time will not be indicated and instead a <10 will be displayed.
- **TTL**. If a packet was to become misrouted or "lost" on the network, the TTL, or Time to Live, setting will keep the packet from wandering about indefinitely by indicating a maximum lifespan after which it is to expire.

11

Post.Office Utilities

This chapter discusses the several special purpose utilities which are distributed with Post.Office. These utilities allow you to interact with Post.Office from the command line and facilitate the handling of large groups of repetitive transactions. The topics in the this chapter include:

- Instructions for using the system configuration, account management, and mailing list management utilities
- The available system configuration utilities
- The available account management utilities
- The available mailing list management utilities
- The postmail utility
- The Post.Office sendmail replacement utility

11.1 Executing the Utilities

This section pertains to the execution of the Post.Office command-line utilities for account management, mailing list management, and system configuration. This information does *not* apply to the postmail utility or sendmail replacement utility. Postmail is a command line mail client that has its own execution instructions, as described in Section 11.5. The sendmail replacement utility is described in Section 11.6.

Note that the instructions differ by operating system. Please refer to the appropriate section below for instructions

11.1.1 Windows NT

To execute the utilities, you must log in to the server system as a member of the administrator user group. Most utilities can be executed regardless of whether or not Post.Office is running, so you may shut down Post.Office before using the utilities if you wish.

The utilities are installed to the `Post.Office\cmdutils` directory. However, because many utilities create files which can clutter up this directory, you should execute the utilities from a working directory where you plan to store the user profiles, list profiles, or other information returned by the utilities. This directory is not added to the system's PATH variable on installation, so if you want to use the utilities, you should update your PATH to include this directory.

11.1.2 UNIX

To execute the account and mailing list management utilities, you must log in to the server system as root. Most utilities can be executed whether Post.Office is running or not,[54] so you may shut down Post.Office before using the utilities if you wish.

The utilities are installed in the `post.office/cmdutils` directory. However, because many utilities create files which can clutter up this directory, you should execute them from a working directory where you plan to store the user profiles, list profiles, or other information returned by the utilities. This directory is not added to the system's PATH environment variable on installation, so if you want to use the utilities, you should update your PATH to include this directory.

11.2 System Utilities

The system utilities provide you with directory locations of Post.Office files on the server file system. This information is useful during troubleshooting. Although these directory locations can be viewed in the Licensing/Configuration Form described in Chapter 4, Post.Office must be running and you must log in to the web interface to access this form. The system utilities, on the other hand, can get this information regardless of whether Post.Office is running.

Utility	Comment
`getmailboxdir`	returns the Post.Office mailboxes directory
`getspooldir`	returns the Post.Office spool directory

11.2.1 getmailboxdir – Get Mailbox Directory Utility

The `getmailboxdir` utility displays the full path on the server file system to the Post.Office mailbox directory.

Usage

```
getmailboxdir
```

54 The utilities for deleting accounts and mailing lists, respectively, require Post.Office to be running at time of execution.

11.2.2 getspooldir – Get Spool Directory Utility

The `getspooldir` utility displays the full path on the server file system to the Post.Office spool directory.

Usage

```
getspooldir
```

11.3 Account Management Utilities

The account management utilities have been designed to aid administrators of larger installations in routine account maintenance, and provide a facility for executing high-volume operations, such as the creation of multiple user accounts. These utilities can also help to automate the moving of accounts from one installation of Post.Office to another, or from another mail server into Post.Office.

11.3.1 Utilities Summary

The following table summarizes the Post.Office account management utilities. Each utility is described in greater detail later in this section.

Utility	Comment
`addacct`	adds a user account to Post.Office
`changeacct`	modifies an existing account
`delacct`	deletes an existing account
`getacct`	gets the user profile for an existing account
`getpopmbox`	gets the full path to the user's mailbox directory on the server file system
`getuid`	gets the account UID (the unique identifier) of an existing account
`listacct`	lists specified account info, for specific accounts or all accounts
`lockacct`	locks an account, preventing POP delivery and user modification
`reportusage`	reports the POP mailbox usage
`unlockacct`	unlocks an account, restoring POP delivery and allowing user modification

11.3.2 Definitions

The following terms are used in this section:

- **address**. SMTP e-mail address of a user, in the format *`user@host.domain`*.
- **user account**. User information stored in the Post.Office user accounts database (MTA-Accounts).
- **user profile**. A formatted text file containing the information of a user account. User profiles are named after the account UID, with an `.acct` file extension. Examples of user profiles are given in the next section.
- **Name**. The real name of a user as found in the Name field of a user profile. This name corresponds to the Real Name field of the Post.Office Account Data Form.
- **UID**. A unique string used to identify an account within Post.Office. By default, the UID for an account is the Real Name, with spaces and other non-alphanumeric characters replaced by underscore (_) characters. The UID of an account is set at time of account creation and cannot be modified. The UID of an account is displayed at the bottom of the Account Data Form.
- **POP name**. The POP3 login name of the specified user (does *not* contain the "@" symbol or host/domain names).

11.3.3 User Profile Form

Creating and modifying Post.Office accounts with the `addacct` and `changeacct` utilities requires a user profile, a specifically-formatted text form that contains account information. User profiles can be passed to these utilities from a file or from the standard input. When given in a file, the file must be named after the account UID, with a `.acct` file extension. For example, modifying the account associated with the UID `John_Doe` requires a file named `John_Doe.acct` which contains the properly-formatted new account values.

The following is a blank user profile file. The new data for each account attribute is provided between the brackets next to the appropriate field label.

```
Access-Domains: []
AutoReply-Info: []
AutoReply-Mode: []
Directory-Access: []
Finger-Access: []
Finger-Info: []
Forward-Delivery: []
Handler-Delivery: []
Home-URL: []
Local-Delivery: []
Logon-Id: []
Mailbox-Quota: []
Name: []
POP-Address: []
ProgDel-Account: []
Program-Deliver-Info: []
Raw-Password: []
SMTP-Address: []
SMTP-RewriteStyle: []
Use-Logon-PW: []
```

Items in the profile for which no data is specified will assume the default value during account creation, or the existing value during account modification. However, when creating accounts, the following items require data to be provided: Name, Password, SMTP-Address, and either Local-Delivery or Forward-Delivery.

Some items in the user profile form can contain only single values, while others allow for multiple values. Some are also limited to a range of specific values. The following table lists the attributes of each user profile item:

Item	Values	Limited to
`Access-Domains`	multiple	legal hostnames, domains, or IP addresses
`AutoReply-Info`	multiple	ASCII text
`AutoReply-Mode`	single	must be one of: vacation, reply, echo
`Directory-Access`	single	any of: d (default), l (local only), r (local and remote), u (unlisted)
`Finger-Access`	multiple	legal hostnames, domains, or IP addresses
`Finger-Info`	multiple	ASCII text
`Forward-Delivery`	multiple	legal RFC821 addresses (e.g., user@domain)
`Handler-Delivery`	single	empty string or AutoReply-Handler
`Home-URL`	single	fully-qualified URL, including the protocol identifier (http, ftp, etc.)
`Local-Delivery`	multiple	one or more of: Mailbox, UNIX, Program
`Logon-Id`	single	Username of NT account corresponding to Post.Office account (NT platforms only)
`Mailbox-Quota`	single	integer, units in Kbytes
`Name`	single	ASCII text
`Password`	single	6 characters minimum, cannot start or end with space/tab
`POP-Address`	single	cannot contain spaces or ‘@’
`ProgramDel-Account`	single	Username of account that has access to execute Program Delivery applications (NT platforms only)
`Program-Deliver-Info`	multiple	ASCII text
`Raw-Password`	single	Encrypted password value. This value should *not* be modified.
`SMTP-Address`	multiple	legal RFC821 addresses (e.g., user@domain)
`SMTP-RewriteStyle`	single	must be one of: comment, quoted, none
`UNIX-UserName`	single	same restrictions as UNIX user names (UNIX platforms only)
`Use-Logon-PW`	single	yes/no (NT platforms only)

Multiple values are represented separately in bracket pairs on different lines. For example, in the following user profile, the SMTP-Address field specifies multiple e-mail addresses for this account:

```
Access-Domains: [software.com]
AutoReply-Info: [Attending E-Mail World in Chicago,]
                [returning June 17.]
AutoReply-Mode: [vacation]
Directory-Access: [d]
Finger-Access: []
Finger-Info: []
Forward-Delivery: [SMTP <mojo@someisp.net>]
Handler-Delivery: []
Home-URL: [http://home.software.com/~mojo]
Local-Delivery: [Mailbox]
Logon-Id: []
Mailbox-Quota: [1000]
Name: [Max Johnson]
POP-Address: [mojo]
ProgDel-Account: []
Program-Deliver-Info: []
Raw-Password: [be813fdc029ec0dcf2aec1bb4f7bc7b88605855f]
SMTP-Address: [max@software.com]
              [maxj@software.com]
              [max.johnson@software.com]
              [mojo@software.com]
SMTP-RewriteStyle: [comment]
Use-Logon-PW: [no]
```

11.3.4 addacct – Add Account Utility

The `addacct` utility adds a user account to Post.Office, given a user profile. By default, user profile information is taken from a file, but can be taken instead from standard input by including "`-`" on the command line after the account UID. The required structure for a user profile is illustrated in the previous section.

Usage

```
addacct UID [-]
```

Example

```
addacct John_Doe
```

This command creates an e-mail account, based on the user profile contained in the file named `John_Doe.acct`.

```
your_program | addacct John_Doe -
```

Redirects output of program `your_program` to `addacct`, which uses it as the user profile data for account `John_Doe`.

11.3.5 changeacct – Change Account Data Utility

The `changeacct` utility updates an existing account with information from a given user profile. By default, user profile information is taken from a file, but can be taken instead from standard input by including "`-`" on the command line after the account UID.

Usage

```
changeacct UID [-]
```

Example

```
changeacct John_Doe
```

Updates the account corresponding to the UID `John_Doe` with information contained in the user profile file `John_Doe.acct`.

```
your_program | changeacct John_Doe -
```

Redirects output of program `your_program` to `changeacct`, which uses it as the user profile data for account `John_Doe`.

11.3.6 delacct – Delete Account Utility

The `delacct` utility deletes an existing account, given an account UID or address. This utility does not operate directly on the accounts database (as do the other utilities), so Post.Office *must* be running when you use this utility.

Usage

```
delacct UID
```

Example

```
delacct John_Doe
```

Deletes the account associated with this UID.

11.3.7 getacct – Get User Account Profile Utility

The `getacct` utility gets the user profile for a Post.Office account, given an account UID. The user profile is returned to the standard output, or to a file named *`UID`*`.acct`.

Although the account password is one of the items returned by `getacct`, this value is returned in encrypted format, which can be subsequently used by other Post.Office utilities. Unencrypted passwords cannot be retrieved from the accounts database.

Usage

```
getacct UID [-]
```

Example

```
getacct John_Doe
```

Returns user profile information to the file `John_Doe.acct`.

```
getacct John_Doe -
```

Prints the user profile for this account to standard output.

11.3.8 getpopmbox – Get POP Mailbox Directory Utility

The `getpopmbox` utility returns the full path to a user's POP mailbox on the server file system, given an account UID. This information is useful for system operations such as moving mailboxes from one drive to another.

Usage

```
getpopmbox UID
```

Example

```
getpopmbox John_Doe
```

This command returns the full path to the POP mailbox directory for the account `John_Doe`.

11.3.9 getuid – Get User ID Utility

The `getuid` utility returns the account UID corresponding to a given address, POP login name, or account Real Name to the standard output.

Usage

```
getuid address
getuid POPname
getuid realname
```

Example

```
getuid john.doe@software.com
```

Retrieves the account UID for the account associated with this address and prints it to the standard output.

```
getuid "John Doe"
```

Gets the account UID for the account associated with this Real Name.

11.3.10 listacct – List Account Data Utility

The `listacct` utility returns specified user profile account attributes, for all users or for a specific user, to the standard output. The requested account attributes are included as command line parameters, and can include any of the following items:

`Name`	`Account-ID`
`Password (encrypted)`	`NT-UsePassword`
`Login-ID`	`SMTP-Address`
`SMTP-RewriteStyle`	`POP-Address`
`UNIX-UserName`	`Local-Delivery`
`Handler-Delivery`	`Account-Delivery`
`Channel-Delivery`	`Manager-Delivery`
`Forward-Delivery`	`Program-Deliver-Info`
`HTML-Info`	`AutoReply-Mode`
`AutoReply-Info`	`Access-Domains`
`Finger-Access`	`Finger-Info`

Although the account password is one of the items that can be returned by `listacct`, this value is returned in encrypted format, which can be subsequently used by other Post.Office utilities. Unencrypted passwords cannot be retrieved from the accounts database.

By default, requested values are comma-separated. However, you can specify a different separator by using the `-s` flag, which must be followed by the new separator (enclosed in ‘single quotes’).

Usage

```
listacct [-s 'separator'] -i [item],[item],... [UID]
```

Example

```
listacct -i Account-ID,Name
```

Prints the UID and Real Name values for all e-mail accounts to the standard output. Values are comma separated.

```
listacct -i POP-Address John_Doe
```

Prints the POP-Address for the account associated with UID John_Doe.

```
listacct -s ':' -i Account-ID,POP-Address
```

Prints the account UID and POP name for all accounts, using the colon (:) character to separate values.

11.3.11 lockacct – Lock Account Utility

The `lockacct` utility locks the specified account. Locking an account prevents user access to account information, and also prevents delivery of mail via POP. Accounts remain locked until unlocked with the `unlockacct` utility or via the web interface.

Usage

```
lockacct UID
```

Example

```
lockacct John_Doe
```

Locks the account associated with the UID `John_Doe`, preventing POP mail delivery and account modification.

11.3.12 reportusage – Report POP Mailbox Usage Utility

The `reportusage` utility returns the POP mailbox usage for the specified Post.Office account to the standard output.

Usage

```
reportusage UID
```

Example

```
reportusage John_Doe
```

Reports the current POP mailbox usage for the account associated with the UID `John_Doe`.

11.3.13 unlockacct – Unlock Account Utility

The `unlockacct` utility is used to unlock previously locked accounts.

Usage

```
unlockacct UID
```

Example

```
unlockacct John_Doe
```

Unlocks the account associated with the UID `John_Doe`, allowing POP mail delivery and account modification.

11.4 Mailing List Management Utilities

Like the account management utilities, the mailing list management utilities have been designed to aid administrators of larger installations in routine mailing list maintenance, and provide a facility for executing high-volume operations, such as the creation of multiple mailing lists.

11.4.1 Utilities Summary

The following table summarizes the Post.Office mailing list management utilities. Each utility is described in greater detail later in this section.

Utility	Comment
`addlist`	adds a mailing list to Post.Office (requires all mailing list data)
`addlistshort`	adds a mailing list to Post.Office (requires minimum list data)
`changelist`	modifies an existing mailing list
`deletelist`	deletes an existing mailing list
`getlist`	gets the mailing list profile for a mailing list
`listmlists`	gets the ULID of a specific mailing list or all mailing lists
`listsubscribers`	gets the list of subscribers for a mailing list
`subscribe`	subscribes a user to a mailing list, regardless of list policies
`unsubscribe`	unsubscribes a user from a mailing list, regardless of list policies

11.4.2 Definitions

The following terms are used in this section:

- **list profile**. A formatted text file containing the information of a mailing list. List profiles are named after the list ULID, with a `.list` file extension. Examples of list profiles are given in the next section.
- **ULID**. A unique string used to identify an mailing list within Post.Office. The ULID for a mailing list is always the same as its List Name, so you think of this as the List Name if you'd like. Like the List Name, the ULID is set at time of mailing list creation and cannot be modified. The ULID of a mailing list is displayed at the bottom of the Mailing List Data Form.

11.4.3 List Profile Form

Creating and modifying mailing lists with the `addlist` and `changelist` utilities requires a list profile, a specifically-formatted text form that contains mailing list information. List profiles can be passed to these utilities from a file or from the standard input. When given in a file, the file must be named after the list ULID, with a `.list` file extension. For example, modifying the mailing list associated with the ULID `employees` requires a file named `employees.list` which contains the properly-formatted new mailing list values.

The following is a blank list profile file. The new data for each mailing list attribute is provided between the brackets next to the appropriate field label.

```
Subscriber-List-Access: []
Finger-Access: []
Finger-Info: []
List-Address-Expansion-Style: []
List-Approved-Posters: []
List-Digest-Restriction: []
List-Digest-Schedule: []
List-Epilogue: []
List-Headers-To-Add: []
List-Headers-To-Remove: []
List-Local-Domains: []
List-Local-Subscriber-Policy: []
List-Locked: []
List-Long-Info: []
List-Max-Kbytes-Per-Day: []
List-Max-Message-Kbytes: []
List-Max-Messages-Per-Day: []
List-Max-Subscribers: []
List-Moderate-Unsubscriptions: []
List-Moderation-Method: []
List-Name: []
List-Nonsubscriber-Posting-Policy: []
List-Other-Rewrites: []
List-Owners: []
List-Priority: []
List-Prologue: []
List-Remote-Subscriber-Policy: []
List-Remove-X-Headers: []
List-Request-Detection: []
List-Short-Info: []
List-Stats-Mode: []
List-Subscriber-Posting-Policy: []
List-Suppress-Duplicates: []
List-Suppress-Notifications: []
List-Unsubscribe-Info: []
List-Verify-Subscriptions: []
List-Verify-Unsubscriptions: []
List-Welcome-Info: []
ListMgr-SMTP-Address: []
ListOwner-SMTP-Address: []
SMTP-Address: []
```


Items in the list profile for which no data is specified will assume the default value during mailing list creation, or the existing value during mailing list modification. Some items in the list profile form can contain only single values, while others allow for multiple values. Some are also limited to a range of specific values. The following table lists the attributes of each list profile item:

Item	Values	Limited to
`Finger-Access`	multiple	legal hostnames, domains, or IP addresses
`Finger-Info`	single	ASCII text
`List-Address-Expansion-Style`	single	One of the following: none, group, expand
`List-Approved-Posters`	multiple	legal RFC821 addresses (user@domain)
`List-Digest-Restriction`	single	One of the following: none, digest-only, immediate-only
`List-Digest-Schedule`	multiple	One or more of the following: daily, weekly. Individual days and times can also be specified as in the Mailing List Data Form; see Chapter 7 for the correct format
`List-Epilogue`	single	ASCII text
`List-Headers-To-Add`	multiple	legal RFC821 headers (X-Post-Office:)
`List-Headers-To-Remove`	multiple	legal RFC821 headers (X-Post-Office:)
`List-Local-Domains`	multiple	legal host and domain names
`List-Local-Subscriber-Policy`	single	One of the following: open, moderated, closed-notify, closed
`List-Locked`	single	yes/no
`List-Long-Info`	single	ASCII text
`List-Max-Kbytes-Per-Day`	single	integer, units in Kbytes
`List-Max-Message-Kbytes`	single	integer, units in Kbytes
`List-Max-Messages-Per-Day`	single	integer
`List-Max-Subscribers`	single	integer
`List-Moderate-Unsubscriptions`	single	yes/no
`List-Moderation-Method`	single	One of the following: queue-no-mail, queue-notify, noqueue-fwd-nomime. These correspond to the moderation modes Web Only, Web and E-mail, and E-mail Only.
`List-Name`	single	letters (A-Z, a-z), numbers (0-9), addition (+), subtraction (-), and underscore (_)

Item	Values	Limited to
`List-Nonsubscriber-Posting-Policy`	single	One of the following: open, moderated, closed-notify, closed
`List-Other-Rewrites`	multiple	See Chapter 7 for the correct syntax for defining header rewriting
`List-Owners`	multiple	legal RFC821 addresses (user@domain) for a local user account
`List-Priority`	single	One of the following: low, normal
`List-Prologue`	single	ASCII text
`List-Remote-Subscriber-Policy`	single	One of the following: open, moderated, closed-notify, closed
`List-Remove-X-Headers`	single	yes/no
`List-Request-Detection`	single	yes/no
`List-Short-Info`	single	ASCII text
`List-Stats-Mode`	single	on/off
`List-Subscriber-Posting-Policy`	single	One of the following: open, moderated, closed-notify, closed
`List-Suppress-Duplicates`	single	yes/no
`List-Suppress-Notifications`	multiple	One or more of: FilterBounceNotice, OverLimitNotification
`List-Unsubscribe-Info`	single	ASCII text
`List-Verify-Subscriptions`	single	yes/no
`List-Verify-Unsubscriptions`	single	yes/no
`List-Welcome-Info`	single	ASCII text
`ListMgr-SMTP-Address`	multiple	legal RFC821 addresses (user@domain)
`ListOwner-SMTP-Address`	multiple	legal RFC821 addresses (user@domain)
`SMTP-Address`	multiple	legal RFC821 addresses (user@domain)
`Subscriber-List-Access`	multiple	legal hostnames, domains, or IP addresses

Multiple values are represented separately in bracket pairs on different lines. For example, in the following list profile, the addresses fields specify multiple e-mail addresses for this mailing list:


```
Subscriber-List-Access: [subscribers]
Finger-Access: []
Finger-Info: [To subscribe to this mailing list, send a]
             ['subscribe' request to the following address:]
             [anglers-request@software.com]
List-Address-Expansion-Style: [none]
List-Approved-Posters: [<postmaster@software.com>]
List-Digest-Restriction: [none]
List-Digest-Schedule: [daily 5 pm]
List-Epilogue: [--------------------------------]
               [To unsubscribe from this list, send an ]
               ['unsubscribe' request to the following address:]
               []
               [anglers-request@software.com]
List-Headers-To-Add: [Reply-To: anglers@software.com]
List-Headers-To-Remove: [Reply-To:]
List-Local-Domains: [rex.software.com]
List-Local-Subscriber-Policy: [moderated]
List-Locked: [no]
List-Long-Info: [This is a group of folks here in Santa Barbara, ]
                [CA who like fishin'.]
List-Max-Kbytes-Per-Day: [1000]
List-Max-Message-Kbytes: [10]
List-Max-Messages-Per-Day: [20]
List-Max-Subscribers: [50]
List-Moderate-Unsubscriptions: [yes]
List-Moderation-Method: [queue-notify]
List-Name: [anglers]
List-Nonsubscriber-Posting-Policy: [closed-notify]
List-Other-Rewrites: [prefix subject: "anglers: "]
List-Owners: [<john.doe@software.com>]
List-Priority: [low]
List-Prologue: [--> The Anglers Mailing List <--]
List-Remote-Subscriber-Policy: [moderated]
List-Remove-X-Headers: [yes]
List-Request-Detection: [yes]
List-Short-Info: [Angler society of Santa Barbara]
List-Stats-Mode: [on]
List-Subscriber-Posting-Policy: [open]
List-Suppress-Duplicates: [no]
List-Suppress-Notifications: [FilterBounceNotice]
                             [OverLimitNotification]
List-Unsubscribe-Info: [Goodbye, angler ... may your bait be ]
                       [lively, and your catch plentiful. To come ]
                       [back into our fishin' family, send your ]
                       [subscription request to:]
                       [anglers-request@software.com]
List-Verify-Subscriptions: [no]
List-Verify-Unsubscriptions: [yes]
List-Welcome-Info: [Welcome, fellow fisherperson, to the angler's ]
                   [mailing list. Send your insightful fishing ]
                   [comments to:]
                   [anglers@software.com]
ListMgr-SMTP-Address: [<anglers-request@software.com>]
ListOwner-SMTP-Address: [<owner-anglers@software.com>]
SMTP-Address: [anglers@software.com]
              [anglers@rex.software.com]
```


11.4.4 addlist – Add Mailing List Utility

The `addlist` utility creates a mailing list in Post.Office, given a list profile. By default, list profile information is taken from a file, but can be taken instead from standard input by including "`-`" on the command line after the list ULID. The required structure for a list profile is illustrated in the previous section.

Usage

```
addlist ULID [-]
```

Example

```
addlist employees
```

This command creates a mailing list, based on the list profile contained in the file named `employees.list`.

```
your_program | addlist employees -
```

Redirects output of program `your_program` to `addlist`, which uses it as the list profile data for the mailing list `employees`.

11.4.5 addlistshort – Add Mailing List Utility

The `addlistshort` utility offers a shortcut for creating mailing lists, and is similar to the New Mailing List – Short Form described in Chapter 7. It requires only a ULID and e-mail address for the owner of the new mailing list, and does not require a list profile. All other mailing list data for the new list is taken from the data specified in the Default Mailing List Data Form in the web interface.

Note: *Only one list owner can be specified when creating mailing lists with the addlistshort utility. To create mailing lists that have more then one owner, use addlist.*

Usage

```
addlistshort ULID owneraddress
```

Example

```
addlistshort employees john.doe@software.com
```

Creates a new mailing list which has the ULID `employees` and which is owned by the local user who has the e-mail address `john.doe@software.com`.

11.4.6 changelist – Change List Data Utility

The `changelist` utility updates an existing mailing list with information from a given list profile. By default, list profile information is taken from a file, but can be taken instead from standard input by including "`-`" on the command line after the list ULID.

Usage

```
changelist ULID [-]
```

Example

```
changelist employees
```

Updates the mailing list corresponding to the ULID `employees` with information contained in the list profile file `employees.list`.

```
your_program | changelist employees -
```

Redirects output of program `your_program` to `changelist`, which uses it as the list profile data for the mailing list `employees`.

11.4.7 deletelist – Delete List Utility

The `deletelist` utility deletes an existing mailing list, given a ULID. This utility does not operate directly on the lists database (as do the other utilities), so Post.Office must be running when you use this utility.

Usage

```
deletelist ULID
```

Example

```
deletelist employees
```

Deletes the list associated with this ULID.

11.4.8 getlist – Get List Profile Utility

The `getlist` utility gets the list profile for a Post.Office mailing list, given a ULID. The list profile is returned to standard output, or to a file named *`ULID`*`.list`. By default, list profile information is written to a file, but can be printed instead to standard output by including "`-`" on the command line after the list ULID.

Usage

```
getlist ULID [-]
```

Example

```
getlist employees
```

Returns list profile information to the file `employees.list`.

```
getlist employees -
```

Prints the list profile for this mailing list to standard output.

11.4.9 listmlists – Get List ULID Utility

The `listmlists` utility generates a list of mailing lists in Post.Office. It can also optionally display the current number of subscribers for each mailing list.

By default, the output of `listmlists` contains the names of all mailing lists. However, to display only those lists which are available for local subscription (i.e., the mailing lists that are visible to local users through the web and e-mail interfaces), you can include the UID of an existing Post.Office account as a command-line parameter.

Usage

```
listmlists [-subscriberCount] [UID]
```

Example

```
listmlists
```

Returns the ULID of all Post.Office mailing lists to the standard output.

```
listmlists John_Doe
```

Returns the ULID of all Post.Office mailing lists that are available to the local user whose UID is `John_Doe`.

```
listmlists -subscriberCount
```

Returns the ULID and subscriber count (separated by a colon character) of all Post.Office mailing lists. For example:

```
archery:23
anglers:13
surfing:4
```


11.4.10 listsubscribers – Get List Subscribers Utility

The `listsubscribers` utility returns the list of subscribers for the mailing list corresponding to a given ULID.

Usage

```
listsubscribers ULID
```

Example

```
listsubscribers employees
```

Returns the list of subscribers for the mailing list `employees`.

11.4.11 subscribe – Add Subscribers Utility

The `subscribe` utility adds users to a mailing list. This command can be used in a variety of ways, depending on how you want the users to be subscribed, and the source of the subscriber addresses. In all cases, the ULID of a specific mailing list must be included as an argument.

By default, this utility submits a subscription request for the given user(s) – a request which is subject to verification and list owner moderation. However, you can instead add the user directly to the subscriber list, regardless of subscription policies, by including the `-f` flag.

The address of the new subscriber must be either given as a command-line argument, entered from standard input, or specified in a file. Only one subscriber address can be specified on the command-line, so subscribing multiple addresses requires you to enter the subscriber addresses from standard input (specified by entering "`-i -`"), or from a file of addresses (specified by "`-i filename`").

Another option for this command is the `-q` flag, which suppresses the sending of welcome messages. By default, subscribers added to a mailing list with the subscribe utility receive the list's welcome message, just as if they had subscribed themselves to the mailing list. When `-q` is included, users who are added to the selected mailing list do not receive the welcome message.

Finally, you can request the mode of delivery (`digest` or `immediate`) for the new subscribers by specifying the appropriate keyword as the final command-line argument. You can omit a delivery mode if you like; in this case, the subscribers are added using the immediate mode of delivery (unless this mode is not supported by the list).

Usage

```
subscribe [-f] [-q] ULID address [ digest | immediate ]
subscribe [-f] [-q] -i filename ULID [ digest | immediate ]
subscribe [-f] [-q] -i - ULID [ digest | immediate ]
```


Example

```
subscribe employees susie.queue@software.com
```

Requests subscription for user `susie.queue@software.com` to the mailing list `employees`. This request may be denied, moderated, or subject to verification, depending on the subscription policies of the mailing list. The user will receive a welcome message if added to the list.

```
subscribe -f employees john.doe@software.com
```

Immediately adds the user `john.doe@software.com` to the mailing list `employees`, regardless of the subscription policies of this mailing list.

```
subscribe -f -q -i - League digest
```

Immediately adds multiple subscribers to the mailing list `League`, with the addresses of the new subscribers entered from standard input. These subscribers will not receive a welcome message, and are added with the digest mode of delivery.

```
subscribe -f -i roster.txt baseball_team immediate
```

Immediately adds the addresses specified in the file `roster.txt` to the mailing list `baseball_team`. These subscribers receive the list's welcome message, and are added with the immediate mode of delivery.

11.4.12 Unsubscribe – Remove Subscribers Utility

The `unsubscribe` utility removes users from mailing lists. This command can be used in three ways: The first removes a particular subscriber from one or more mailing lists. The second removes multiple subscribers from one or more mailing lists, with the users entered from standard input (indicated by specifying "`-i -`"). The third option is similar to the second, but takes subscriber addresses from a file (indicated by specifying "`-i` *`filename`*") instead of from standard input.

For all three uses of this command, you can either specify the ULID of a particular list, or use the `-a` flag to remove the specified user(s) from all mailing lists.

As with the `subscribe` command, the unsubscribe command includes a `-f` option to circumvent the unsubscription policies of the selected list(s). If this flag is omitted, the unsubscribe utility will simply make unsubscription requests, which are subject to verification and list owner moderation.

Yet another option is the `-q` flag, which suppresses the sending of farewell messages. By default, subscribers removed from a mailing list with the unsubscribe utility receive the list's farewell message, just as if they had unsubscribed themselves from the mailing list. When the `-q` is included, users who are removed from the selected mailing list(s) do not receive the farewell message.

The typical usage of the unsubscribe utility is the removal of a particular user from all mailing lists (either because the account has been discontinued, or because the user has violated some policy). In this case, verification, moderation, and the farewell message are not desirable, so you would execute unsubscribe with the `-f`, `-q`, and `-a` flags (as shown in the first example below).

Usage

```
unsubscribe [-f] [-q] -a|ULID  address
unsubscribe [-f] [-q] -i filename  -a|ULID
unsubscribe [-f] [-q] -i -  -a|ULID
```

Note that any execution of this utility requires either the `-a` flag or the ULID of an existing mailing list.

Example

```
unsubscribe -f -q -a bugs.meany@software.com
```

Immediately removes the user `bugs.meany@software.com` from all Post.Office mailing lists, and does not send farewell messages to this user.

```
unsubscribe -f -q -a -i -
```

Immediately removes the users entered from standard input from all Post.Office mailing lists, and does not send farewell messages to these users.

```
unsubscribe -f baseball_team -i cutlist.txt
```

Immediately removes from the mailing list `baseball_team` the users whose addresses are listed in the file `cutlist.txt`. These users receive a farewell message to inform them that they have been removed from the list.

11.5 postmail (NT only)

Postmail is a “command line mailer” program that allows you to send mail to any SMTP host. It was written to support common sendmail/mail functionality while extending that functionality for the Windows NT environment.

.The most common uses of postmail include:

- Inclusion in batch (`.BAT`) files to notify users of errors or results
- Inclusion in CGI scripts integrated in web servers
- Inclusion in mailing scripts

11.5.1 Using postmail

To use postmail to send a message, you should first create a text file that contains the complete message (headers[55] plus body). This filename is then specified as a command parameter, along with the hostname of the mail server system that will receive the message.

Usage

```
postmail [-t] -H mailhost [address] [-S] "subject" [< messagefile]
```

The optional `-t` flag specifies that the destination address(es) of the message is/are specified in the **To:** header of the message file. If this flag is not used, the address of each recipient should be specified as a command parameter. Multiple addresses can be specified between the -`mailhost` and `messagefile` parameters, with only commas or spaces between them. The optional `-S` can be used to specify the subject of the message.

When used with no command-line arguments, postmail displays help information on its usage.

Examples

The following are examples of simple batch files that invoke postmail. In these examples, Post.Office is installed on the host machine identified as `sparky.software.com`.

```
echo To: joey@software.com, tommy@software.com > tstmsg.txt
echo From: billy@software.com >> tstmsg.txt
echo Subject: Example 1 >> tstmsg.txt
echo "" >> tstmsg.txt
echo This is an example message for postmail >> tstmsg.txt
postmail -t -Hsparky.software.com < tstmsg.txt
```

Figure 11-1: Sample postmail batch file that uses the -t flag.

Notice that the first four lines of this batch file build the message file that will be sent by postmail; lines 1-3 generate the relevant headers, while line 4 is the actual body of the message. In this example, the `-t message` flag causes postmail to look at the **To:** line for the destination address. In this case, the message is sent to `joey@software.com` and `tommy@software.com`.

```
echo To: joey@software.com, tommy@software.com > tstmsg.txt
echo From: billy@software.com >> tstmsg.txt
echo Subject: Example 2 >> tstmsg.txt
echo This is an example message for postmail >> tstmsg.txt
postmail -Hsparky.software.com bobby@software.com < tstmsg.txt
```

Figure 11-2: Sample postmail batch file that specifies destination address as a command-line argument.

55 The headers of the message must be specified in the message file according to the relevant RFCs. The postmail examples illustrate the correct syntax for these headers.

In this example, the mail is sent only to `bobby@software.com`, even though the **To:** header specifies `joey` and `tommy`. Why? Because the `-t` flag is not included, so the actual destination address (that is, the address on the *envelope*) is taken only from the address specified on the command-line. This may seem strange, but it's perfectly legal – after all, you can write an ink-on-paper letter to Joey and Tommy, but stick it in an envelope that has Bobby's address written on it; Bobby gets the letter, and Joey and Tommy get nothing.

```
echo To: joey@software.com, tommy@software.com > tstmsg.txt
echo From: billy@software.com >> tstmsg.txt
echo Subject: Example 2 >> tstmsg.txt
echo This is an example message for postmail >> tstmsg.txt
postmail -t -Hsparky.software.com bobby@software.com < tstmsg.txt
```

Figure 11-3: Sample postmail batch file that mixes the -t option and command-line address

Because this batch file includes both the `-t` flag, the addresses specified in the **To:** header (`joey` and `tommy` again) are treated as the actual recipients of this message. This option does not override the address(es) specified as command-line arguments to postmail, so in this example, `bobby` also receives the message.

Although all of the previous examples used batch files to create and send a message, postmail can be used to write messages interactively from the NT command prompt. The following example illustrates this technique.

```
C:\>postmail -Hsparky.software.com -t
To: support@software.com
From: John.Doe@yourcompany.com
Subject: comment on postmail 1.0

I think postmail is dy-no-mite!

-jdoe
.
C:\>
```

Figure 11-4: Example of sending mail interactively using postmail

When invoked in this manner, postmail reads the message from standard input (the terminal) rather than a message file. After invoking postmail, the user in this example first enters the headers of the message, followed by the body of the message. To complete the message, enter a period on an otherwise blank line, and press Return. This signals the postmail program that it can now mail the message. As always, you should follow standard mail protocol by having at minimum a **To:** and **From:** header, followed by at least one space before the recipient or sender's address. You must separate the header information from the body with a blank line, as required by RFC 822.

11.5.2 Common Problems

This section describes some of the common problems that you may experience with postmail.

DLL Error

On Windows NT 3.51, certain system library files (DLLs) required for execution of postmail may not be available. If you are running NT 3.51, check to make sure that the latest Service Pack (software update) has been installed. This service pack is available from Microsoft's ftp site.

Information on the latest Service Packs can be found on the Service Pack Awareness page of the Technical Support area of the Software.com web site (`http://www.software.com`).

Perl Script Pipe Problem

Perl for Windows NT does not send an end-of-file (EOF) character when the pipe to a program is closed. Since postmail reads input lines until EOF, it will experience problems under Windows NT when used with Perl scripts (Perl for UNIX does not have this problem). To work around the problem, we suggest that you create a temporary message file with the Perl script, and then direct that file into postmail.

11.6 sendmail (UNIX only)

On UNIX platforms, Post.Office includes a utility that replaces the standard UNIX sendmail utility. This Post.Office utility, also named sendmail, actually has fairly limited use since practically all of the functions of the standard UNIX sendmail are performed by Post.Office. However, this additional Post.Office sendmail utility is needed for compatibility with many mail programs that employ the standard sendmail utility to deliver their mail, rather than using SMTP. It can also be used to start up the Post.Office mail system, and check and deliver the mail queue.

Note: *Sorry, we know it can be confusing to talk about two different UNIX programs which are both named sendmail. For clarity, we'll refer to the regular ol' sendmail as the standard sendmail, with our own version referred to as the Post.Office sendmail utility, or the sendmail replacement.*

Probably the most important reason to have the Post.Office replacement sendmail program is to maintain compatibility with existing software that delivers mail using the sendmail command. This software runs the sendmail command and feeds it the message to be delivered. It is then left up to sendmail to deliver the message to all the recipients.

Some examples of commands that work for sending mail are:

```
/usr/lib/sendmail -t < /tmp/message
cat file1 | /usr/lib/sendmail -oem recip1,recip2
```

For a complete list of command-line switches and options related to sending mail, see the reference section that follows (11.6.4).

11.6.1 Starting Post.Office with sendmail

Since the standard sendmail comes installed on most UNIX-based machines, many scripts such as system boot scripts exist to start up sendmail. This is done with a command like:

```
/usr/lib/sendmail -bd -q30m
```

The sendmail replacement provided with Post.Office recognizes this and starts up Post.Office if it is not already running. The -q30m switch is ignored in this command since queue intervals are set up in the system configuration of Post.Office.

11.6.2 Checking the Mail Queue

Many system administrators are used to typing `'mailq'` to check for queued messages. The sendmail replacement provided with Post.Office will respond to this command with the contents of the mail queue. However, most users prefer to use the Post.Office List of Queued Mail Forms (both web and e-mail), which makes processing the queue a little easier. We recommend that you use these forms (as described in Chapter 8) when dealing with queued mail.

11.6.3 Other Modes

The standard sendmail program has a few other operating modes which are not necessary or are not supported in Post.Office. For a complete list of supported operating modes, command-line switches and options, see the reference section below.

11.6.4 Reference Guides

This reference section lists all the available command-line arguments that the Post.Office sendmail replacement program recognizes, along with the behavior that can be expected when they are used. Certain options are recognized (via the `-o` command-line switch) and their effects are noted in a separate table.

Alternate Names for sendmail

The standard sendmail program can be run under several names as a shorthand way to specify the action to perform. The sendmail replacement program recognizes several alternate names. The behavior that results from invoking the sendmail replacement with one of the alternate names is summarized in the following table.

Name	Default Behavior
`sendmail`	Send a single mail message
`newaliases`	Prints an error message since the aliases file is not used
`mailq`	Report the contents of the mail queue
`smtpd`	Run the Post.Office daemon
`bsmtp`	Prints an error message since batch SMTP is not supported

It is important to note that the behavior listed in the above table is the behavior that will result if no other behavior is specified using a command-line option such as `-b` or `-I`.

sendmail Command Line Switches

Command-line switches are processed using getopt(3) as in V8 sendmail. All of the switches supported by V8 sendmail, IDA sendmail and other versions of the standard sendmail are recognized, and the extent of support for these switches is noted in the following table.

Switch	Impact on Behavior
`-B`	If set to 7bit, the high bit is stripped from every byte of the input message
`-b`	Changes the mode of operation. The following modes are supported: `-bd` Start the Post.Office mail system `-bm` Send a single mail message `-bp` Show the status of the mail queue These modes are recognized but *not* supported: `-ba` Use Arpanet protocols `-bb` Do batch SMTP on standard input `-bi` Initialize the aliases database `-bs` Do SMTP on standard input `-bt` Go into address testing mode `-bz` Freeze the configuration
`-C`	None. There is no configuration file, so this switch is ignored.
`-c`	None. This switch is obsolete.
`-d`	None. This switch is ignored since there is no debug mode.
`-e`	Sets the error reporting mode (see option ‘e’ below).

Switch	Impact on Behavior
`-F`	Sets the full name of the sender. If the user running sendmail is not either root, daemon, uucp, smtp, mail, or sendmail, then a header is added to the message indicating the actual sender.
`-f`	Sets the e-mail address of the sender. The same precaution is taken as in the -F switch above.
`-h`	None. The hop count is determined by counting the number of Received headers in the message.
`-I`	Runs as if invoked as 'newaliases' which just prints an informational message.
`-i`	None. This is the default behavior. If sendmail is run interactively, a single '.' will end the message. If it is run non-interactively, e.g. via a pipe to standard input, then the end-of-file condition determines the end of the message.
`-M`	The entire queue is processed regardless of the specified Message ID.
`-m`	None. This is the default behavior. The sender is never removed from the list of recipients if it is listed as a recipient.
`-n`	None. This switch is not supported.
`-o`	Sets an option. See the next section for a list of supported options.
`-p`	None. This switch is not supported.
`-q`	The deferred message queue is processed. If a time interval is given (as in '`sendmail -bd -q30m`') then this switch is ignored. If specified as `-qR`, `-qS`, or `-qI` (as in V8 sendmail), then the behavior is the same as `-R`, `-S`, or `-M` respectively.
`-R`	Attempts to process the queue for hosts matching the pattern provided (e.g. `sendmail -Rabc` will start delivery of queued messages for all hosts containing the string 'abc').
`-r`	Same as `-f` switch above.
`-S`	The entire queue is processed regardless of the specified sender.
`-s`	None. This switch is obsolete.
`-T`	None. This switch is obsolete.
`-t`	Recipients are gathered from both the command line and the message header and the message is delivered.
`-v`	Output is more verbose when sending mail.
`-x`	None. This is an illegal switch which is only recognized to prevent printing an error message.
`-Z`	None. There is no frozen configuration file or even a regular configuration file for that matter.

sendmail Options

The sendmail replacement provided with Post.Office does not need a configuration file (`sendmail.cf`), yet most of sendmail's options can be set from the command line. Many of the options are meant for the sendmail daemon, but some of them are relevant to the normal operation of sending mail.

All of the options supported by V8 sendmail are recognized and the extent of the support for these options is shown below. Note that the options below only refer to the replacement sendmail program, not to Post.Office as a whole. Many of the options not supported by the sendmail replacement are supported by Post.Office in one way or another. Refer to the relevant sections of the manual to determine how to set parameters within Post.Office.

Option	Impact on Behavior
`7`	If set, the high bit is stripped from every byte of the input message. Also see the `-B` command-line switch
`B`	This is always set to '.' and can not be changed.
`d`	Currently none. Since messages are always posted to the local SMTP server, the turn-around time is fairly quick so the 'i' or interactive mode is always used. However, support for other delivery modes may be added in the future.
`e`	Changes the error reporting mode. Valid modes are 'e', 'm', 'p', 'q', and 'w'. The behavior for each mode is the same as sendmail. However, if the local SMTP server is unavailable for some reason and mode 'm' is chosen, the error message will not be deliverable either. In this case, the message is saved in the sender's ~/dead.letter file.
`f`	None. When a "From " line is received, it is changed to "X-UNIX-From:" to be RFC822 compliant.
`I`	None. See the `-i` command-line switch for details.
`o`	None. This is the default behavior and can not be disabled.
`v`	Turns on verbose output. Also see the `-v` command-line switch.
`others`	No other options have any effect. All other options, even invalid ones, are silently ignored.

A

Appendix A: Post.Office Architecture

This appendix contains a comprehensive discussion of the conceptual architecture employed in the design of the Post.Office software. Most people don't need or have much interest in this information. It's dry and unbearably detailed, even according to the unbelievably stultifying standards of the manual-aficionado community.

Unless you are the type of person who looks under the hood before renting a car, asks for a list of ingredients when dining at a restaurant, or wishes you knew how your favorite word processing program was designed rather than just how to use it, you don't need to read this. If you are among the few people who need to assess the software architecture to alleviate security concerns or are just so plain curious you read the dictionary for fun, this one's for you!

As mentioned briefly in Chapter 2, Post.Office functions are distributed among a number of software components which work together to carry out message handling and other activities. Briefly they are:

- A Dispatcher, the daemon or service[56] component of Post.Office which coordinates the activities of all other modules (the Dispatcher is not shown in Figure A-1).
- Account and module configuration databases that contain mail account data, mailing list data, and general configuration information for the other Post.Office modules.
- A message transport agent (MTA) which handles all tasks related to the acceptance, routing, and delivery of mail.
- Post.Office managers, the computerized versions of middle management, which facilitate remote configuration and operation of the system, and make sure that the other modules are doing their job and following the rules.
- A finger server which allows Post.Office users to make directory information about themselves available to the public in response to finger inquires.
- A password server which allows the Eudora mail client to communicate with Post.Office for the purpose of updating a user's POP3 mail account password.

56 A daemon or service is a program that is always running. Except for the Dispatcher, Post.Office modules run only while they are carrying out a task.

Figure A-1 (below) shows Post.Office broken down into its major functional pieces. These pieces run under the supervision of the Dispatcher (not shown).

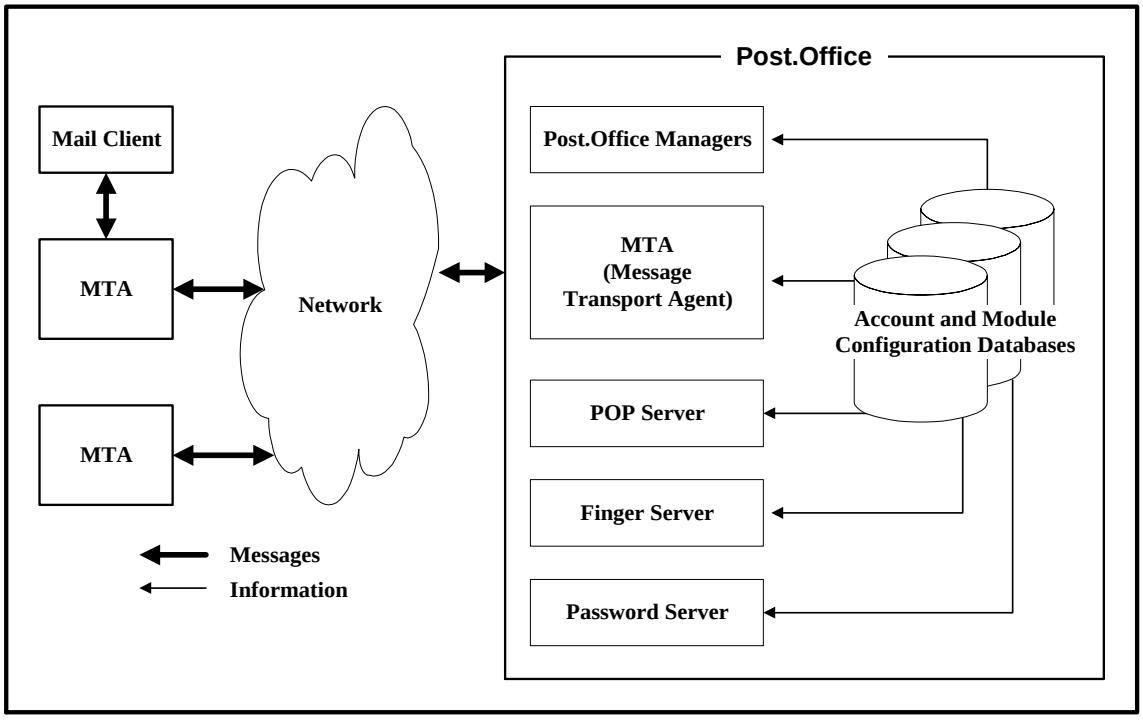

Figure A-1: A bird's eye view: the primary components of Post.Office running under the control of the Dispatcher.

We will refer back to this top-level architecture several times as we fill out the details of the lower-level components and reveal the deep, dark truths about Post.Office.

A.1 The Dispatcher

The Dispatcher's job is simple; it monitors all network ports[57] related to e-mail and launches the appropriate modules to handle incoming connections.

The Dispatcher also controls the number of simultaneous processes which can be operating, in order to limit the amount of computer resources used to process e-mail. For example, when the Dispatcher notices an incoming e-mail message on port 25, it starts up the MTA, which takes the message in and decides what to do with it. In terms of Figure A-1, the Dispatcher can be thought of as an all-encompassing envelope containing and initiating the processes of all items shown within the box labeled "Post.Office."

57 All e-mail transactions begin when one computer contacts another. Different ports are assigned to different types of transactions, so that when a certain port is contacted, the dispatcher immediately knows what kind of transaction is involved and activates the appropriate module.

A.2 Account and Module Configuration Databases

Every Post.Office module has a database containing the configuration information for that module. In addition, a separate database exists for the storage of account and mailing list information.

The database for each module is fairly small, containing a few configuration options which allow customization of the module's operation, and a list of error messages.

The account database, however, holds all mail account and mailing list information, so it can be quite large. All modules refer to the account database whenever they need information in order to process a message or otherwise carry out a task, and since all user information is stored in one place, a single configuration change to the accounts database provides updated information to all modules at once.

A.3 The Message Transport Agent

Post.Office is first and foremost a message transport agent, or MTA. Since the MTA represents the primary functionality of the entire Post.Office system, we will take a look at its role in much greater detail.

An MTA can be used to coordinate message transfer between a small number of computers on a local network, or to orchestrate the transfer of thousands of messages beyond the local network to the millions of on-line users around the world hooked up to the Internet.

Distinct tasks are assigned to the various components of the MTA:

- Messages leaving the local host computer are sent to other MTAs through the SMTP message channel which formats messages according to the open-standard employed on the Internet (the Simple Mail Transfer Protocol).
- Local mail and incoming messages for people with local accounts are delivered by the local delivery channel.
- The POP Server provides an interface to the local user's mail client, thus enabling them to retrieve mail from the user's Post.Office POP3 mailbox.
- The handlers are the workhorses of the MTA. They execute the tasks required to route messages through the system. Usually, incoming messages are sent to the Account Handler, which decides to whom the message should be delivered. If there is some insurmountable problem (such as a bad address), the Account Handler may route the message to the Error Handler. The job of providing automatic responses goes to the Auto-Reply Handler.
- The List Exploder and List Scheduler are responsible for the distribution of mailing list messages. They forward mailing list messages to the appropriate subscribers and send statistics reports to the list owners.

These basic components of the Post.Office MTA are depicted in Figure A-2 and will be discussed in greater detail in the sections that follow.

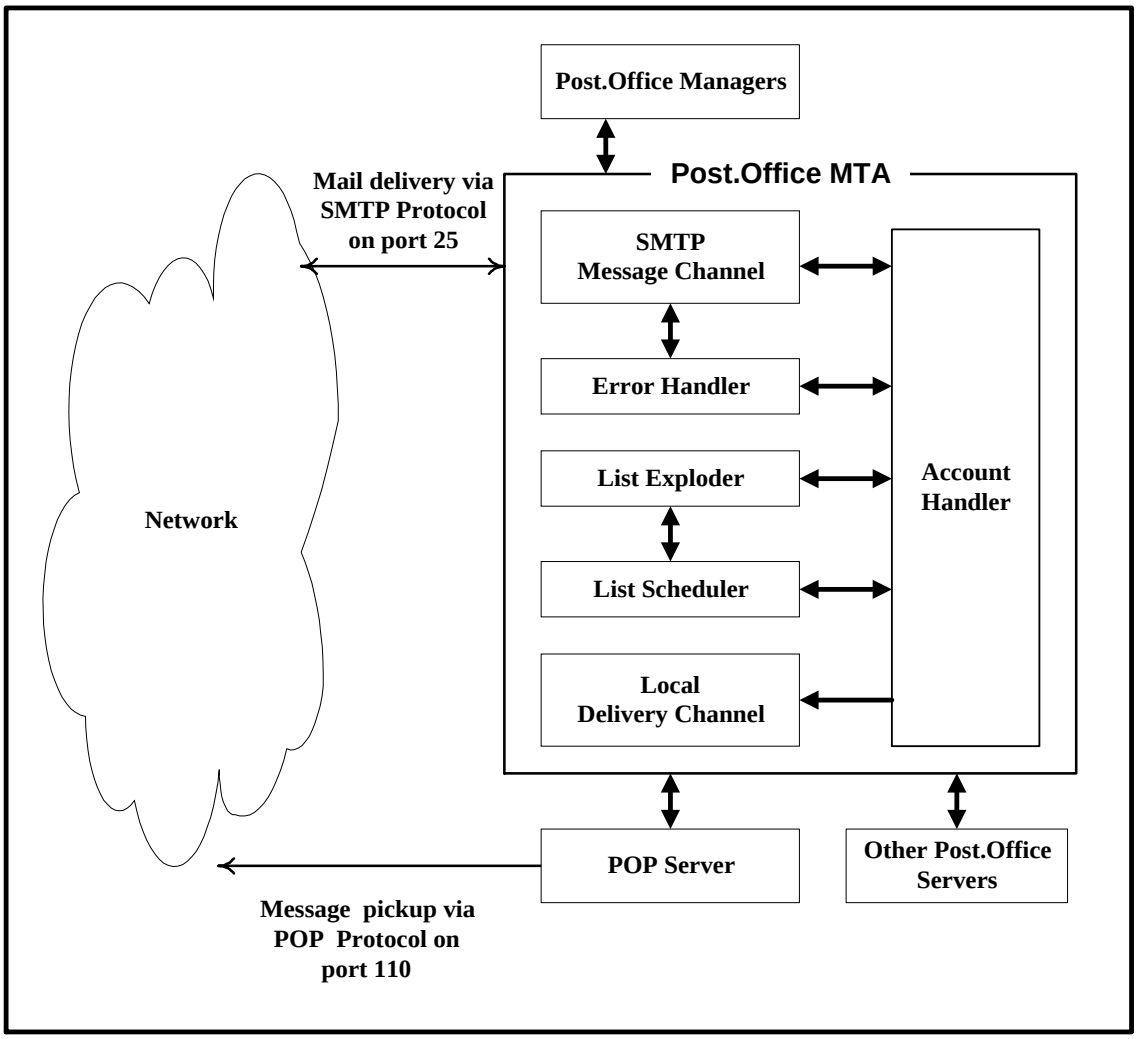

Figure A-2: The components of the Post.Office Message Transport Agent.

A.3.1 The SMTP Message Channel

The Post.Office SMTP channel exchanges messages with other mail servers and accepts new messages from mail clients. It can relay messages to and from other modules of Post.Office and, in some cases, accept and dispatch messages without consulting other modules.[58] One of its principal tasks is to accept messages addressed to local recipients (users with accounts on the server running Post.Office) and forward them to the local delivery channel.

58 If a channel alias is set up (see Chapter 4) an incoming message can be immediately forwarded to another MTA without being handled by any other modules of Post.Office.

Components of the SMTP Message Channel

The SMTP channel is a group of Post.Office modules which transfer messages using the Simple Mail Transfer Protocol. It consists of an agent that accepts mail (SMTP-Accept), one that routes mail (SMTP-Router), and one that delivers mail (SMTP-Deliver).

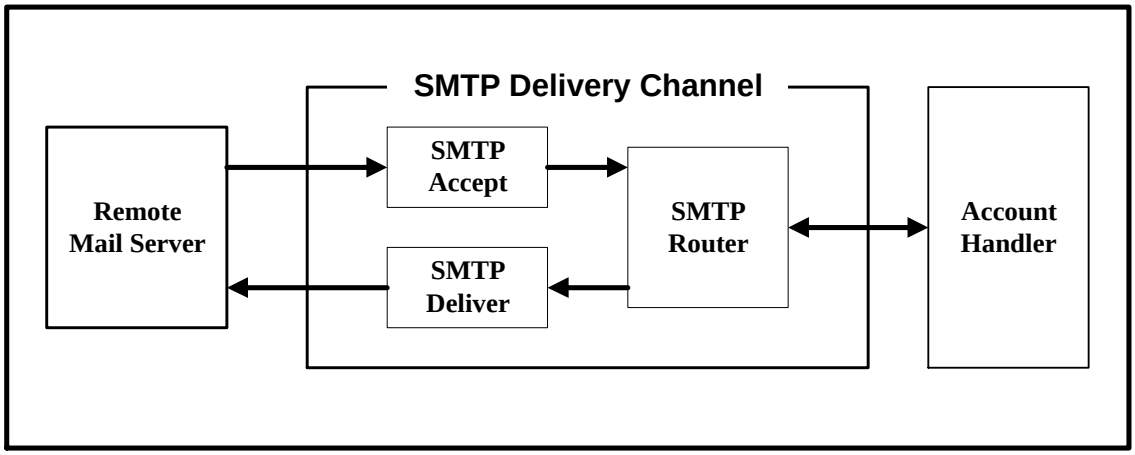

Figure A-3: The innards of the SMTP Message Channel.

SMTP-Accept is the module which accepts incoming messages on port 25 as detected by the Post.Office Dispatcher. SMTP-Accept can be configured to accept messages only for recipients that it recognizes, or you can allow the server to accept messages and forward them to other mail servers[59]. Once a message comes in, it is immediately transferred to the SMTP routing agent.

Note: *Messages may bypass the SMTP-Accept module and enter the message channel at the SMTP-Router (see description that follows) if they originated in another module of Post.Office (such as the Auto-Reply handler).*

SMTP-Router, the SMTP routing agent, examines message headers and, based on the instructions they contain, determines how the mail should be handled. If the message is to be relayed to another MTA, the router forwards it to SMTP-Deliver. If it should be delivered to a local user, the router hands off to the Account Handler. And if there is some confusion as to where the message should go, the router passes it to the Error Handler.

SMTP-Deliver sends messages to other remote MTAs which speak the same protocol (SMTP). This module takes a message from SMTP-Router, contacts the remote MTA responsible for mail addressed to the recipient of the message, and delivers the message to that MTA. Part of this module's task is determining the remote MTA to which the message should be sent.

59 In some cases it is worthwhile to accept messages which are not addressed to local recipients. This is discussed in the Operations Chapters.

Mail Flow Through the SMTP Channel

In some cases, a message is transferred solely by the SMTP channel. This is true for mail addressed to users with mail accounts somewhere out on the Internet or on another machine on your local network; those messages are accepted, routed, and delivered (to another MTA) through the SMTP channel only. However, if a message is addressed to a local user (someone with a mail account on your Post.Office server), the process is different. In such cases, the message is accepted by SMTP-Accept, passed through SMTP-Router to the Account Handler, and then forwarded to the local delivery channel for delivery to the user on this particular Post.Office host (see next section for more information on the local delivery channel).

A.3.2 The Local Delivery Channel

When Post.Office accepts a message that it recognizes as being addressed to a local recipient (that is, someone with an e-mail account on this server[60]), the message is routed to the local delivery channel. The local delivery channel carries out the final delivery of all local e-mail messages.

The local delivery channel consists of those Post.Office modules that transfer messages to mail clients via POP3 delivery, UNIX delivery, and/or program delivery. The path a message takes to reach a particular recipient is determined by the delivery method(s) recorded in the Account Data Form for that user (see Chapter 5 for details).

POP3 Delivery

Messages destined for POP3 delivery are passed to the Mailbox Deliver module. That module takes the incoming mail and files it in the local user's POP3 mailbox directory. The mail can then be retrieved by any POP3-compatible mail client running on a remote computer.

To access his POP3 mailbox and retrieve mail, a user need only have the required client software and a means of establishing temporary contact with the host machine (the computer on which Post.Office is running). Thus any computer that can connect to the host machine (generally via a network such as an Ethernet or the Internet but a modem will do the trick in a pinch[61]) can be part of the e-mail system that Post.Office coordinates.

Note: *If you currently run an Internet Message Access Protocol or IMAP server (an enhanced protocol similar to POP3), you should continue to retrieve your messages from your UNIX maildrop file.*

60 Technically, Post.Office recognizes an address when it matches an entry in the account database.

61 Modems are slow, so it is always preferable to do things through a network. Slow computers are worse than fingernails on a chalkboard, which is why the eighties were so painful.

UNIX Delivery

UNIX delivery is available on UNIX machines only. It operates in a manner similar to POP3 delivery, but in this case UNIX Deliver is the responsible module, and the messages are deposited to the user's UNIX mail drop file. The user's mail client looks to this file for any new messages.

UNIX delivery is only available to people who have a system account on the Post.Office host (in addition to their Post.Office mail account). When Post.Office is installed on a system running UNIX, all existing users on that machine are automatically given Post.Office e-mail accounts and are initially assigned UNIX delivery.

Program Delivery

Program delivery allows users to have their messages delivered to a program. The selected program can perform any operation of the user's choosing provided it meets with the approval of you, the Postmaster.

When a user's mail account specifies program delivery, all messages addressed to that account are passed to the Program Deliver module. The Program Deliver module takes the incoming mail, locates and launches the target program, hands off the message, and waits patiently for notice that program processing is complete.

For the purpose of understanding system architecture, this discussion is sufficient. In truth, establishing program delivery is a bit more complicated. First, it requires that the user have a system account. Second, it requires that the target program be stored in a particular Post.Office directory. Finally, there may be issues with passwords and file permissions. For an exhaustive explanation of the program delivery feature, please refer to Chapter 6.

The Local Delivery Channel In Summary

The three components of the local delivery channel are depicted in Figure A-4.

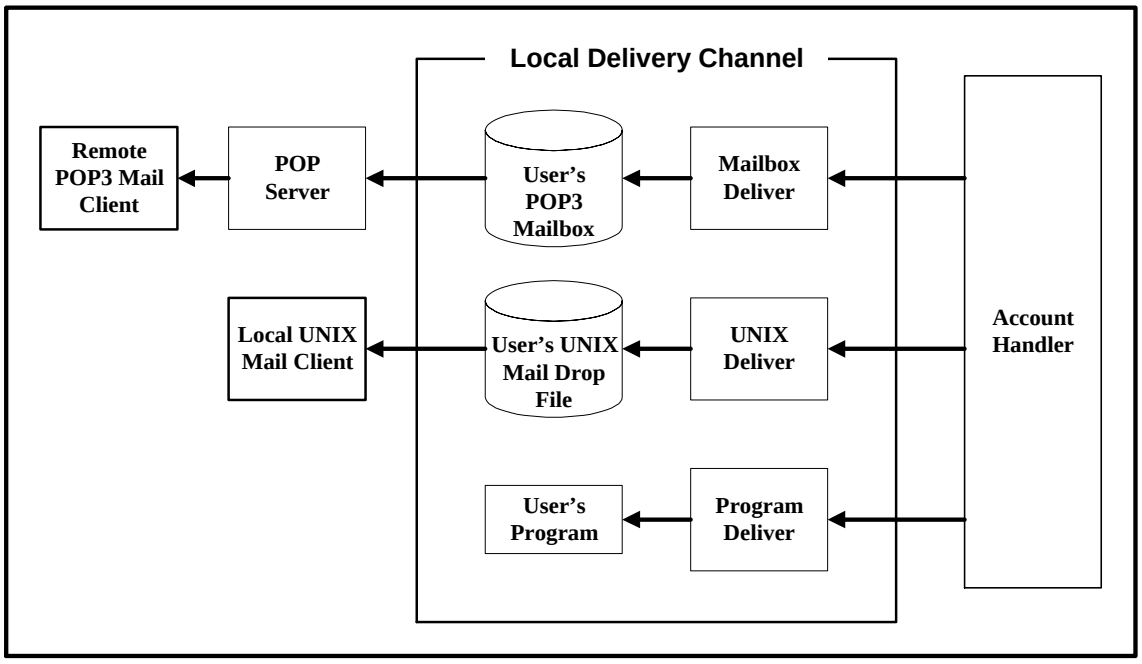

Figure A-4: The Local Delivery Channel -- POP3, UNIX and Program delivery.

A.3.3 The MTA Handlers

Handlers are the brains in Post.Office. They decide what to do with messages and supervise the transport of e-mail between modules. The Account Handler bears primary responsibility for the routing of mail through your Post.Office mail server. It handles all standard transactions and forwards strange messages that it can't understand to the Error Handler. The Error Handler is responsible for the resolution of problem messages. You, as Postmaster, communicate directly with the Error Handler when responding to Message Action Forms (see Chapter 8 for details).

The Auto-Reply Handler is responsible for providing an automatic e-mail response for those accounts which have the Auto-Reply feature selected. It formulates the outgoing messages and forwards them for delivery.

The Account Handler

Figure A-2 hints at the central role the Account Handler plays in coordinating message transport. It routes messages between channels and between the various Post.Office modules, making its decisions based on information located in the accounts and configuration databases.

Like all computer programs, the Account Handler is limited by its startling insensitivity, complete lack of intuition, and implacable lack of creativity. Since humans sometimes make mistakes in addressing and sending e-mail, occasionally messages are consigned to the Error Handler, most often due to a faulty address.

The Error Handler

Anytime Post.Office receives a message it does not understand, it gives it to the Error Handler. In some cases error messages are simply returned to the sender. In others, they are held until the Postmaster decides what to do with them.

If the offending message is returned, the Postmaster is advised via e-mail that an error occurred. Such notices are provided for information only, no further action is necessary.

Messages with error conditions that require Postmaster intervention are held pending resolution. An entry is posted to the Error Message Handler web form advising the Postmaster of the problem. In addition, the Error Handler can advise the Postmaster via e-mail.

The Error Handler provides the Postmaster with several options for resolving the error. The Postmaster must then decide which option to select. Submission of a correction resolves the error and removes the entry from the Error Message Handler web form.

Note: *Details of what types of errors can occur and the Postmaster's role in resolving them are discussed in the Chapter 8.*

The Auto-Reply Handler

The Auto-Reply Handler allows Post.Office to automatically reply to messages received by a local account. When an incoming message is processed by the Account Handler, a check is made to determine if an automatic response should be invoked. If so, the Auto-Reply module receives the intended content of the message (which is specific for each account), formats an outbound message in response to the one received, and passes this reply back to the Post.Office MTA for delivery.

There are three Auto-Reply operating modes. They are:

- **Vacation.** The first message sent to an account receives an automatic response. Subsequent messages from the same address are ignored.
- **Reply.** An automatic response is returned to every incoming message.
- **Echo**. For every incoming message, an automatic response is generated. The response contains the account-specific reply message *and* the text of the original incoming mail.

A.4 The List Exploder and List Scheduler

Messages submitted to a mailing list for posting are handled by the List Exploder and List Scheduler modules. The List Exploder assumes primary responsibility for all such mail and hands off to the List Scheduler as required.

The List Exploder

The Account Handler directs mail addressed to a mailing list address to the List Exploder. The List Exploder obtains mailing list data from the accounts database, expands (explodes) list traffic to the subscribed user addresses, and rewrites the messages so that they contain the appropriate list information (prologue text, epilogue text, etc.) List messages slated for immediate delivery are sent back to the Account Handler. Messages to be held for a regularly scheduled digest delivery are forwarded to the List Scheduler.

The List Scheduler

Messages received by the List Scheduler are held pending digest delivery. At the scheduled time, all mail addressed to a particular list is combined into a single message and passed off to the Account Handler for delivery.

Mailing List statistics are generated by the List Scheduler and sent to the appropriate list owner once a day. Additional statistics can be written to the Post.Office log files if exact tracking of digest delivery is required.

A.5 Post.Office Managers

Post.Office managers are responsible for the maintenance of the Post.Office accounts and configuration databases. The databases are maintained via entries made in web or e-mail forms. The form used and the type of data being manipulated determine which of the Post.Office managers will be called.

E-mail forms are issued and processed by three modules: the Configuration Manager, the Account Manager, and the List Manager. The Configuration Manager supervises configuration for all Post.Office modules via interaction with the configuration database. The Account Manager oversees the addition, deletion, and modification of mail account data stored in the account database. The List Manager also interacts with the account database, but it handles mailing list information exclusively.

Database management via web forms is the responsibility of the WWW-Server regardless of the database in which the information resides.

A.5.1. The Configuration Manager

Every module includes a few configuration options which can be set by the Postmaster. A minimal set of essential options are established at time of installation, allowing Post.Office to do its job "straight out of the box." If additional configuration is required, the Postmaster is free to make changes as desired.

Typically, the Postmaster would make such changes via the convenient web form interface. But in the event that web access is unavailable, those same changes can be made via e-mail. When modifying configuration information via e-mail you are interacting with the Configuration Manager. The Configuration Manager provides the required e-mail entry forms, reviews the returned forms for proper permissions (your password), then executes the changes specified.

A.5.2. The Account Manager

Like the Configuration Manager, the Account Manager handles e-mail forms only. Again, this is not the preferred method of database management, but it's quick, it's easy, and it's available if you're interested.

A.5.3. The List Manager

The List Manager also processes e-mail forms, but the requests it handles pertain to mailing lists only. Mailing list requests (such as the `subscribe` command) are passed to the List Manager for processing. Because mailing lists frequently involve large numbers of remote users, e-mail management of mailing list data is more common than e-mail management of account or configuration information. In fact, certain operations, like the request for a list of subscribers, can only be completed via e-mail.

A.5.4. The WWW Server

The WWW Server is roughly equivalent to the Account Manager, Configuration Manager and List Manager combined. It maintains the same database information, but handles the transactions via a Web interface rather than an e-mail interface.

A.6 The POP Server

The POP server component of Post.Office is responsible for handling POP3 e-mail client requests to download messages. The POP3 protocol is the most common method of retrieving messages from a mail server, and allows users to download mail to their local system, where it can be read, replied to, deleted, ignored, etc. This server operates independently of the Post.Office MTA, except for the fact that they refer to a common accounts database.

The POP server receives inquiries directly from the network according to the POP3 protocol. When an e-mail client contacts the POP server, it provides a login name and a password for a particular user. The POP server then checks with the accounts database to confirm that an existing account corresponds to the given login name, and that the given password is correct for this account. If the login information is found to be valid, the POP server then answers subsequent client commands to list and download the messages waiting in the account's Post.Office mailbox.

For maximum security, access to retrieving mail from the POP server can be limited to specific domains or hosts (or eliminated completely) at the discretion of the administrator.

A.7 The Finger Server

Post.Office includes a finger server. Finger is the most common directory service on the Internet, and provides a means of obtaining basic information about someone by using their e-mail address.

The finger server receives inquiries directly from the network according to the finger protocol. Referring to the appropriate information in the account database, it responds directly to the query over the network. This server operates independently of the Post.Office Message Transport Agent, except for the fact that they refer to a common database (the account database). This, coupled with the fact that the finger service can be automated, facilitates easy management and consistency of information, and greatly reduces the workload of the administrator. Additionally, it provides the added security benefit of employing one coordinated set of services (to which access can be limited as desired) rather than a haphazard array of individual servers.

For maximum security, access to the finger service can be limited to specific domains or hosts (or eliminated completely) at the discretion of the administrator. The individual information provided on the finger server can be modified by the users, but this activity is made secure through the use of passwords and finger access domain restrictions.

A.8 The Password Server

The Password Server allows user's to access the mail server and change their Post.Office password via the Eudora mail client. This feature, however, affects the Post.Office password only and will have no apparent effect on access privileges if you have selected to use NT Logon passwords instead of the standard Post.Office passwords.

A.9 Network vs. Local Modules

By now you should have a pretty good understanding of the various Post.Office software modules and how they interact to handle mail, configure the system, respond to finger queries, etc. It is also important to remember that only certain Post.Office modules communicate with the outside world. Those network modules are listed below with the number of the port over which they communicate.

- Finger-Server - port 79
- Password-Server - port 106
- POP3-Server - port 110
- SMTP-Accept - port 25
- WWW-Server - the port designated at time of installation (Consult the Post.Office Licensing/Configuration Information web form for the port number used by your server.)

The remaining modules are local; they operate entirely within the Post.Office mail server and are invisible to the outside world.

The distinction between network and local processes is significant when evaluating load on your mail server. Although the number of concurrent processes allowed is under Postmaster control, we recommend that you maintain low limits to ensure smooth running of your mail server. For additional information on recommended levels, review the section on concurrent network and/or local processes found in Chapter 4.

A.10 The Whole Enchilada

Figure A-5 puts it all together - the complete workings of the Post.Office mail server.

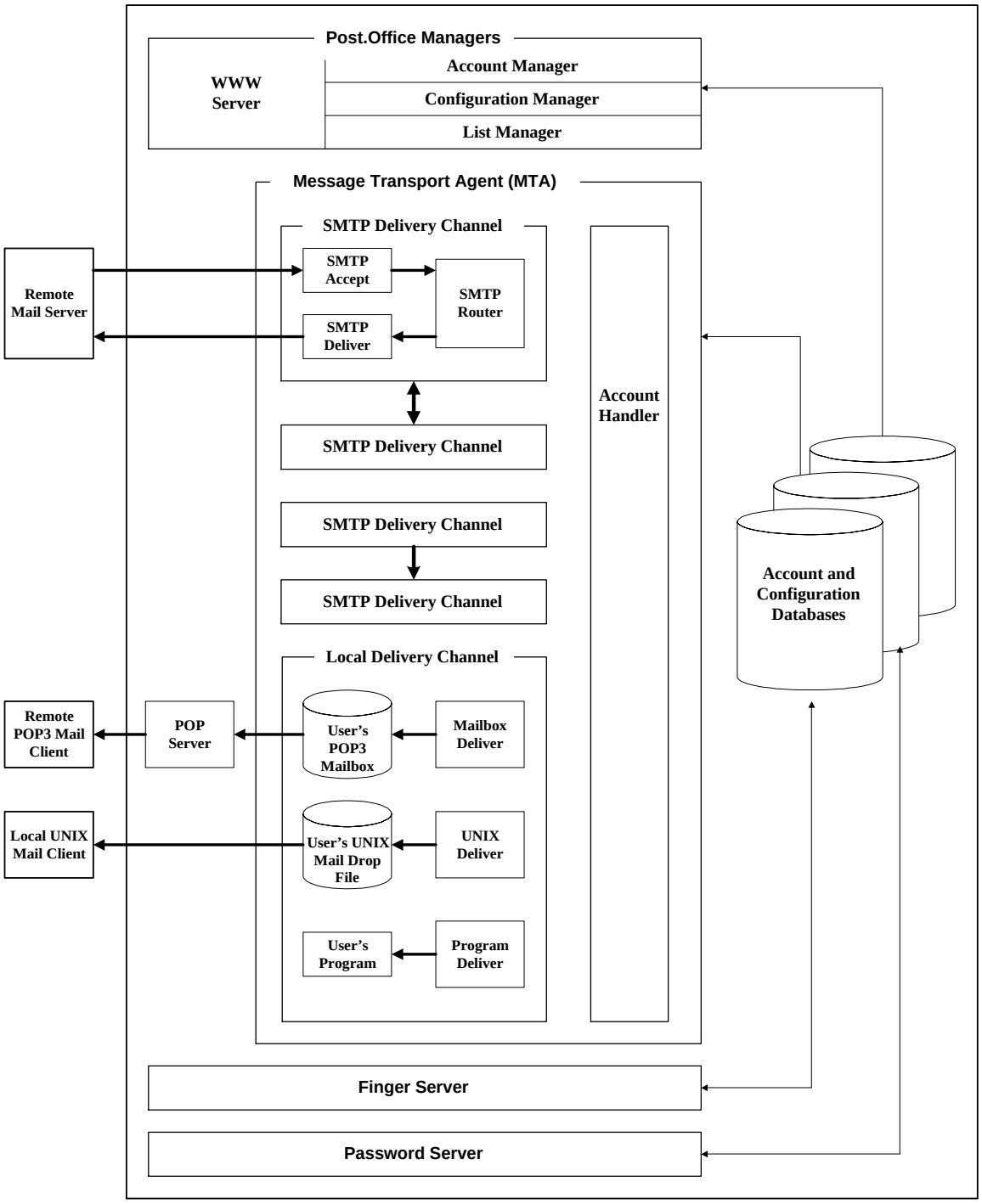

Figure A-5: The whole Post.Office enchilada.

Thank you for your interest in Post.Office.

B

Appendix B: Standards Conformance

The purpose of this appendix is to list the *open standard* protocols that are used by Post.Office. The idea behind open standards is that they are publicly available so anybody can implement them and inter-operate with other conforming software implementations. The Internet is built on the concept of open standards.

Since Post.Office is built on open standards, you are assured that it will work with other software products written to work on the Internet (or other TCP/IP networks) where these standards are used.

The Post.Office mail system conforms to the standards listed in this section. Every effort has been made to verify that Post.Office conforms completely with the specifications to ensure interoperability.

The letters "RFC" in all of the standard names mean *Request for Comments*. These are published protocol standards based on work done by working groups in the Internet Engineering Task Force (IETF).

Mail Transport Protocols

RFC 821 J. Postel, *Simple Mail Transfer Protocol*, August, 1982.

RFC 822 D. Crocker, *Standard for the Format of ARPA Internet Text Messages*, August, 1982.

RFC 974 C. Partridge, *Mail Routing and the Domain Name System*, January, 1986.

RFC 1123 R. Braden, *Requirements for Internet Hosts - Application and Support*, October, 1989.

RFC 1521 N. Borenstein, N. Freed, *MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies*, September, 1993.

RFC 1651 J. Klensin, N. Freed, M. Rose, E. Stefferud, D. Crocker, *SMTP Service Extensions*, July 1994.

RFC 1653 J. Klensin, N. Freed, K. Moore, *SMTP Service Extension for Message Size Declaration*, July 1994.

RFC 1985 J. De Winter, *SMTP Service Extension for Remote Message Queue Starting*, August 1996.

Mail Access Protocols

RFC 1725 J. Myers, M. Rose, *Post Office Protocol - Version 3*, November, 1994.

Directory Services

RFC 1288 D. Zimmerman, *The Finger User Information Protocol*, December, 1991.

C

Appendix C: References

Much of the knowledge and information required to administer a Wide Area Network goes well beyond the scope of simple electronic mail. In fact, even a good deal of information that is specific to e-mail in general is beyond the scope of this Post.Office manual.

Therefore, we have included the following list of references which may be useful to you when delving into these expanded subject areas. Software.com, Inc. has no affiliation with any of the listed references, and therefore cannot ensure or be responsible for any of the information contained therein; nor can we provide a specific endorsement of these products since we have not reviewed them in detail. This is simply a list of reference books that have been useful to us at one time or another.

Entry-Level Information

The Whole Internet User's Guide and Catalog

Krol, Ed. Sebastopol, CA: O'Reilly & Associates, Inc., 1992.

ISBN: 1-56592-025-2

System Administration

The Internet System Handbook

Lynch, Daniel C. and Marshall T. Rose. Reading, Massachusetts: Addison-Wesley Publishing Company, 1993.

ISBN: 0-201-56741-5

The Simple Book

Rose, Marshall T. Englewood Cliffs, New Jersey: P T R Prentice Hall, 1994.

ISBN: 0-13-177254-6

The Domain Name System (DNS)

DNS and BIND

Albitz, Paul and Cricket Liu. Sebastopol, CA: O'Reilly & Associates, Inc., 1992.

ISBN: 0-56592-010-4

Transmission Control Protocol/Internet Protocol (TCP/IP)

TCP/IP Network Administration

Hunt, Craig. Sebastopol, CA: O'Reilly & Associates, Inc., 1992.

ISBN: 0-937175-82-X

General Internet Messaging

The Internet Message

Rose, Marshall

A Directory of Electronic Mail Addressing and Networks

Frey, Donnalyn & Rick Adams. Sebastopol, CA: O'Reilly & Associates, Inc., 1993.

ISBN: 1-56592-031-7

sendmail

sendmail

Costales, Bryan with Eric Allman and Neil Rickert. Sebastopol, CA: O'Reilly & Associates, Inc., 1993.

ISBN: 1-56592-056-2

UNIX System Administration

Essential System Administration

Frisch, AEleen. Sebastopol, CA: O'Reilly & Associates, Inc., 1992.

ISBN: 0-937175-80-3

Security and Firewalls

Firewalls and Internet Security Repelling the Wily Hacker

Cheswick, William R. and Steven M. Bellovin. Reading, Massachusetts: Addison-Wesley Publishing Company, 1994.

ISBN: 0-201-63357-4

Index

