CanIt-PRO Administration Guide

for Version 3.4.6

Roaring Penguin Software Inc.

14 June 2007

Contents

List of Figures

Chapter 1

Introduction

CanIt-PRO is server-based anti-spam software that stops spam from entering your network. This guide explains how to administer CanIt-PRO, and is intended for e-mail administrators. For installation instructions, please see the Installation Guide, and for end-user instructions, see the User's Guide.

1.1 Principles of Operation

CanIt-PRO uses many sophisticated rules and mechanisms to detect spam. These rules are part of an open-source anti-spam package, and are very effective and broad-spectrum. Once CanIt-PRO decides that a message is probably spam, it is held for review by a spam-control officer. CanIt-PRO returns an SMTP "temporary failure" code to the sending relay host. In this way, the message body is held in the *sender's* spool directory and not in yours. A more complete description of how CanIt-PRO operates is given in Chapter 2.

1.2 Handling False-Positives

Although CanIt-PRO's rules for identifying spam are very accurate, we realize that no purely automated process can be 100% correct. That is why CanIt-PRO relies, in the end, on human intervention. In this way, it can guarantee that no legitimate e-mail message will ever be rejected, and you will never lose a customer or prospect e-mail because of automated scanning.

At first glance, it seems that requiring human intervention is a step backwards—spam messages again must be reviewed by a person. In reality, CanIt-PRO still saves time and money for the following reasons:

- CanIt-PRO includes many features to lower the spam-control officer's workload. (These features are described later in this manual.) A spam-control officer can scan and categorize e-mail messages much more quickly than end-users using mail reader software. In addition, only one person has to scan the messages, instead of tens or hundreds.

- As time passes, the spam control officer can recognize mailing-list traffic and other traffic which tends to be falsely flagged as spam, and tell CanIt-PRO to always allow that traffic. Over time, this reduces the amount of human intervention required.

- If you are willing to take the risk of inappropriately rejected messages, you can configure CanIt-PRO to automatically reject very high-scoring messages.

1.2.1 Spam-Control Delegation

CanIt-PRO operates similarly to CanIt, except that it allows delegation of spam-control responsibility. With CanIt-PRO, each end-user can be responsible for his own "virtual CanIt", or you can delegate responsibility by department head, or any mixture. Chapter 2 explains the operation of CanIt-PRO in detail.

Each virtual CanIt is called a *stream*, and incoming messages are *streamed* so that different users' spam decisions do not affect other users.

1.3 Organization of this Manual

This manual is divided as follows:

Chapter 1, "Introduction", is this chapter. You should familiarize yourself with the terms in Section 1.4 before proceeding.

Chapter 2, "Operation", describes the principles behind CanIt-PRO's operation.

Chapter 3, "Streams", describes the concepts behind streaming. You must read and understand this chapter before using CanIt-PRO in production.

Chapter 4, "CanIt-PRO Setup", describes basic setup steps you need to take to configure CanIt-PRO.

Chapter 5, "CanIt-PRO Administration", describes tasks undertaken by the CanIt-PRO administrator.

Chapter 6, "External Authentication", describes how to integrate CanIt-PRO with an external authentication mechanism (such as LDAP or POP3.)

Chapter 7, "Bayesian Filtering", explains CanIt-PRO's Bayesian filtering module. Bayesian filtering uses statistical analysis and training so that CanIt-PRO "learns" to recognize spam based on user feedback.

Chapter 8, "Permissions", describes how to control access to various parts of the CanIt-PRO Web interface.

Chapter 9, "Streams, Inheritance and the Simple GUI", describes how the CanIt-PRO administrator can set up different groups of spam-handling settings and allow end-users to select from one of a limited number of predetermined setups. The simplified interface is very useful if you wish to provide "canned" settings for unsophisticated users.

Chapter 10, "Locked Addresses", describes how CanIt-PRO permits users to generate addresses that they can give out to strangers, but that those strangers cannot in turn give or sell to third-parties.

Chapter 11, "Attachment Handling", describes CanIt-PRO options for handling various attachments.

Chapter 12, "Tips", contains guidelines for reducing the workload of the spam-control officer and dealing with spam more effectively.

Chapter 13, “Security”, contains information about CanIt-PRO security.

Appendix A, “A Testing Topology for CanIt-PRO”, gives tips on how to test CanIt-PRO before putting it into production. This appendix also contains useful information on production network topology, so if you are planning on using CanIt-PRO as a relay-only server, you should read this appendix.

Appendix B, “CanIt-PRO Architecture”, discusses CanIt-PRO’s filter architecture in detail. It provides tips on tuning CanIt-PRO and describes the various configuration files used by CanIt-PRO.

Appendix C, “CanIt-PRO HOWTOs”, gives short “how-to” recipes for performing common CanIt-PRO administrative tasks, such as restoring a database from the text dump, or moving CanIt-PRO to another machine. It also briefly describes the command-line tool `canit-cmd`.

Appendix D, “CanIt-PRO Logging”, explains how CanIt-PRO logs statistics, warning, and error messages.

Appendix E, “Additional Scripts”, describes some additional scripts bundled with CanIt-PRO that you might find useful.

1.4 Definitions

We use many terms related to Internet e-mail in this manual. Here is a definition of some of the terms we use.

SMTP “Simple Mail Transfer Protocol”, as described in Internet RFC 2821. This is the protocol used to transmit e-mail over the Internet.

Relay Host When a mail server wishes to transmit e-mail to your server using SMTP, it establishes a connection with your mail server. The machine attempting to transmit mail to your server is called a **relay host**.

SMTP Dialog During the course of e-mail transmission, the two ends of an SMTP connection transmit commands and results back and forth. This conversation is called the **SMTP dialog**.

Temporary Failure Code Also called **tempfail**, this is a code sent to a relay host telling it that e-mail transmission has failed temporarily, and it should retry in a little while. Typically, the relay host retains the e-mail message in a spool directory and retries transmission periodically. The host eventually gives up after a certain period (typically, a few days) has elapsed without successful transmission.

Permanent Failure Code Also called **reject**, this is a code sent to a relay host telling it that e-mail transmission has failed and will not succeed. (For example, this code is sent if someone tries to send e-mail to a nonexistent user.) The relay host typically e-mails a failure notification to the original sender and discards the message.

Hit-and-Run Detection Also known as **Greylisting**, this technique blocks spam from certain spam-sending software. It works by issuing a Temporary Failure Code the first time an e-mail arrives from an unknown sender and IP address. Legitimate SMTP servers will retry, allowing the message to be delivered. Some spam-sending software does not retry, and messages sent by such software will be blocked without any content-scanning if Hit-and-Run detection is enabled.

Envelope Mail messages often have *headers* specifying the sender (the “From:” header) and recipients (typically the “To:” header.) However, SMTP has a completely separate set of commands for specifying the sender and recipients. The sender and recipients specified in the SMTP commands are referred to as the *envelope sender* and *envelope recipients*, and do not necessarily match the information in the message headers. CanIt-PRO always uses the envelope sender and recipient addresses in its rules.

Sender’s Domain This is the domain part (everything after the `@` sign) in the sender’s e-mail address.

Milter is a Sendmail interface that allows external programs to listen in on the SMTP dialog, and potentially modify Sendmail’s actions and SMTP responses.

MIMEDefang is a free (GPL’d) e-mail scanning program that integrates with Sendmail’s Milter API. It forms the basis for CanIt.

SPF stands for “Sender Policy Framework”. It is a mechanism that allows a domain’s administrator to list which hosts are allowed to originate e-mail claiming to come from that domain. For more detauls, please see `http://www.openspf.org`.

Bayesian Analysis is a method whereby an anti-spam system keeps track of how often words appear in spam and non-spam. Once enough statistics have been accumulated, the system can calculate the likelihood that a new message is spam.

RPTN is the Roaring Penguin Traning Network. This is a system whereby multiple CanIt-PRO installations can share Bayes training data.

CanIt is extra software built on top of MIMEDefang that provides sophisticated spam-management functions.

CanIt-PRO is an enhanced version of CanIt that allows flexible delegation of spam-control responsibilities rather than requiring a single spam-control officer.

Stream is a “virtual CanIt” machine offered by CanIt-PRO. If an incoming e-mail arrives for more than one recipient, and the recipients each wish to have his or her own private spam trap, CanIt-PRO re-mails the original message so each recipient has his or her own copy, and can dispatch it as he or she sees fit.

For every user, a “home stream” is defined. This home stream is normally the same as the user’s login name, but can be changed by the administrator.

Other streams can be created manually to handle mailing lists or group departmental mail into a single stream.

Chapter 2

Operation

2.1 Principles of Operation

CanIt-PRO watches each incoming SMTP message and operates as follows:

- If the SMTP connection is from a blacklisted host, the connection is rejected.
- If the message sender is blacklisted (or the domain is blacklisted), the message is rejected.
- Otherwise, the message is collected and scanned.

After CanIt-PRO has scanned the message, it performs the following operations:

- Dangerous files (for example, files named `*.exe`) are removed from the message (if you have selected this option.)
- If the sender, relay host or domain are whitelisted, the message is accepted without being scanned for spam.
- Many spam-detection rules are applied to the message. If the message is judged not to be spam, it is accepted and the SMTP transaction succeeds.

For messages judged to be spam, CanIt-PRO takes the following steps:

- A unique ID is calculated by running the message body through a special hash function. The hash calculation is designed to be resistant to some forms of trivial message modification.
- The ID is looked up in a database.

  1. If the ID is not found in the database, it is entered as a **one-shot** message, and a temporary failure code is sent to the SMTP sender.
  2. If the ID is in the database with status **one-shot**, the status is changed to **pending** and a temporary failure code is sent to the SMTP sender.

3. If the ID is in the database with status **spam**, a permanent rejection code is sent to the SMTP sender.
4. If the ID is in the database with status **not-spam**, the message is accepted for delivery.

The flow of mail through CanIt-PRO is summarized in Figure 2.1. Note that this is the *conceptual* flow; in reality, several optimizations are performed that would only complicate the figure. See also Figures 2.2 on page 17 and 2.3 on page 19 for more accurate details about blacklisting and whitelisting.

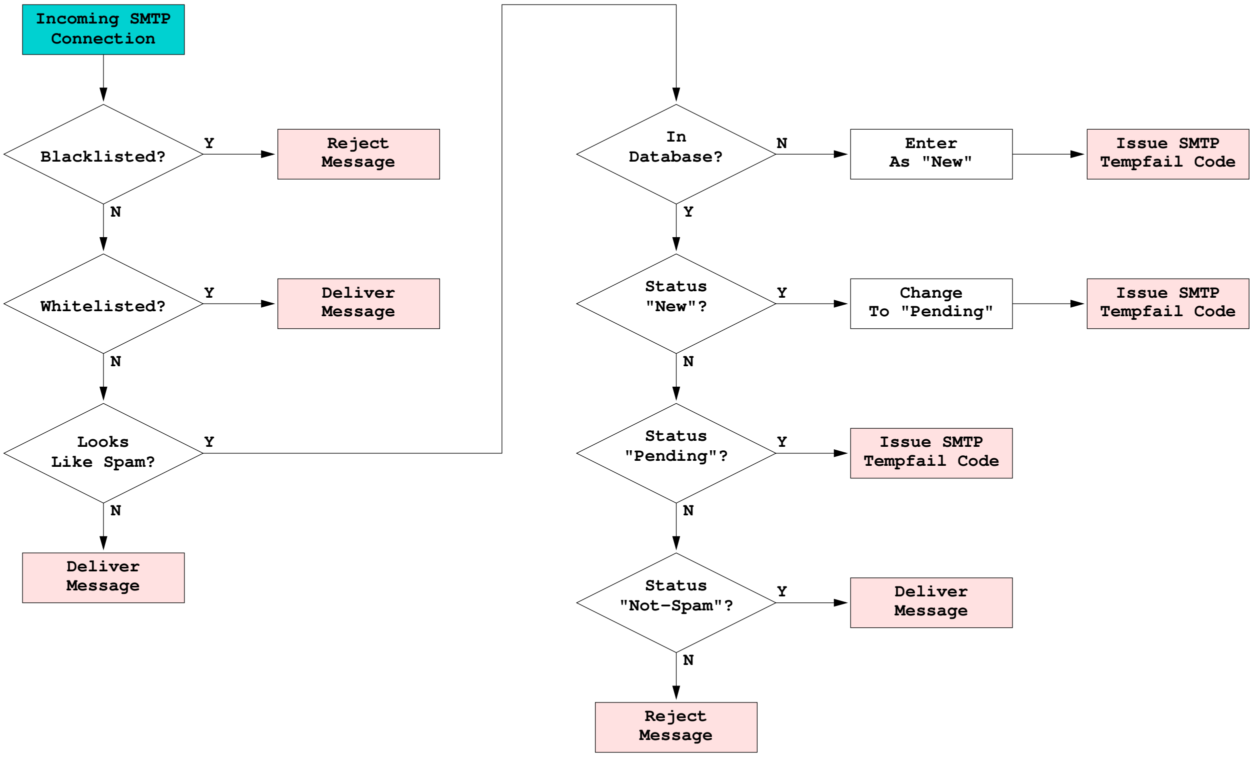


Figure 2.1: Flow of Mail through CanIt-PRO

2.2 Interaction between Whitelists, Blacklists and Mismatch Rules

CanIt-PRO must prioritize whitelists and blacklists. For example, suppose a sender is whitelisted, but the host the message comes from is blacklisted. What should CanIt-PRO do?

2.2.1 RCPT TO: Actions

At the SMTP RCPT TO: command, CanIt-PRO examines the envelope sender and SMTP relay address, and makes decisions according to Figure 2.2.

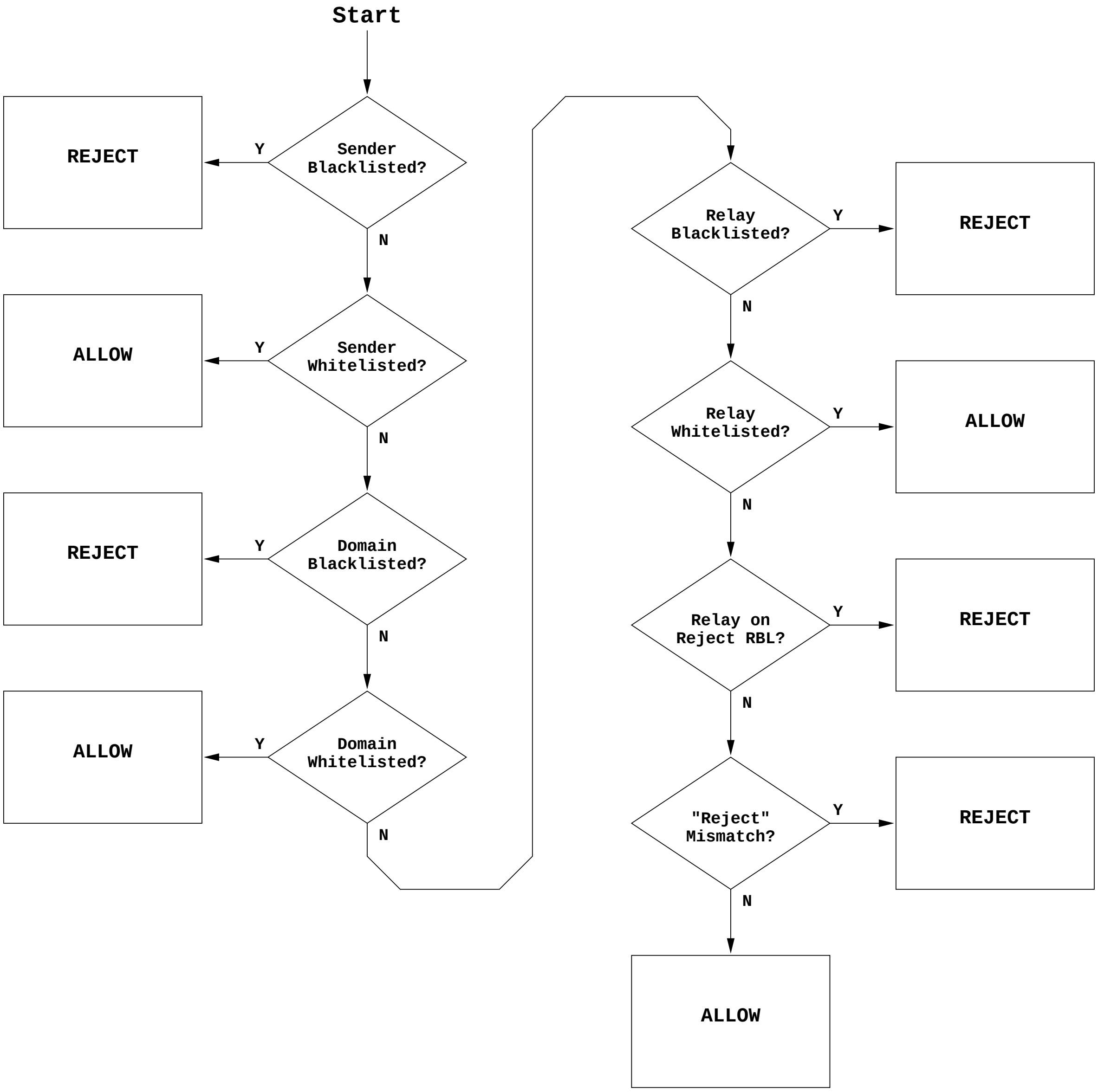


Figure 2.2: RCPT TO: Decision

Here are the steps illustrated in Figure 2.2. They determine the response to the RCPT TO: command. The first rule that matches returns the result; subsequent rules are not tested.

1. If the recipient is blacklisted, the command is rejected. Blacklisted recipients can *never* receive e-mail.
2. If the recipient has opted out of spam-scanning, the command is accepted.
3. If the sender address is blacklisted, reject the command with an SMTP failure code.

4. If the sender address is whitelisted, accept the command. (That is, permit the SMTP transaction to continue. The message may be rejected later for other reasons.)
5. If the domain of the sender is blacklisted, reject the command.
6. If the domain of the sender is whitelisted, accept the command.
7. If the sending relay's IP address is blacklisted, reject the command.
8. If the sending relay's IP address is whitelisted, accept the command.
9. If the sending relay is on a real-time blacklist for rejection, then reject the command.
10. If a mismatch rule is triggered based on the sender and the host name, and the action for the mismatch rule is "reject", then reject the command.
11. Otherwise, accept the command.

2.2.2 Post-DATA Actions

After the SMTP "DATA" command has transmitted the entire message, CanIt-PRO has enough information to determine a spam score. At this point, it makes decisions according to Figure 2.3.

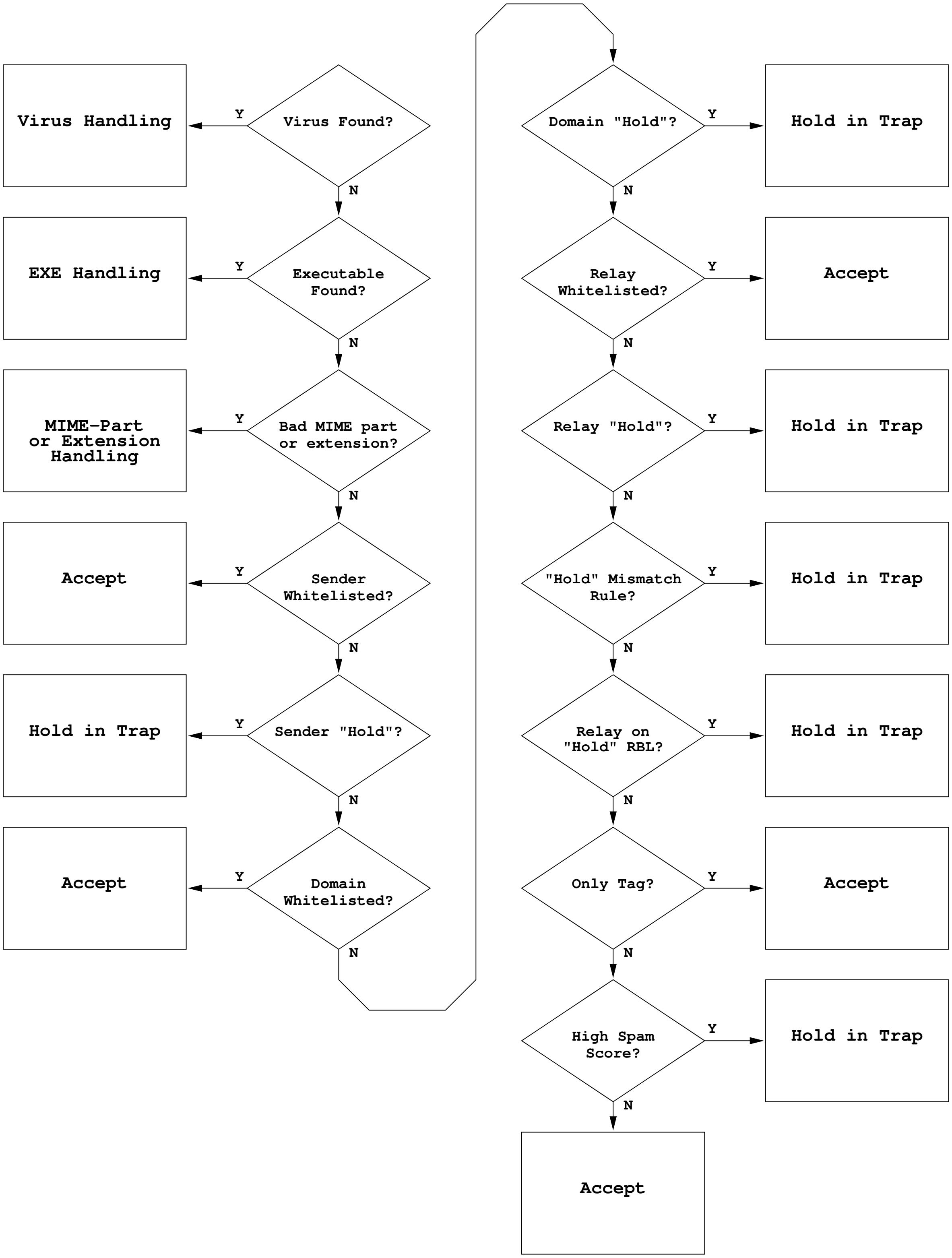


Figure 2.3: Post-Data Decision

Here are the steps illustrated in Figure 2.3. They determine the response to the DATA command. The first rule which matches returns the result; subsequent rules are not tested.

When a message is “held in the trap”, an SMTP temporary-failure code may be issued, or the message may be queued locally, depending on your global settings. When a message is “rejected”, the sending relay receives an SMTP failure code. If the message being rejected was queued locally, it is simply discarded. When a message is “accepted”, it is delivered, and removed from the local queue if it was queued locally.

1. If a virus was found in the message, then the action depends on the virus-handling setting. Here’s what happens for the various settings:
    - **Hold** – the message is held in the trap.
    - **Reject** – the message is rejected with an SMTP failure code.
    - **Discard** – the message is discarded. An SMTP success code is returned.
    - **Accept** – processing continues to step (2) below.
2. If an executable was found in the message, then the action depends on the executable-attachment setting. This follows the same flow as virus-handling above.
3. If a bad MIME part was found, then if the bad part has a “Reject” setting, the message is rejected. Otherwise, the message is held in the trap.
4. If the user has opted-out of spam-scanning, the message is accepted
5. If the sender is whitelisted, the message is accepted.
6. If the sender has a “Hold” setting, the message is held in the trap.
7. If the domain is whitelisted, the message is accepted.
8. If the domain has a “Hold” setting, the message is held in the trap.
9. If the relay is whitelisted, the message is accepted.
10. If the relay has a “Hold” setting, the message is held in the trap.
11. If a relay “Hold” mismatch rule applies, the message is held in the trap.
12. If the relay is on a “Hold” real-time DNS blacklist, the message is held in the trap.
13. If CanIt-PRO is in “Tag Only” mode, the message is tagged (if it looks like spam) and accepted.
14. If the spam score is excessive, the message is held in the trap.
15. Otherwise, the message is accepted.

2.3 Streaming

Because CanIt-PRO allows different recipients to have different spam-processing rules, an incoming message for more than one recipient must be *streamed*.

The diagram in Figure 2.1 shows what happens to messages *after* they have been streamed. If an incoming message arrives for more than one stream, copies are re-mailed to recipients in each stream, and the original message is discarded. Then, each re-mailed message folows the flow in Figure 2.1, with some minor differences that will be explained later.

In Figure 2.1, all of the blacklisting and whitelisting decisions are unique to a stream. It is perfectly feasible for one stream to whitelist a sender, a second stream to blacklist it, and a third stream to do neither.

Messages that are streamed and re-mailed are not held by issuing a temporary-failure code, because they would then reside in your own mail spool and waste resources during repeated sending attempts (until they are approved or rejected.) Instead, held messages are stored in the database, and re-mailed if approved or discarded if rejected.

2.4 How Addresses are Streamed

CanIt-PRO can map e-mail addresses to streams using the following techniques:

Sendmail The Sendmail program is invoked with the `-bv` switch. If the e-mail address maps to a local user, either directly or via a virtusertable or alias table entry, then the local user name is used as the stream name.

For example, if you host virtual domains that are delivered to multidrop mailboxes, and `domain1.tld` is delivered to the user `dom1-user` while `domain2.tld` is delivered to the user `dom2-user`, then **Sendmail** mapping will map any address in `domain1.tld` to `dom1-user` and any address in `domain2.tld` to `dom2-user`.

If the e-mail address resolves to something other than a local user, such as a file, or a piped program, CanIt-PRO will fall back to the default stream for that address.

Note: For the **Sendmail** streaming method to work properly, your Sendmail aliases database must be world-readable (or readable by the `defang` user and group, at any rate.)

Database CanIt-PRO maintains a table of address-to-stream mappings in the Address Mapping Table. If you choose the **Database** technique, then this table is consulted to perform the mapping. You hand-enter the mappings between addresses and streams. In addition, the **Database** technique allows a “wildcard” lookup if the original lookup does not exist.

AsIs This method simply uses the entire e-mail address as the stream name, after stripping angle-brackets and converting to lower-case. Therefore, `xzY@EXAMPLE.com` gets mapped to `xzy@example.com`,

ChopDomain This method simply chops the domain part off the e-mail address. Therefore, `xZyyz@example.com` gets mapped to `xzyyz`.

ChopUser This method chops the user part off the e-mail address. Therefore, `xzyyz@example.COM` gets mapped to `example.com`.

Program This method runs the `account-info` program to determine the stream. Please see Section 6.2.4 on page 74 for details.

Note that no matter what stream method you choose, an exact-match database lookup is always done first. This lets you override the mapping for special cases.

For example, if you host only a single domain, then the **ChopDomain** method is probably fine for most addresses. However, if you also host mailing lists, you'd like to stream spam for the lists to the mailing list owners. In that case, you can add special mappings mapping `list-name@example.com` to `joe-owner`, (where `joe-owner` is the person responsible for `list-name`.)

Because the **Sendmail** and **Program** methods are somewhat inefficient, CanIt-PRO caches results in the database table. This improves efficiency while retaining flexibility. By default, cached entries are valid for 24 hours, but you can adjust the timeout.

2.5 How Streaming Methods are Chosen

Each domain can be streamed using its own method. To select a streaming method, CanIt-PRO first looks up the domain in the Domain Mapping Table. This table holds a list of streaming methods for each domain. If the lookup fails, CanIt-PRO looks up the wildcard entry "`*`" in the Domain Mapping Table and uses that method to stream the address.

Figure 2.4 illustrates how addresses are streamed.

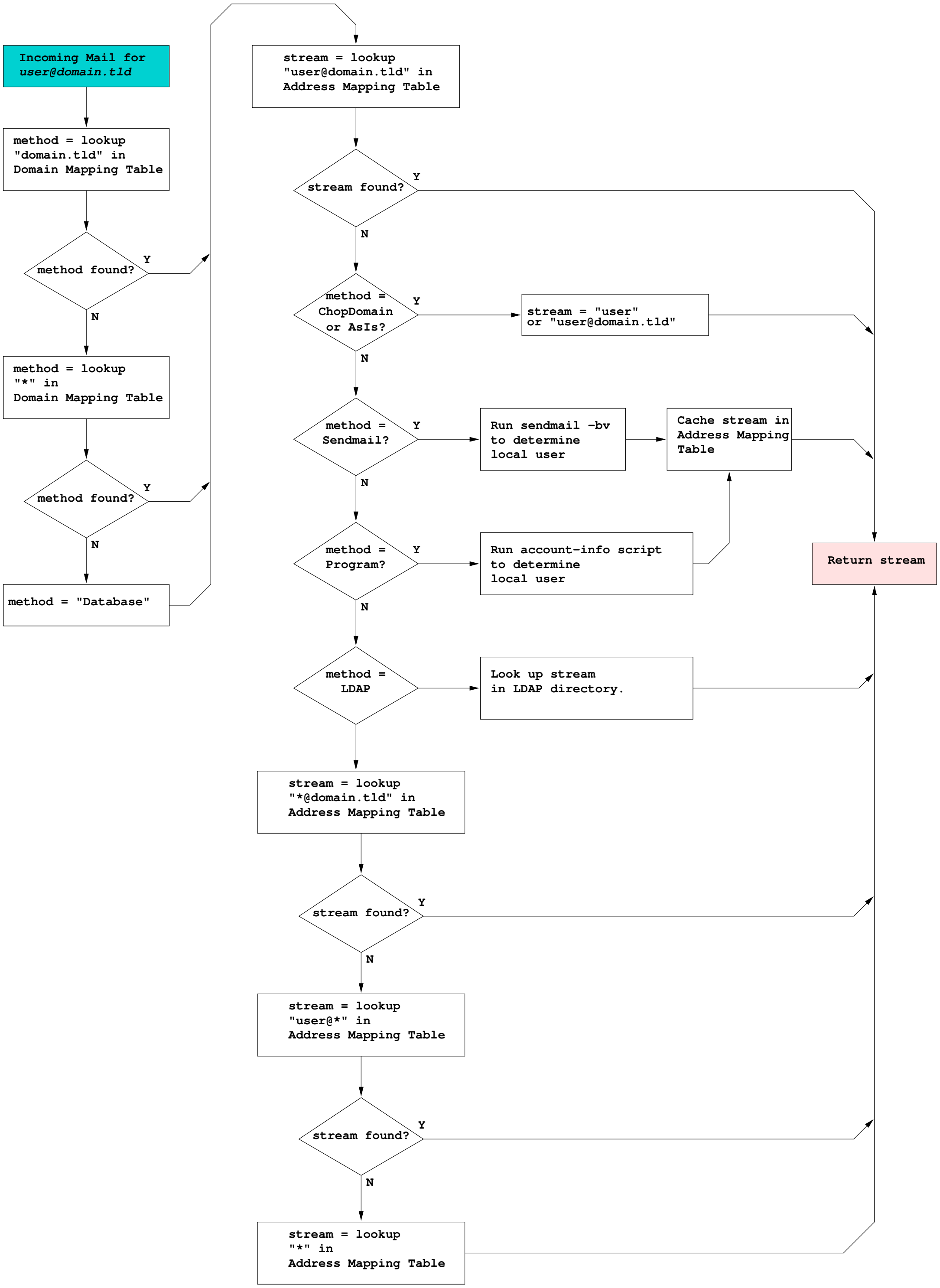


Figure 2.4: Address Streaming

Figure 2.4 looks complicated, but the streaming process is very flexible, and actually quite simple. Here is a description of the figure, with some more details that would crowd the figure too much.

1. For an incoming message to *address@example.com*, CanIt-PRO first looks up *example.com* in the Domain Mapping Table. If that lookup succeeds, CanIt-PRO will have a method (**ChopDomain**, **ChopUser**, **Sendmail**, **Program** or **Database**), and CanIt-PRO proceeds to Step 4.

2. If the lookup fails, the leading component of the domain name is dropped (ie: "subdomain.example.com" becomes "example.com") and we retry Step 1 with the shorter name.

3. If lookups on all domain components fail, CanIt-PRO looks up * in the Domain Mapping Table. This allows you to set a default streaming method for all domains. If that lookup fails, the method defaults to **Database**.

4. Regardless of the method chosen, CanIt-PRO looks up *address@example.com* in the Address Mapping Table. If an exact match is found (and it is not expired if it is a cached entry), the result of that lookup is used as the stream.

5. Otherwise, CanIt-PRO determines the stream as follows:

   - If the method is **ChopDomain**, the *@example.com* part is deleted, and the stream becomes *address*.
   - If the method is **ChopUser**, the *address@* part is deleted, and the stream becomes *example.com*.
   - If the method is **AsIs**, the entire e-mail address is used as the stream name.
   - If the method is **Sendmail**, CanIt-PRO runs "sendmail -bv" with *address@example.com* as an argument. If that address resolves to a single local mailbox, that local mailbox name is used as the stream.
   - If the method is **Program**, CanIt-PRO runs the `account-info` program as described in Section 6.2.4.

   If the stream determination succeeded (**ChopDomain** and **ChopUser** always succeed; **Sendmail** fails if the address does not resolve to a single local mailbox and **Program** fails if the program produces no output), then the stream is returned. In the case of **Sendmail** and **Program**, the stream is also cached in the database.

6. If the previous step failed to determine a mapping method, or the method was set to **Database**, CanIt-PRO looks up *user@**. If that fails, then **@example.com* in the address mapping table. This allows you to map all addresses in a particular domain to a stream. If that fails, as a last resort, CanIt-PRO looks up * in the address mapping table. If that final lookup fails, then a special stream named `default` is used.

2.6 Status of Messages

Every message in the database has one of four statuses. The status names and their meanings are:

one-shot The very first time a message is entered into the database, it is given status **one-shot**. By default, the Web-based interface does *not* display **one-shot** messages.

pending If a message is received and is in the database with status **one-shot**, then its status is changed to **pending**. Pending messages have been received at least twice, and are displayed in the Web-based "Pending Messages" list.

spam The spam-control officer can mark a message as **spam**. If a message marked as **spam** is received, a rejection notice is sent to the sending mail server, and the message is not delivered.

not-spam The spam-control officer can mark a message as **not-spam**. If a message marked as **not-spam** is received, it is delivered as usual.

2.7 Handling of Suspect Messages

As you saw in Figure 2.1 on page 16, CanIt-PRO normally issues an SMTP temporary failure response if a message is held because it is suspected of being spam. This response ensures that the message remains in the sender's queue. The sender will retry transmission periodically, until one of three things happens:

- The message is marked as **spam**. On the next transmission attempt, it will be rejected with a permanent failure response.
- The message is marked as **not-spam**. On the next transmission attempt, it will be accepted and delivered.
- The sending relay times out and bounces the message. Most relays retry transmissions for at least 3 days, so this will not happen unless you do not check the spam trap often enough.

2.7.1 Handling Methods

While keeping the message in the sender's queue is useful, it does mean that your CanIt-PRO installation relies on the server to retransmit. It also may consume excessive bandwidth on a busy site. Therefore, CanIt-PRO has three options for handling suspicious messages:

1. The default handling, **Until-Dispatched**, always replies with a temporary failure indication until the CanIt-PRO operator marks a message as **spam** or **not-spam**.
2. The **First-Time** handling replies with a temporary failure indication the *first* time a suspicious message is received. A lot of spamming software ignores error returns and will never retransmit the message. Failing it the first time, therefore, stops a lot of spam without human intervention. If the message is transmitted a second time, however, it is accepted and held in the CanIt-PRO

database. If the operator marks the message **spam**, it is simply deleted from the database. If the message is marked **not-spam**, CanIt-PRO re-mails it to the original recipient before deleting it from the database.

3. The **Never** handling never replies with a temporary failure indication. Suspicious messages are always accepted and then held in CanIt-PRO's database. Incoming messages immediately move to the **pending** state.

Please note that holding messages locally may greatly increase the disk space used by your PostgreSQL database. Be sure to leave enough disk space to handle all messages you anticipate will be held locally.

2.7.2 Secondary MX Relays

Most organizations have secondary MX hosts that queue mail if the primary host is down. They then relay the queued mail when the primary MX host comes back up. Ideally, CanIt-PRO should run on all of your MX hosts. However, if it can only run on your primary MX host, then all other MX hosts should relay to the CanIt-PRO machine. If you tell CanIt-PRO the IP addresses of the secondary MX hosts, it will automatically use the **Never Tempfail** handling for messages from thoses hosts. (There is no point in keeping mail queued and retransmitted on your secondary MX hosts; it's better to accept and hold the message on the CanIt-PRO machine.)

2.8 One-Shot Messages

As you have seen, a message is not marked as **pending** until the *second* time it is received by CanIt-PRO. The reason for this is that many spammers on dial-up accounts send spam and ignore failure codes. They never retransmit the message again. By having CanIt-PRO not display "one-shot" messages, the workload of the spam-control officer is reduced.

Real mail servers always retransmit messages if they receive a temporary failure code, so the one-shot message handling feature will never prevent legitimate e-mail from going through.

Once a night, a cron job runs on the database and changes all **one-shot** messages to **spam** if the entry is older than one week. In this way, the **one-shot** list is automatically disposed of by CanIt-PRO.

2.9 Database

The incident database is key to the correct operation of CanIt-PRO. Three different agents operate on the database as shown in Figure 2.5:

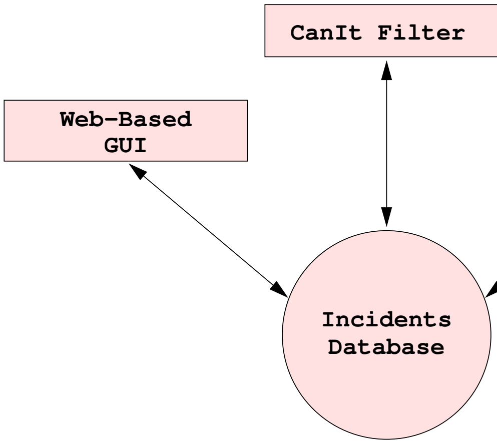


Figure 2.5: Database Agents

The agents operating on the database are:

- The CanIt-PRO Filter – This is the portion of CanIt-PRO that integrates with Sendmail and disposes of spam messages.
- The Web-Based GUI – This is used by users or administrators to mark messages as spam or legitimate. The Web-Based GUI also lets you monitor the levels of spam and take action against specific senders, domains or relay hosts.
- Periodic Jobs – These housekeeping jobs perform operations like moving expired one-shot messages into **spam** status and purging very old messages from the database. Periodic jobs may be started from one of two places:
  1. The `/etc/mail/canit/canit.cron` script, which should be run once a night.
  2. As part of the operation of the CanIt-PRO *ticker*. The ticker is a daemon that starts on bootup and runs continuously, performing background maintenance tasks.

Chapter 3

Streams

3.1 Introduction to Streams

A crucial design element of CanIt-PRO is the *stream*. It is critical to undestand streams before you can use CanIt-PRO effectively. Streams were designed to solve a difficult and complex problem; unfortunately, therefore, understanding them can take some time. However, once you understand streams, you will appreciate their power and flexibility, and understand how to apply them to your particular e-mail setup.

3.2 The Definition of a Stream

A *stream* is a collection of rules and policies. Each stream in CanIt-PRO can have its own rules, settings, thresholds and policies.

Associated with each stream is a *trap*. A trap consists of messages that have been held based on the streams settings. For example, a message can be held because of its spam score, or because it contains a suspicious MIME type.

3.3 Users and E-Mail Addresses

Under many circumstances, a single e-mail address corresponds to a single user. For example, the e-mail address `dfs@roaringpenguin.com` corresponds to the single user `dfs`.

However, most mail setups are more complicated than this. The first complication comes from aliases. For example, the user `dfs` may have, in addition to his normal e-mail address, aliases like `dskoll@roaringpenguin.com` and `davids@roaringpenguin.com`. We would most likely want the same settings and policies to apply to all three aliases.

Another complication comes from list addresses. For example, the e-mail address `sales@roaringpenguin.com` does not correspond to any particular user. Instead, it is a list alias that expands to several users. It might make sense to have a separate set of policies for `sales` than for real users, or it might make sense to assign the policies used by one of the recipients on the sales list.

As we see above, the mapping between users and e-mail addresses is not simple. A single e-mail address may result in delivery to several users (the sales example), or a single user may have several e-mail addresses that all deliver to the same place (the aliases example.)

Streams were invented to give you the flexibility of assigning policies. They act as an intermediate container between e-mail addresses and actual users, and let you assign policies any way you choose. As an example, consider Figure 3.1:

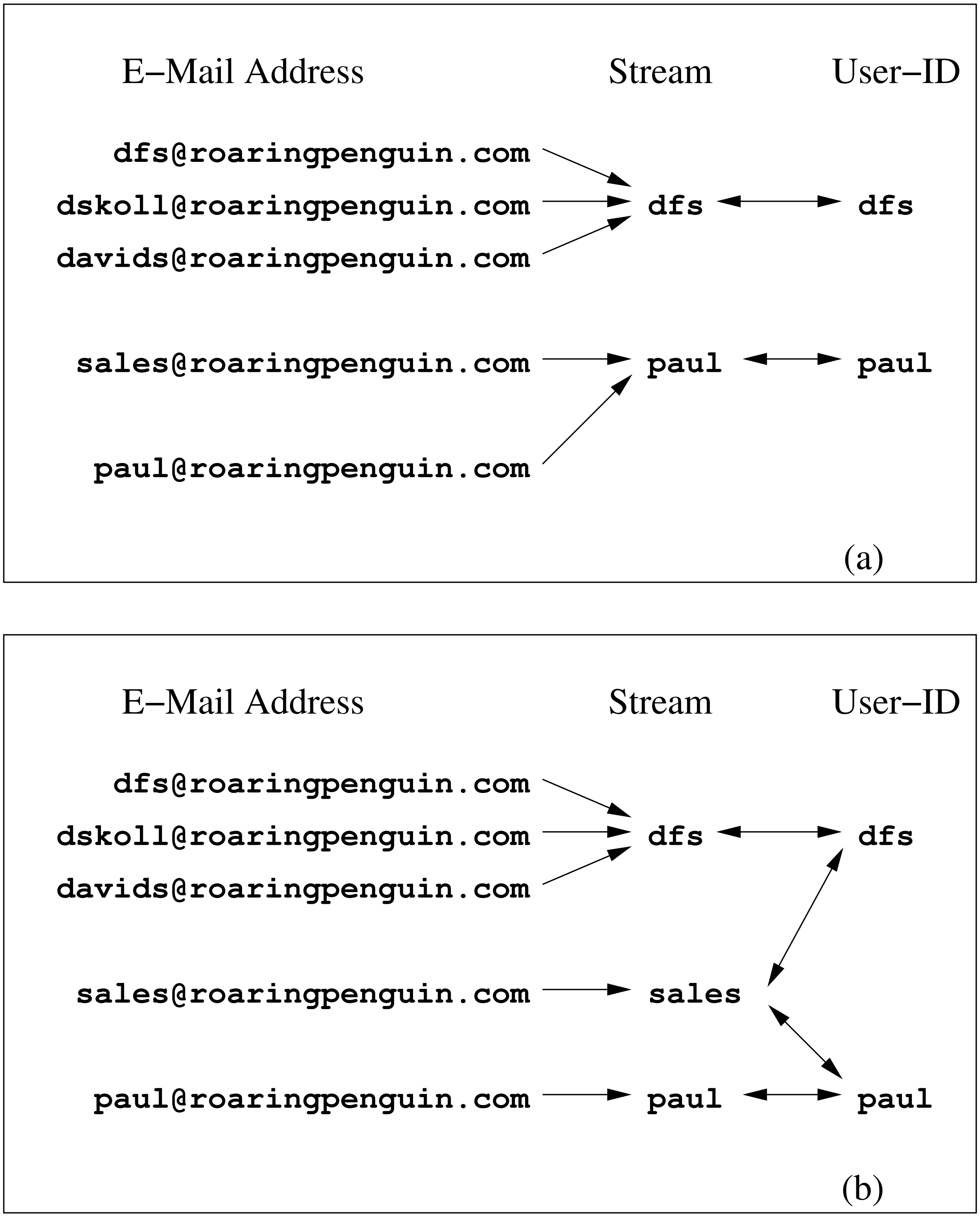


Figure 3.1: Streaming Scenarios

We assume that there are two users, `dfs` and `paul`. We assume that `dfs` has the three aliases shown, and that the `sales` address actually gets delivered to both `dfs` and `paul`.

In Figure 3.1(a), all mail for dfs's aliases go into the dfs stream. Mail for paul goes into the paul stream. Furthermore, mail for sales also goes into paul. Although mail for sales is delivered to two users, all of the settings and policies are controlled by the paul stream, and paul is responsible for clearing the trap.

In Figure 3.1(b), sales has its own stream. It can thus have different settings and rules from either paul or dfs. Furthermore, *both* paul and dfs are given access to the stream, so either of those users can adjust the settings and check the trap for sales.

3.4 Mapping

When e-mail comes in, each recipient address is *mapped* to a stream. We call this process *address mapping*. Once the stream is determined, CanIt-PRO knows which settings and rules to apply for that recipient. The process by which CanIt-PRO maps addresses to streams is illustrated in Figure 2.4 on page 23.

An e-mail address is mapped to a stream in a two-step process:

1. The domain part of the address (everything after the "@" sign) is looked up in the *Domain Mapping Table*. This lookup results in a *method* by which to map the address to a stream.

2. Once the method has been determined, then the address is mapped to a stream using the appropriate method. Details are in Section 4.8 on page 42.

When a user logs in to the Web interface, CanIt-PRO must associate a stream with the user name. By default, CanIt-PRO chooses a stream with the same name as the user's login—this is called the *home stream*. For example, the user dfs would automatically be sent to the stream dfs upon login. However, it is possible to give users access to additional streams, and to change the default login stream. Also, it is possible to change the user's home stream with the account-info script (Section 6.2.4).

Note: Stream names are *case-sensitive*. Thus, a stream called dfs is completely separate from a stream called DFS.

3.5 The Home Stream

A user's home stream is the stream he is placed in when he first logs in to CanIt-PRO. By default, the home stream has the same name as the user's login name. However, you can change this by writing an appropriate account-info script (Section 6.2.4). A user always has access to his or her home stream.

3.6 The "default" Stream

CanIt-PRO treats the stream named **default** specially in several ways:

- When the database initialization script runs, it sets the login stream for the CanIt-PRO administrator to **default**.
- If a stream mapping cannot be found for an address, the address is mapped to **default**.
- Any blacklists, whitelists and rules defined in the **default** stream are inherited by all other streams. (However, stream owners can turn this inheritance off if they wish.)

Chapter 4

CanIt-PRO Setup

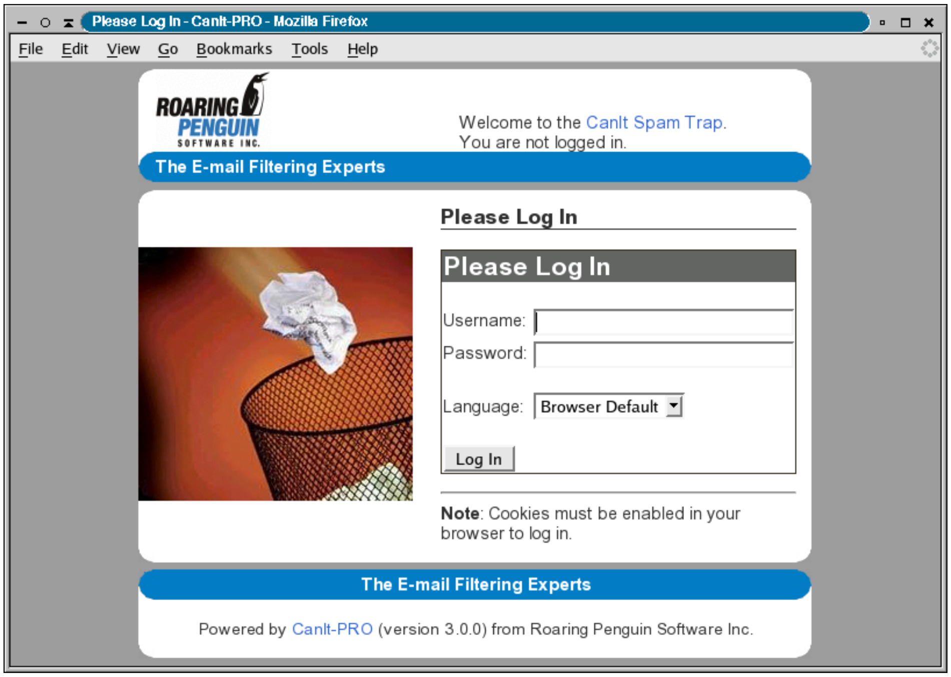
4.1 The Web Interface

Using your Web browser, open the URL where you installed the CanIt-PRO PHP pages.

For example, if your server is `mailserver.mydomain.com` and you installed the GUI in the directory `canit` under your Apache document root, the URL to open would be:

```
http://mailserver.mydomain.com/canit/
```

You will see the Login Screen (Figure 4.1):


Please Log In - CanIt-PRO - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
ROARING PENGUIN SOFTWARE INC.
The E-mail Filtering Experts
Log In
The E-mail Filtering Experts

Welcome to the CanIt Spam Trap.
You are not logged in.

Please Log In

Please Log In

Username:

Password:

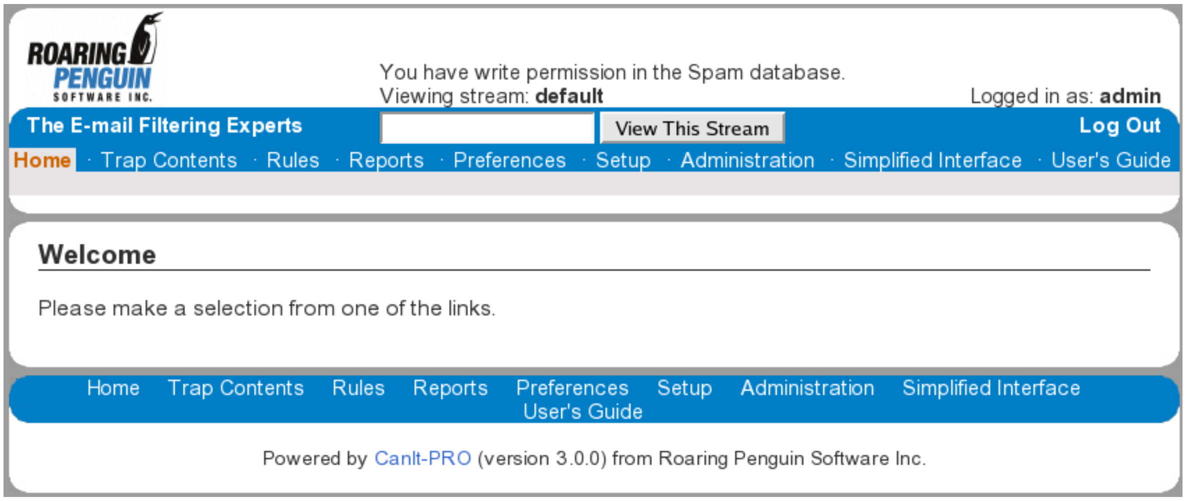
Language: Browser Default

Note: Cookies must be enabled in your browser to log in.

Powered by CanIt-PRO (version 3.0.0) from Roaring Penguin Software Inc.

Figure 4.1: Login Screen

Log in using the name and password you selected when you initialized the CanIt-PRO database. (See Section E.2 on page 130 if you've forgotten the password.) In our example, we used "admin" and "secret". You should see the CanIt-PRO welcome screen:


ROARING PENGUIN SOFTWARE INC.
The E-mail Filtering Experts
View This Stream
Log Out
Home · Trap Contents · Rules · Reports · Preferences · Setup · Administration · Simplified Interface · User's Guide
Home Trap Contents Rules Reports Preferences Setup Administration Simplified Interface User's Guide

You have write permission in the Spam database.
Viewing stream: **default**

Logged in as: **admin**

Welcome

Please make a selection from one of the links.

Powered by CanIt-PRO (version 3.0.0) from Roaring Penguin Software Inc.

Figure 4.2: Welcome Screen

4.2 The Setup Menu

The "Setup" main menu entry contains sub-entries for various parts of basic CanIt-PRO setup. Under the "Setup" menu, you will find:

- "Wizards" – a collection of tools for easily configuring certain common scenarios.
- "License Key" – a page to enter your CanIt-PRO license key.
- "Verification Servers" – a table allowing you to check recipients against internal servers before CanIt-PRO will accept them.
- "Known Networks" – a table allowing you to change aspects of CanIt-PRO behavior for mail originating from certain known networks.
- "Features" – a page allowing you to turn off certain CanIt-PRO functionality to improve performance.
- "System Check" – a page that performs a few simple "sanity checks" on your CanIt-PRO system.
- "Templates" – a page for configuring templates that control how CanIt-PRO appends Bayesian voting information to e-mail and the format of Pending Message Notifications.
- "Domain Mappings" and "Address Mappings" – two tables that tell CanIt-PRO how to convert an e-mail address to a stream.

- "Authentication Mappings" and "User Lookups" – pages for integrating CanIt-PRO with external directories or authentication mechanisms. These are fully described in Chapter 6.

4.3 Wizards

The "Wizards" menu item allows you to ease CanIt-PRO setup by using a *wizard* to speed through choosing some basic settings.

4.3.1 Basic Setup Wizard

The **Basic Setup Wizard** helps you set some basic settings essential to the operation of CanIt-PRO. On a new CanIt-PRO installation, you should follow the steps in this wizard to set some basic settings to sensible values.

4.3.2 RPTN Setup Wizard

The **RPTN Setup Wizard** configures RPTN, the Roaring Penguin Training Network. See Section 7.4 on page 78 for details.

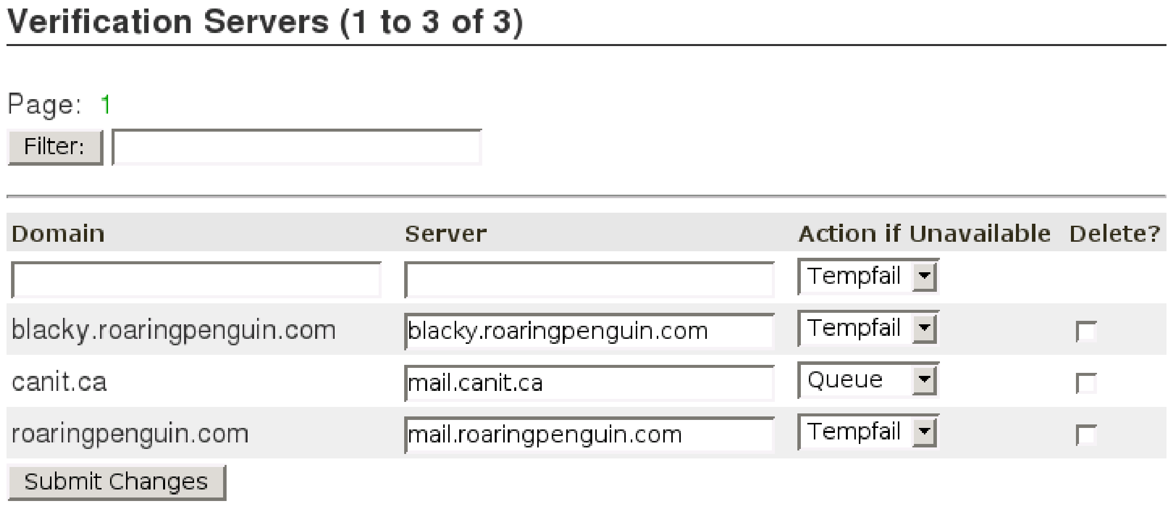
4.4 Verification Servers

If CanIt-PRO acts as a filtering server that always forwards mail on to other machines, you can have it check recipient addresses against other machines. The internal machine that verifies recipient addresses is called a *Verification Server*.

Note: This feature only works if the internal machines fail RCPT commands for unknown users. Versions of Microsoft Exchange prior to Exchange 2003 cannot do this; recent versions can be configured to do it following the instructions at `http://support.microsoft.com/kb/823866/#6`.

CanIt-PRO allows you to enter a list of domains and the machines that will verify mail for the domains. (Note that this does *not* change your Sendmail configuration; you need to ensure that Sendmail's `mailertable` routes mail appropriately.)

To edit the verification server list, click on "Setup" and then "Verification Servers". The following page appears:



Verification Servers (1 to 3 of 3)

Page: 1

Domain	Server	Action if Unavailable	Delete?
		Tempfail	
blacky.roaringpenguin.com	blacky.roaringpenguin.com	Tempfail	□
canit.ca	mail.canit.ca	Queue	□
roaringpenguin.com	mail.roaringpenguin.com	Tempfail	□

Figure 4.3: Verification Servers

In this example, CanIt-PRO performs the following checks:

- Any recipient whose domain is `blacky.roaringpenguin.com` is verified against the machine `blacky.roaringpenguin.com`
- Any recipient whose domain is `canit.ca` is verified against the machine `mail.canit.ca`
- Any recipient whose domain is `roaringpenguin.com` is verified against the machine `shevy.roaringpenguin.com`

To add a domain/server pair to the table:

- Enter the domain name in the **Domain** box and the server name or IP address in the **Server** box.
- Sometimes, your verification server may be down or unreachable. If you would like CanIt-PRO to tempfail mail in this case, then select "Tempfail" as the **Action if Unavailable**. If you would prefer CanIt-PRO to queue mail, select "Queue".

Note: Be careful: If you choose to queue mail if the verification server is unavailable, you may end up scanning and queuing a large amount of mail for nonexistent recipients. We recommend setting the action to "Tempfail" unless you are willing to accept this risk.

- Click **Submit Changes**

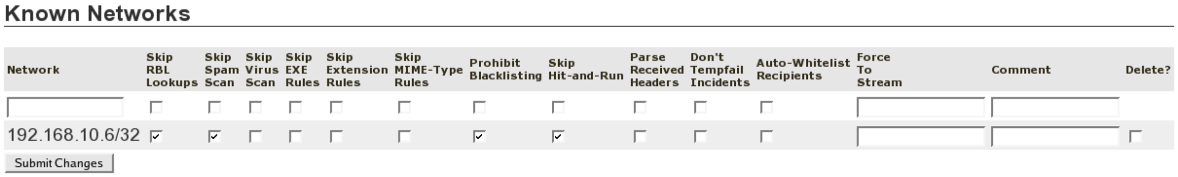
To delete a domain/server pair from the table, enable the appropriate **Delete** checkbox and click **Submit Changes**.

If you enter a string in the "Filter:" box, then CanIt-PRO limits the display to entries whose **Domain** or **Server** columns contain that string.

4.5 Known Networks

CanIt-PRO allows you to enter a list of “known networks”. These are typically networks that you control, and for which you wish to alter the normal CanIt-PRO processing flow. For example, you may not wish to scan outgoing mail for spam; if all outgoing mail originates from a known set of IP addresses, you can tell CanIt-PRO to skip spam-scanning for mail originating from those IP addresses.

To edit the list of known networks, click on “Setup” and then “Known Networks”. The Known Networks page appears:



Known Networks

Network	Skip RBL Lookups	Skip Spam Scan	Skip Virus Scan	Skip EXE Rules	Skip Extension Rules	Skip MIME-Type Rules	Prohibit Blacklisting	Skip Hit-and-Run	Parse Received Headers	Don't Tempfail Incidents	Auto-Whitelist Recipients	Force To Stream	Comment	Delete?
	□	□	□	□	□	□	□	□	□	□	□			
192.168.10.6/32	☑	☑	□	□	□	□	☑	☑	□	□	□			□

Figure 4.4: Known Networks

In the example in Figure 4.4:

- The host 192.168.10.6 will not be looked up in any RBL blacklists.
- Mail originating from 192.168.10.6 will not be scanned for spam:
- Mail originating from 192.168.10.6 cannot be blacklisted. That is, any sender, domain or host blacklists will be ignored.
- Hit-and-run detection will be turned off for 192.168.10.6.
- Mail originating from 192.168.10.6 will be streamed into the **Outgoing** stream, no matter what.

To add a network to the list of known networks:

1. Enter the network address in the **Network** box. A network address can either be a single IP address, or a network address in CIDR notation: `a.b.c.d/bits`. In this notation, `a` through `d` are decimal numbers from 0 to 255, and `bits` is a number from 1 to 32 specifying how many bits of the address are significant. Note that the remaining bits (32 − `bits`) must be zero.

   Here are examples of network addresses:

   - `192.168.1.0/24` specifies the Class C network 192.168.1.0 through 192.168.1.255.
   - `10.5.2.0/23` specifies the IP addresses 10.5.2.0 through 10.5.3.255.
   - `192.168.5.5/24` is *invalid*, because the lower 8 bits of the address must be zero.

2. Choose the characteristics you wish to apply to hosts in the known network:

   - To skip DNS-based RBL lookups, enable **Skip RBL Lookups**.

- To skip spam-scanning, enable **Skip Spam Scan**.
- To skip virus-scanning, enable **Skip Virus Scan**.
- To skip checking for Windows executables, enable **Skip EXE Rules**.
- To skip filename and filename extension checking, enable **Skip Extension Rules**.
- To skip MIME-type checking, enable **Skip MIME-Type Rules**.
- To skip enforcement by CanIt-PRO of maximum message size, enable **Skip Size Limit Checks**.
- To prevent sender, domain or host blacklists from applying to mail sent from the network, enable **Prohibit Blacklisting**.
- To skip hit-and-run checks for hosts in the network, enable **Skip Hit-and-Run**.
- To have CanIt-PRO parse Received: headers to find the actual relay, enable **Parse Received Headers**. CanIt-PRO parses through the headers until it finds a host that isn't a secondary MX machine, and isn't in a known-network with this flag set.
- To have CanIt-PRO hold suspect messages locally if they originate from the known network (rather than tempfailing them), enable **Don't Tempfail Incidents**.
- To auto-whitelist recipients of messages from a known network, enable **Auto-Whitelist Recipients**. This means that for messages originating from the network, the *recipients* of the message are whitelisted in the *Sender Rule* table.

  Note that auto-whitelisting is not applied if any of these conditions holds:

  - The message has a "Precedence: bulk" or "Precedence: junk" header.
  - The message has an "Auto-Submitted" header, as specified in RFC 3834.
  - The message is a bounce message (in other words, the sender is <>.
  - The message subject contains "[no-whitelist]". In this case, the [no-whitelist] tag is removed before the message is delivered (so that the recipients do not see it.)
  - Auto-whitelisting has been disabled under **Preferences : Stream Settings** for the sender's stream.

  Note that some auto-responder software ignores RFC 3834 and fails to add an "Auto-Submitted" header. This could lead to situations in which CanIt-PRO auto-whitelists someone because of an auto-response. If you cannot convince your auto-responder software to add an Auto-Submitted header, you should complain to the vendor of that software in an attempt to make it RFC-compliant.

  If a stream inherits from a *final* stream, then the whitelist rule is created in the final stream. Otherwise, it is created in the actual stream itself.
- To force all mail from the network to be streamed into a specific stream, enter the name of the stream in the **Force To Stream** box.

3. Click **Submit Changes** to have your changes take effect.

To edit an existing known network, simply adjust the attributes as required and click **Submit Changes**. To delete a known network, enable the **Delete?** checkbox and click **Submit Changes**.

Note: Known Networks are stored in memory in each Perl scanner. CanIt-PRO is not designed to operate with more than a few dozen known networks; you should avoid creating more than 50 or so known networks.

4.5.1 Overlapping Networks

If you add two networks that overlap, CanIt-PRO will use the most-specific network for a given host. That is, CanIt-PRO will choose the smallest network that contains a given host. For example, if you create the known networks `192.168.1.0/24` and `192.168.1.240/28`, then hosts in the range `192.168.1.240` through `192.168.1.255` will use the `192.168.1.240/28` settings, whereas hosts from `192.168.1.0` through `192.168.1.239` will use the `192.168.1.0/24` settings.

4.6 Features

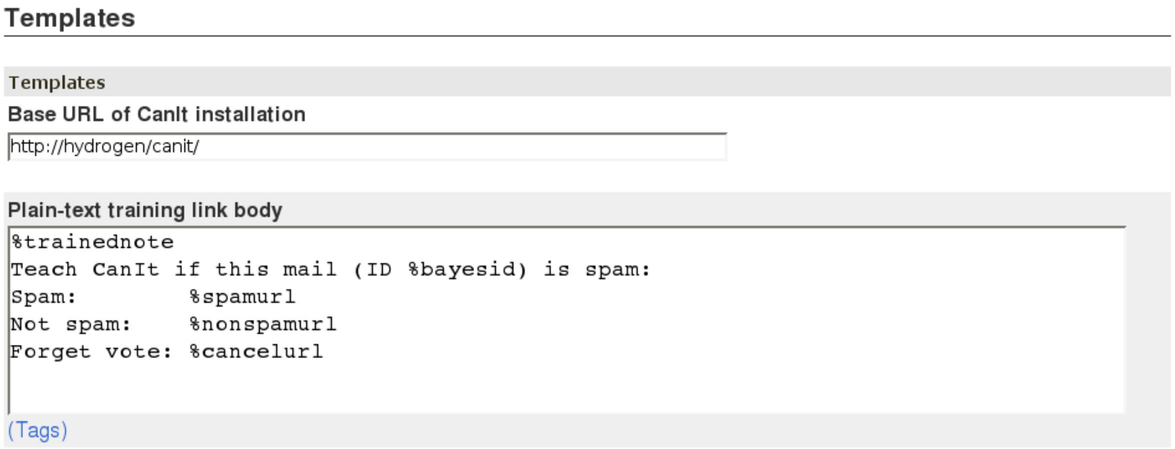
The "Features" page allows you to globally disable certain CanIt-PRO features to reduce the number of database queries. Note that disabling a feature *completely disables it system-wide*. Unless you know for sure that you don't need a feature, and you know that the load savings will be worth turning it off, you should leave all features enabled.

To disable a set of features, click on **No** in the **Enabled** column for the features you want to disable. Then click **Submit Changes**.

4.7 Templates

CanIt-PRO uses templates to configure how Bayes training information is added to messages and to configure the appearance of Pending Message Notifications.

To configure templates, click on **Setup** and then **Templates**. The Templates screen appears:



Templates

Templates

Base URL of CanIt installation

http://hydrogen/canit/

Plain-text training link body

```
%trainednote
Teach CanIt if this mail (ID %bayesid) is spam:
Spam:         %spamurl
Not spam:     %nonspamurl
Forget vote:  %cancelurl
```

Figure 4.5: Templates

The various templates you can configure are:

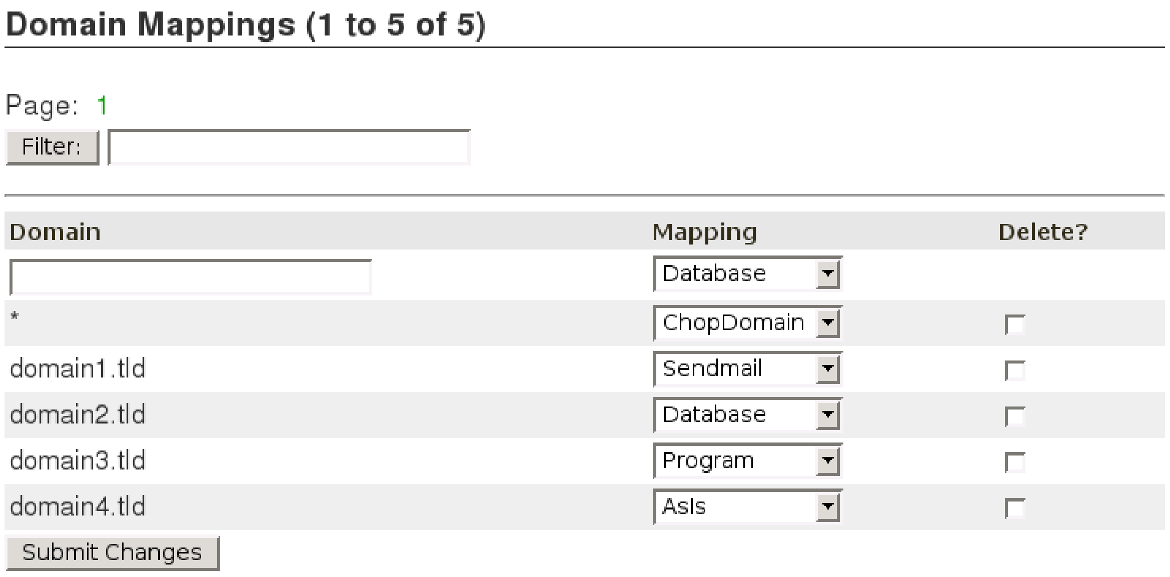
- **Base URL of CanIt installation** is used to construct URLs in messages sent out by CanIt-PRO. It is the same as the corresponding global setting.

- **Plain-text training link body** specifies the appearance of Bayesian training links added to plain-text messages.
- **HTML training link body** specifies the appearance of Bayesian training links added to HTML messages.
- **Pending notification e-mail subject** specifies the subject to put in Pending Notification messages.
- **Pending notification e-mail body** specifies the body of Pending Notification messages

Note that most templates include various “replacement tags”. For example, in the training link templates, the sequence of characters **`%spamurl`** will be replaced with a URL that votes the message as spam. To see the list of available replacement tags, click on the “(Tags)” link near the template entry box.

4.8 The Domain Mapping Table

Recall from Figure 2.4 on page 23 that CanIt-PRO uses a Domain Mapping Table to determine how to stream messages for each domain. The table contains a list of domains with a corresponding *lookup method*. To edit the Domain Mapping Table, click on “Setup” and then “Domain Mappings”. The Domain Mappings page appears:



Domain Mappings (1 to 5 of 5)

Page: 1

Domain	Mapping	Delete?
	Database	
*	ChopDomain	□
domain1.tld	Sendmail	□
domain2.tld	Database	□
domain3.tld	Program	□
domain4.tld	AsIs	□

Figure 4.6: Domain Mappings

To add a mapping method for a particular domain, enter the domain name in the top row of the table and select a value in the **Mapping** column. The possible choices are:

- **Sendmail**—CanIt-PRO will invoke the Sendmail program like this:

  ```
  sendmail -OForwardPath=/dev/null -bv address@domain.tld
  ```

  If the address resolves as "deliverable" to a *single* local mailbox, that mailbox name is used as the stream name.

- **Database**—CanIt-PRO will look up a stream mapping in the Address Mapping Table (Section 4.9).

- **AsIs**—CanIt-PRO converts an address to a stream by removing any angle-brackets and converting letters to lower-case.

- **ChopDomain**—CanIt-PRO converts an address to a stream simply by chopping off the *`@domain.tld`* part, removing any angle-brackets, and converting to lower-case.

- **ChopUser**—CanIt-PRO converts an address to a stream simply by chopping off the *`address@`* part, leaving just the domain (without angle-brackets and converted to lower-case.)

- **Program**—CanIt-PRO converts an address to a stream by executing the `account-info` program. Please see Section 6.2.4 on page 74 for more details. Note that **Program** is deprecated; you should create and use a User Lookup method instead.

- **None**—CanIt-PRO removes the domain from the Domain Mapping Table.

- If you have added external User Lookup methods (Chapter 6), some of them may appear as additional choices. For example, the **LDAP** and **Program** User Lookup methods can convert an address to a stream.

Click **Submit Changes** to save your changes.

To modify the mapping for an existing domain, select a new mapping in the **Mapping** column and click **Submit Changes**.

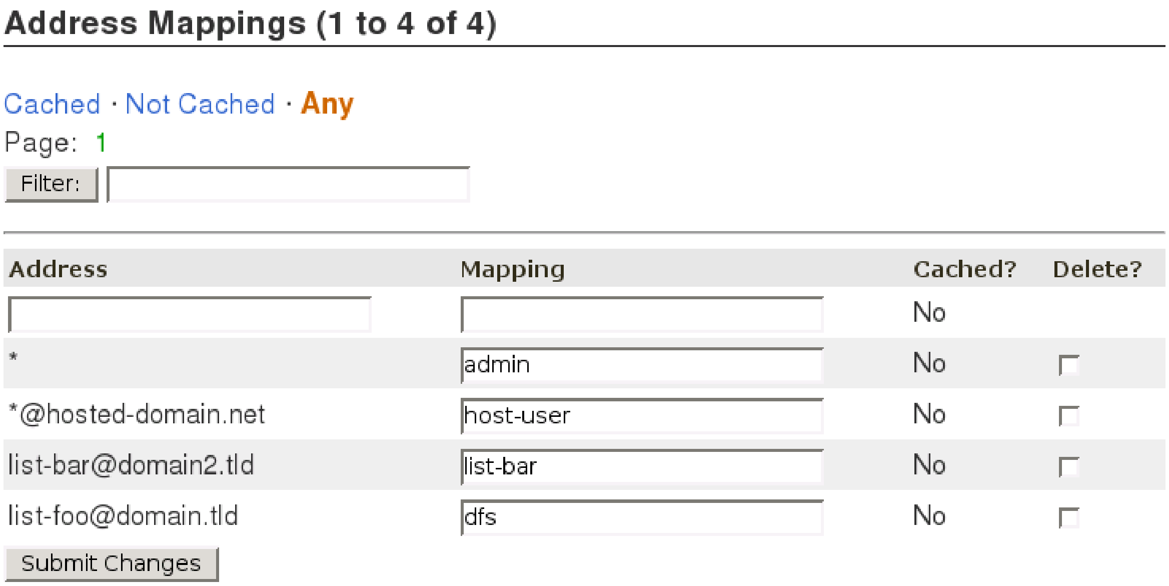
The special domain `*` is used as a last resort if the actual domain is not found. You may enter a mapping for `*` to set a default mapping. If there is no `*` entry and a domain is not found in the Domain Mapping Table, then CanIt-PRO uses a default lookup method of **Database**.

If you enter a string in the "Filter:" box, then CanIt-PRO limits the display to entries whose **Domain** or **Mapping** columns contain that string.

4.9 The Address Mapping Table

CanIt-PRO uses an Address Mapping Table (Figure 2.4 on page 23) to map e-mail addresses to streams. The Address Mapping Table is used both for hand-entered entries placed there by the CanIt-PRO administrator, and for caching the results of the **Sendmail** and **Program** mapping methods. These methods are potentially slow, so caching their results can improve performance.

To edit the address mapping table, click on "Setup" and then "Address Mappings". The Address Mappings page will appear:



Address Mappings (1 to 4 of 4)

Cached · Not Cached · Any

Page: 1

Address	Mapping	Cached?	Delete?
		No	
*	admin	No	□
*@hosted-domain.net	host-user	No	□
list-bar@domain2.tld	list-bar	No	□
list-foo@domain.tld	dfs	No	□

Figure 4.7: Address Mappings

To add an entry for a new e-mail address, enter the new address in the **Address** column of the first row, and enter the stream name in the **Mapping** column. Then click **Submit Changes**.

To edit an existing entry, edit the text in the **Mapping** column and click **Submit Changes**. To delete an entry from the table, click the **Delete** link in the appropriate row.

Click on “Not Cached” to see only non-cached (hand-entered) entries, “Cached” to see only cached entries, or “Any” to see all entries in the Address Mapping Table.

If you enter a string in the “Filter:” box, then CanIt-PRO limits the display to entries whose **Address** or **Mapping** columns contain that string.

4.9.1 Wild-Card Entries

The address mapping table may contain three types of wildcard entries:

1. The entry `user@*` is used if CanIt-PRO is unable to map an address to a stream with an exact match (or if the **Sendmail** or **Program** methods fail.) If you run several domains, but all user-parts are the same, this wildcard can be useful.

2. The entry `*@domain.tld` is used if the previous wildcard does not match anything. Use this entry to set up a default stream for e-mail to a particular domain.

3. The entry `*` is used as a last resort if the previous wildcards did not match.

4.10 The default Stream

CanIt-PRO has a built-in stream name which is reserved, and which cannot be used for other purposes. This stream is named `default`, and is used as follows:

If CanIt-PRO is unable to map an address to a stream (for example, if there are no exact or wildcard matches in the database and the Sendmail or Program methods fail), the address is mapped to the hard-coded stream `default`. The CanIt-PRO administrator should check the `default` stream from time to time.

The `default` stream also contains whitelists, blacklists, custom rules and mismatch rules that all other streams can inherit. The factory default is for all streams to inherit the lists and rules from `default`, but you can disable this if you wish. List and rule inheritance work as follows, for streams which inherit from `default`:

- Senders, hosts, domains, extension rules, MIME type rules and mismatch rules are first looked up in the stream's table. If no entry is found, they are looked up in `default`'s table.
- Custom rules are evaluated first for the given stream, and then for `default`. Their scores are added together. Note that if the same rule appears in both the stream's rule set and `default`'s rule set, it is counted twice.

Normal (non-administrator) users can always switch to the `default` stream (to review the rules, blacklists and whitelists), but only the administrator has write access in the `default` stream.

4.11 Mapping Scenarios

To give a feel for how to use the mapping, we illustrate a few common scenarios.

4.11.1 Central Scanning with Opt-Out

If you run a mail server and wish to centralize spam-scanning, but you have some users who wish to opt out or handle their own spam, you can do it as follows:

In the Address Mapping Table, add this catch-all entry:

Address	Stream
`*`	`admin`

This streams most users' e-mail to the “admin” stream for centralized processing. If user `joe@mydomain.tld` does not want his mail examined by the spam control officer, simply add another entry:

Address	Stream
`joe@mydomain.tld`	`joe`

This streams mail for `joe@mydomain.tld` to `joe`.

4.11.2 Single Domain

If you host a single e-mail domain, and each user's login name is simply the first part of his/her e-mail address, setting up mappings is easy. In the Domain Mapping Table, add a single entry:

Domain	Mapping Method
*	ChopDomain

4.11.3 Single Domain with Aliases and Mailing Lists

Most likely, your scenario is more complex than in Section 4.11.2. You probably host mailing lists, and have aliases. Let's suppose you host a list called `tv-list@domain.tld`, which is run by `jane`, and that your `sales@domain.tld` is an alias which gets expanded to `jim` and `bob`.

You can still use the same Domain Mapping as Section 4.11.2. You have two options for handling the mailing list and sales alias:

1. Allow `jane` to access the `tv-list` stream, and allow `jim` and `bob` (or delegate one of them) to access the `sales` stream. Jane will have to remember to check the `tv-list` trap as well as her own trap, and similarly for Bob and Jim.

2. Add address mappings like this:

Address	Stream
tv-list@domain.tld	jane
sales@domain.tld	bob

   Explicit entries in the Address Mapping Table will override even the **ChopDomain** method.

   Here, Jane's trap will contain messages both for herself directly and the mailing list she runs. Bob's trap will contain his messages and messages for `sales`. (Clearly, you've delegated spam handling for `sales` to Bob alone.)

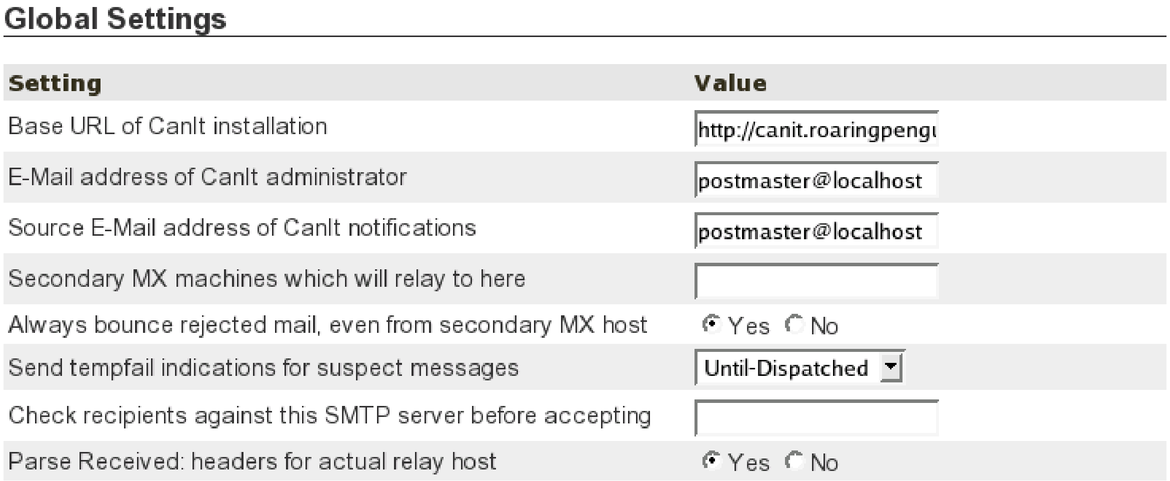
   (You can, of course, use Method 1 for `tv-list` and Method 2 for `sales`. It's up to you.)

Chapter 5

CanIt-PRO Administration

5.1 Global Settings

The first administrative task you should undertake is to set up global settings. Click on the “Administration” link. You will see the global settings screen:



Global Settings

Setting	Value
Base URL of CanIt installation	http://canit.roaringpengu
E-Mail address of CanIt administrator	postmaster@localhost
Source E-Mail address of CanIt notifications	postmaster@localhost
Secondary MX machines which will relay to here	
Always bounce rejected mail, even from secondary MX host	● Yes ○ No
Send tempfail indications for suspect messages	Until-Dispatched ▼
Check recipients against this SMTP server before accepting	
Parse Received: headers for actual relay host	● Yes ○ No

Figure 5.1: Global Settings

Note: Some of the settings below have per-stream equivalents. If no per-stream value is set, then the global values are used as the default setting. However, if a user sets per-stream settings, then they override the global settings. Per-stream settings will be indicated in the description which follows.

The global settings have the following meanings:

Base URL of CanIt installation Enter the base URL of CanIt-PRO. This should be the hostname and path to the web interface that you wish CanIt-PRO to use when generating URLs. If your CanIt-PRO host has multiple hostnames, you will wish to pick one hostname here.

Note: If you do not set the Base URL, then CanIt-PRO cannot create voting links for Bayesian training (See the CanIt-PRO User's Manual for details.)

E-Mail address of CanIt administrator Enter the e-mail address of the person responsible for CanIt-PRO.

Source E-Mail address of CanIt notifications Enter the e-mail address from which CanIt-PRO notifications should be sent.

Secondary MX machines which will relay to here If you have secondary MX machines which relay to your CanIt-PRO machine, enter their IP addresses here. Separate multiple entries with commas. CanIt-PRO will skip most relay-based tests for secondary MX machines. It will also never send temporary-failure indications if a suspicious message comes from a secondary MX host. Instead, CanIt-PRO will hold the message in its local database until approval.

For a full description of the handling of Secondary MX hosts, please see Section 5.9 on page 63.

Avoid generating DSNs for rejected mail from our MX hosts If set to **Yes** (the default) then CanIt-PRO will silently discard messages that would be rejected due to a high spam score if those messages are relayed from localhost or one of the configured secondary MX hosts. If set to **No**, then the message will be rejected with an SMTP failure code.

Note: If a message is split for multiple streams, it will be considered to have been relayed through localhost

This setting can help in preventing "backscatter" or "joe-jobbing", whereby spammers cause bounce messages to be sent to an unrelated third party. See Section 5.10 on page 64 for more details.

Send tempfail indications for suspect messages This entry controls how CanIt-PRO holds messages. There are four choices:

- **Until-Dispatched** makes CanIt-PRO send temporary failure responses until the administrator handles the held message. This is the default setting. It keeps suspicious messages trapped in the sending relay's queue. Note, however, that CanIt-PRO will not tempfail messages from secondary MX machines; it will store those locally.
- **First-Time** makes CanIt-PRO send a temporary failure notification the first time a suspicious message is received. If the message is retransmitted, CanIt-PRO accepts it, but holds it in its local database until the message is approved or discarded.
- **Never** makes CanIt-PRO always hold suspicious messages in its internal database. It will never keep a message in the sender's queue by replying with a temporary failure response.
- **Always** makes CanIt-PRO send temporary failure responses until the administrator handles the held message, no matter what. It differs from **Until-Dispatched** in that it will even tempfail messages from secondary MX hosts. *Do not* use this setting unless you have administrative control over all your secondary MX hosts and are willing to put up with the extra load the retransmissions place on them.

Unless bandwidth is very scarce, we strongly recommend leaving the setting at **Until-Dispatched**.

Parse Received: headers for actual relay host For mail coming from a secondary MX host, CanIt-PRO can parse the **Received:** headers to determine the actual IP address of the sending host. CanIt-PRO reads each **Received:** header, and the first one from an IP address that is *not* listed as a secondary MX host (and is not 127.0.0.1) is taken as the actual relay address. See Section 5.9 for more details.

Automatically reject messages scoring more than this amount Normally, CanIt-PRO does not automatically reject messages. The default value for this setting is 2000, and it's impossible for a mail message to score that high (unless you create high-scoring custom rules). However, if you are willing to risk losing legitimate e-mail in return for reduced human intervention, you can set this value to something like 10 to 20. Mail scoring more than this value will be rejected without human intervention.

Note that even in tag-only mode, messages scoring over this amount will be rejected. If you never want messages rejected in tag-only mode, you should leave this setting at 2000.

Note: This is a per-stream setting.

Auto-reject messages scoring more than this amount without creating an incident If a message scores higher than this setting, CanIt-PRO rejects it *and does not create an incident.* There is therefore no way to search the trap for such messages. Be sure to set this score high enough that the chances of a false positive are extremely remote. On very busy mail servers, rejecting obvious spam without creating an incident can reduce the load on the database server.

Note: This is a per-stream setting.

Spam threshold CanIt-PRO will hold any messages scoring higher than this amount. The default value of 5 has been carefully tuned to minimize errors. You should not change it lightly.

Note: This is a per-stream setting.

Maximum size of message to scan for spam (kB) Spam-scanning can be very slow on large messages. Furthermore, most spam messages are relatively small, probably under 50kB. Therefore, CanIt-PRO does not scan very large messages for spam. (It still scans for viruses and dangerous attachments.) This setting lets you adjust the size beyond which messages are not scanned for spam.

Maximum allowable message size (kB) - 0 means unlimited This setting specifies the maximum message size CanIt-PRO will accept. Note that if you have a message size limit in your Sendmail configuration file, this setting can be used only to *reduce* the limit, not increase it. As a safety measure, CanIt-PRO will not reject messages smaller than 100kB, regardless of the value of this setting.

Note: This is a per-stream setting.

One reason you might use a limit here rather than in your Sendmail configuration file is that the Known Networks settings (Section 4.5 allow you to disable the size limit check for mail originating from known networks.

Only accept mail for accounts in the Valid Recipients table If this is set to **Yes**, then CanIt-PRO refuses to accept mail for recipients unless they are listed in the Valid Recipients Table (see the CanIt-PRO User's Manual for details.)

Note: This is a per-stream setting.

Reject mail from domains with bogus MX records This setting can take one of three values:

- **No** – do not test sender domains for bogus MX records.
- **Loopback** (the default) – reject mail from any domain that has an MX record in the 127.0.0.0/8 network.
- **All-Bogus** – reject mail from any domain that has an MX record in any of the following networks: 10.0.0.0/8, 127.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16, 169.254.0.0/16, 224.0.0.0/4, 240.0.0.0/5, 0.0.0.0/32 and 255.255.255.255/32.

Note: This is a per-stream setting.

Expire statistics after this many days Once a day, a cron job removes old entries from the statistics table. By default, CanIt-PRO keeps statistics for 10,000 days (around 27 years), but you can lower this setting to as low as 90 days if you do not want to keep old statistics around.

Expire old data after this many days Once a day, a cron job purges old messages, log entries and incidents from the database. We recommend retaining at least 45 days' worth of data, although you might want to lower this on a busy mail server.

Expire messages marked as spam after this many days This setting controls when the cron job expires messages you have marked as spam. Note that it only applies to "frozen" messages—that is, messages that have not only been marked as spam, but have also actually been rejected by CanIt-PRO.

Expire messages marked as non-spam after this many days This setting controls when the cron job expires messages you have marked as non-spam. Note that it only applies to "frozen" messages—that is, messages that have not only been marked as non-spam, but have also actually been delivered by CanIt-PRO.

Mark one-shot messages as spam after this many days The cron job automatically moves **one-shot** messages to **spam** after a certain number of days without a retransmission. We recommend leaving this setting at 7 days.

Mark pending messages as spam after this many days The cron job automatically moves **pending** messages to **spam** after a certain number of days. By default, this is set to 180 days, essentially meaning that pending messages will never automatically be marked as spam. However, you may wish to lower it to 14 days or so if you do not clear the pending trap regularly.

Move new incidents directly to ”Pending If you set this to **Yes**, then the entire "One-Shot" infrastructure is bypassed. New incidents move directly into "Pending", and the Web interface contains no mention of "One-Shot" messages.

Handling for messages containing viruses If you have a virus-scanner compatible with CanIt-PRO, this setting controls how CanIt-PRO deals with virus-bearing messages. **Hold** holds the message in the trap for approval. **Accept** permits the message to pass, while **Reject** rejects it with an SMTP failure code. Finally, **Discard** simply discards the message. We recommend setting this option to **Discard**.

Note: This is a per-stream setting.

Handling for messages containing Windows executables CanIt-PRO can recognize attachments with many “dangerous” Window extensions like `.exe`, `.bat`, etc. The settings **Hold**, **Accept**, **Reject** and **Discard** correspond to the settings described previously for virus-bearing messages.

Note: This is a per-stream setting.

CanIt-PRO considers the following filename extensions to be “dangerous”:

ade	adp	app	asd	asf	asx	bas	bat	chm	cmd
com	cpl	crt	dll	exe	fxp	hlp	hta	hto	inf
ini	ins	isp	js	jse	lib	lnk	mdb	mde	msc
msi	msp	mst	ocx	pcd	pif	prg	reg	scr	sct
sh	shb	shs	sys	url	vb	vbe	vbs	vcs	vxd
wmd	wms	wmz	wsc	wsf	wsh				

If you wish to permit certain extensions that would normally be in the banned list above, you can explicitly permit them using the method described for Filename Extensions in the CanIt-PRO User’s Manual.

Tempfail unknown senders on first transmission CanIt-PRO can keep track of senders, and can send a temporary-failure indication the first time an unknown sender attempts to send e-mail.

Note: This is a per-stream setting.

For more details, see Section 12.1, “Hit-and-Run Spam” on page 97.

WARNING: Do not enable this feature on mail servers that allow direct connection from mail clients. You should only enable this feature if your CanIt-PRO server accepts mail only from other mail servers.

Minimum delay in minutes before accepting retry from unknown senders CanIt-PRO can enforce a minimum retry delay the first time it hears from an unknown sender. This is to prevent spammers from defeating the hit-and-run detection by immediately retrying. The default value, 0, imposes no minimum retry interval. We do not recommend setting this value higher than 30 minutes.

Maximum delay in minutes before accepting retry from unknown senders CanIt-PRO can enforce a maximum retry delay also; if the sender does not retry within this interval, then it is once again considered “unknown”. The default value is 2880 minutes, or two days. We *strongly* suggest keeping this setting at 1440 minutes (one day) or higher. There are real-world mail servers that retry tempfailed mail very infrequently.

Log CanIt actions using syslog If you set this to **Yes**, then CanIt-PRO logs messages using the “info” level and “mail” facility.

Only tag spam – do not hold any messages If you set this to **Yes**, then *no messages are held in the trap because of high spam scores*. CanIt-PRO simply tags the subject line of each message which would have been held with the string “`[SPAM:***]`” and delivers it normally. The number of stars after the SPAM: tag is the integer part of the spam score. Note, however, that any message scoring over the auto-reject threshold will still be rejected.

Be aware that in tag-only mode, CanIt-PRO will still hold messages because of viruses, executables and disallowed MIME types. CanIt-PRO will not hold messages because of sender, host or domain “Hold” rules, but any “Reject” rules will still apply. (But see the next setting.)

Note: This is a per-stream setting.

String to put in tagged subjects This is the string that gets prepended to the subject line in tag-only mode. The default setting is `[Spam:%* %?]`. The following special sequences of characters may be used:

- `%*` is replaced with a string of asterisks, where the length of the string equals the integer part of the spam score.
- `%?` is replaced with the reason a message was tagged, such as **SpamScore**, **HoldSender**, etc.
- `%d` is replaced with the actual spam score as a decimal number (e.g. 13.6)
- `%h` is replaced with the actual spam score as a four-digit integer with leading zeros (e.g. 0013)
- `%p` is replaced with the Bayes probability (a real number from 0 to 1.)
- `%%` is replaced with a percent sign.

Note: This is a per-stream setting.

String to put in subjects of approved messages This string gets prepended to the subject line of messages that you release from the trap. It is useful if you release messages on behalf of others; it lets them know at a glance that the message was trapped as spam, but subsequently approved. The following special sequences of characters may be used:

- `%i` is replaced with the Incident ID.
- `%d` is replaced with the spam score.
- `%?` is replaced with the hold reason.
- `%u` is replaced with the user-ID of the person who approved the message.

If you plan on using this feature, we recommend the following string:

```
[Approved by %u:  #%i]
```

Note: This is a per-stream setting.

Whitelist users who use SMTP authentication If your version of Sendmail is compiled to support the SMTP AUTH extension, you can whitelist mail from authenticated senders by setting this to **Yes**. (The default is **No**.) In this case, mail from authenticated users will not be scanned for spam (but will still be scanned for viruses and executables.)

Silently discard rejected messages rather than remailing with ticker Normally, if a locally-held message is rejected, CanIt-PRO pulls it out of the database and re-mails it, relying on another filtering pass to reject the message and to cause a delivery failure notification to be generated. If you’d prefer to simply discard locally-held-and-then-rejected messages, change this setting to **Yes**.

Text to add to SMTP rejection messages When CanIt-PRO rejects a message, it issues an SMTP failure code. You can add extra text to the rejection code if you like. The text should be short and simple; something like `For assistance, please call the helpdesk at +1-613-555-1234` would be appropriate.

Store both raw and decoded messages in incident database Some e-mail messages are obscured using Base64 encoding or some other encoding scheme. If you change this setting to **Yes**, CanIt-PRO stores both the "raw" and "decoded" message in the incident database. This lets you view encoded messages more reliably, but approximately doubles the disk space used by the incident database. If you set it to **No** (the default), CanIt-PRO stores only the raw message.

The message display Web page can decode some encoded messages, but it is not completely reliable. If you need a completely reliable way to view encoded messages, you should change this setting to **Yes**.

Obscure To, Cc and Bcc fields for non-root users Because CanIt-PRO stores messages that hash identically only once, the To:, Cc: and Bcc: headers of messages may leak recipient information to other recipients of the message. To hide this information, change this setting to **Yes**.

Number of hours to cache address-to-stream lookups As mentioned in Section 2.4, address-to-stream mappings may be cached in the Address Mapping Table. This setting specifies for how long cached entries remain valid.

Users must opt in to anti-spam scanning? If you set this to **Yes**, then users must explicitly opt-in to anti-spam scanning. If users do not opt-in, their mail is simply passed through unchanged. If you set this to **No**, then all users are implicitly opted-in. They can, however, explicitly opt out if they choose.

Users must be approved for anti-spam scanning? If you set this to **Yes**, then the CanIt-PRO administrator's approval is required before a user can opt in to anti-spam scanning. If you are selling anti-spam scanning as a value-added service, you should set this to **Yes**. If anti-spam scanning is part of your basic service, set it to **No**.

Note that opting in and opting out is done on a per-stream basis. Usually, a stream corresponds to a user, but it is possible for a stream to correspond to more than one user, and for a single user to be responsible for more than one stream.

Users authenticated by external means default to simple GUI? If you set this to **Yes**, then users who authenticate via an external authentication mechanism have a much simplified interface to CanIt-PRO by default. This simplified interface is described in Chapter 9.

Switching to expert mode cancels stream inheritance If you use the Simple Interface (Chapter 9), then you may wish to cancel inheritance whenever a user selects the expert interface. In that case, change this setting to **Yes**.

Support the Sendmail 'plus hack' for streaming Some Sendmail configuration files allow users to add a "+" sign followed by arbitrary text to their user names, and use the resulting e-mail addresses for various purposes such as filtering e-mail. If you change this setting to **Yes**, then CanIt-PRO ignores a "+" sign and any following text after the user name part when mapping e-mail addresses to streams.

Note that if you use the “Program” method to stream e-mail, the “+” sign and any following text is retained; it is up to your program to implement the sendmail “plus hack” if you choose.

Scan for viruses prior to streaming incoming mail If you know for sure that you *always* want to reject or discard viruses, regardless of any per-stream settings, then change this setting to **Yes**. It causes any viruses to be discarded or rejected (according to the global virus-handling setting) before any streaming takes place. If a virus comes in for more than one recipient, this can greatly reduce the load on CanIt-PRO. Note that the global virus-handling setting must *not* be set to **Hold** for this setting to take effect.

Enable Bayesian analysis If you set this to **Yes**, then CanIt-PRO’s Bayesian Analysis module is enabled. This setting, and the remaining settings related to Bayesian analysis, are explained in the User’s Manual in the chapter “Bayesian Filtering”.

To make your changes permanent:

- Click on “Update Global Settings”

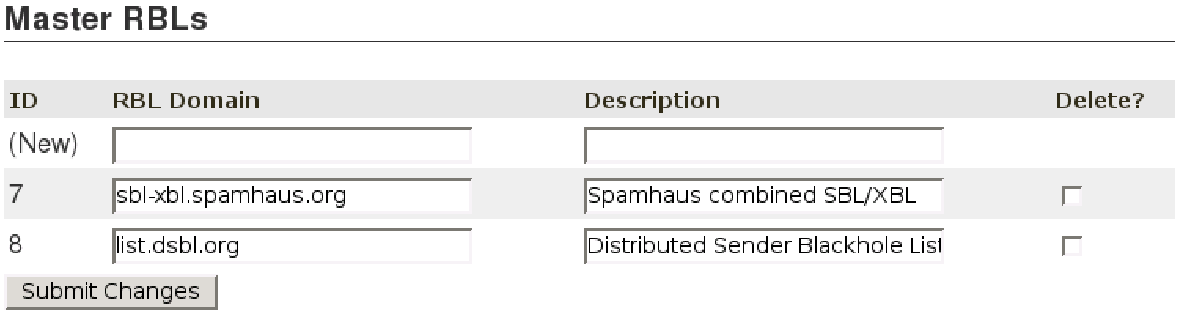
5.2 Real-Time DNS Blacklists

Both Sendmail and CanIt-PRO can make use of DNS-based real-time blacklists. These blacklists allow you to look up the IP address of a host in a special DNS domain, and take action if the host is blacklisted.

You can configure Sendmail to use DNS-based blacklists directly, but you may prefer to handle this with CanIt-PRO, because CanIt-PRO allows you to hold or score messages from hosts on the blacklist rather than outright rejecting them.

5.2.1 Entering the Master List of DNS RBLs

To use DNS-based RBLS, you first enter a *master list* of RBLs that CanIt-PRO can potentially use. To do this, click on “Administration” and then “Master RBLs”. The Master RBLs page appears:



Master RBLs

ID	RBL Domain	Description	Delete?
(New)			
7	sbl-xbl.spamhaus.org	Spamhaus combined SBL/XBL	☐
8	list.dsbl.org	Distributed Sender Blackhole List	☐

Figure 5.2: Master RBLs

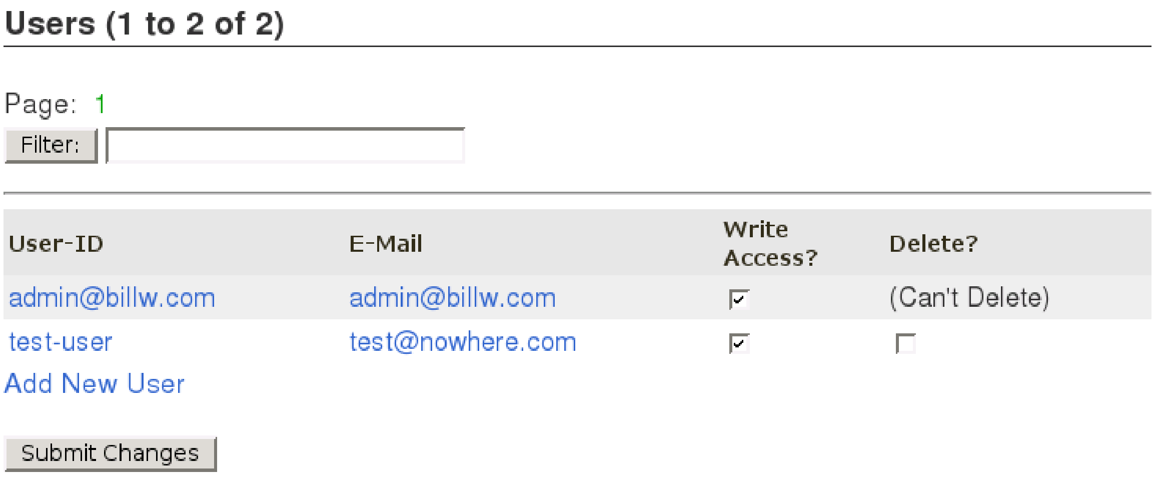
To enter an RBL domain, enter the domain in the **RBL Domain** box and a brief description in the **Description** box. Then click **Submit Changes**.

To delete an RBL domain, enable the checkbox beside the domain you wish to delete and click **Submit Changes**. Note that deleting a master RBL domain also deletes *all* RBL rules that refer to the domain.

Note that the master RBL list is merely a list of all the RBL domains that CanIt-PRO can *potentially* use. To actually set up RBL rules, please see the User's Guide. Note that RBL rules can be created on a per-stream basis, so different streams can elect to use different RBLs from the master list.

5.3 Users

CanIt-PRO maintains its own table of users. You should enter users into this table to create CanIt-PRO administrative users, or users with different privileges from the default (for example, a demo user.) Click on "Administration" and then "Users" to set up users. You will see the user management screen.



Users (1 to 2 of 2)

Page: 1

User-ID	E-Mail	Write Access?	Delete?
admin@billw.com	admin@billw.com	☑	(Can't Delete)
test-user	test@nowhere.com	☑	□

Figure 5.3: Users

If you enter a string in the "Filter:" box, then CanIt-PRO limits the display to entries whose **User-ID** or **E-Mail** column contain that string.

5.3.1 User Privileges

When a user logs in to CanIt-PRO, he can see a single stream at a time. Every user always has access to a stream with the same name as his user name, and this is the default stream. The CanIt-PRO administrator can give users permission to see additional streams. For example, the user `janedoe` always has access to the stream `janedoe`. However, if she manages a mailing list called `joke-list`, you have two options:

1. You can stream messages for the list to `janedoe`, so she has only a single spam trap to consider.

2. You can create a new stream called `joke-list` and give access to that stream to `janedoe`. In this way, she can use different settings, blacklists and whitelists for the list than she does for her personal e-mail.

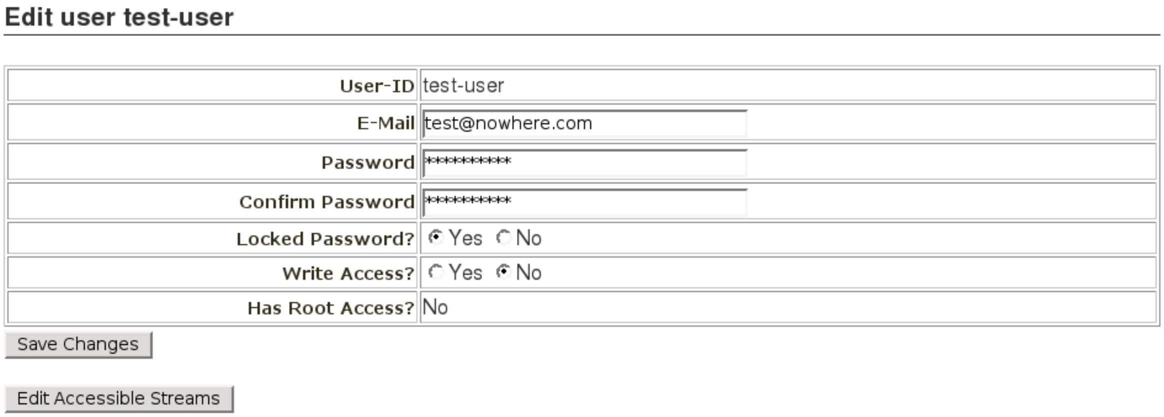
Each CanIt-PRO user has two special privileges, which can be on or off:

- A user with **root privilege** can add, edit and delete other users.
- A user with **write privilege** can mark messages as **spam** or **not-spam**, and can blacklist and whitelist hosts, domains and senders.

Note that CanIt-PRO allows for additional flexibility in controlling which parts of the Web interface are available to various users. For details, see Chapter 8.

5.3.2 Editing a User

To edit a user, click on the User-ID on the user management screen. You will see the user-editing screen.



Edit user test-user

User-ID	test-user
E-Mail	test@nowhere.com
Password	***********
Confirm Password	***********
Locked Password?	● Yes ○ No
Write Access?	○ Yes ● No
Has Root Access?	No

Figure 5.4: User-Editing

- To set the user's e-mail address, enter it in the **E-Mail** field.
- If you wish to change the user's password, enter it in the **Password** and **Confirm Password** fields. If you leave these fields blank, the password will not be changed.
- If you set **Locked Password?** to **Yes**, then the user will have a "locked" password and will not be able to log in. However, if you have configured an alternate user authentication method, the user will be able to log in using a password that the alternate method accepts.
- Adjust the write-access privilege by setting the **Write-Access?** checkbox appropriately. Note that you *cannot* grant or revoke **root** privileges by editing a user; **root** privileges are given or denied at user-creation time.

To make the changes take effect, click **Submit Changes**.

Note: Both user-names and passwords are *case-sensitive*; a used named `user1` is completely different from one named `User1`.

5.3.3 Adding a User

To add a user, click on the “Add User” link. A form similar to the one used to edit users will appear. Fill it in and click **Submit Changes** to add the user.

Note: Both user-names and passwords are *case-sensitive*; a used named `user1` is completely different from one named `User1`.

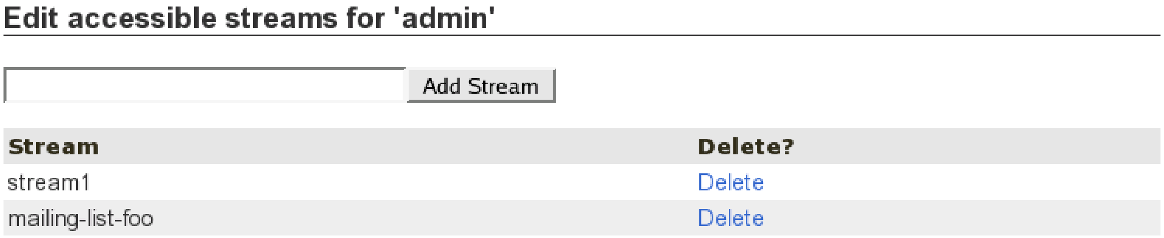
5.3.4 Deleting a User

If there is more than one user, a “Delete” checkbox appears beside those users that can be deleted. Enable the checkbox and then click **Submit Changes** to delete the selected user or users. Note that it is not possible to undo the deletion!

Note that if you delete a user, he may still have access if he can be authenticated using an external authentication mechanism.

5.3.5 Granting Access to Streams

If you wish to grant a user access to additional streams, click on the “Edit Accessible Streams” button (Figure 5.4). The stream editing page will appear:



Edit accessible streams for 'admin'

Stream	Delete?
stream1	Delete
mailing-list-foo	Delete

Figure 5.5: Granting Access to Streams

To grant access to a stream, enter the stream name in the input box and click **Add Stream**. To revoke access to a stream, click on the “Delete” link next to the stream name.

Note that a user *always* has access to a stream with the same name as his user name, and this access cannot be revoked. Also, the CanIt-PRO administrator can access any stream, regardless of the settings on this page.

5.4 Permitting Users to Opt In

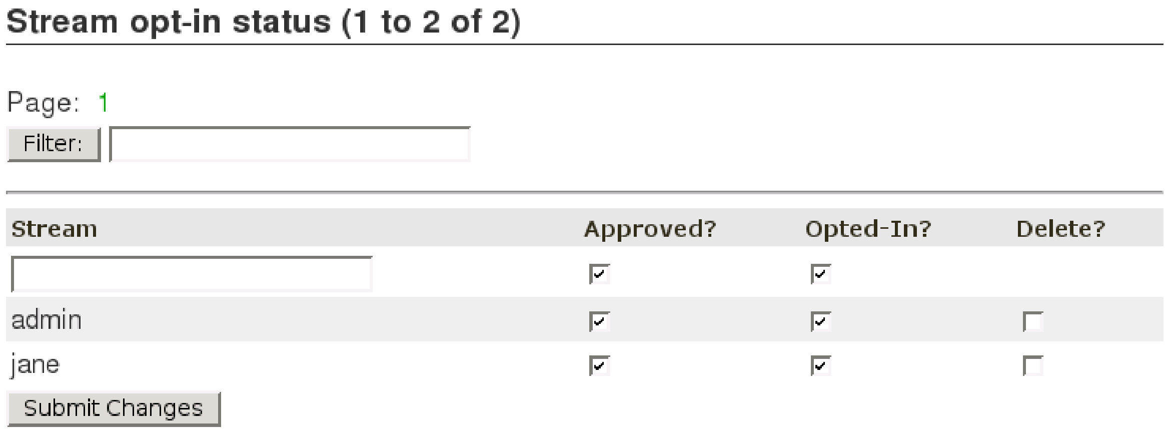
In the CanIt-PRO global settings (Section 5.1), the CanIt-PRO administrator can control:

- Whether or not people are permitted to opt-in to spam scanning.
- Whether the default setting is opt-in or opt-out.

There are three useful combinations:

1. Permit everyone to opt-in, and have the default be opt-in.
2. Permit everyone to opt-in, and have the default be opt-out.
3. Permit only selected people to opt-in, and have the default be opt-out.

In the first two cases, the administrator need not do anything special. In the third case, you must add entries to the Stream Approval Table. Click on "Administration" and then "Opt Others In/Out" to see this table:



Stream opt-in status (1 to 2 of 2)

Page: 1

Stream	Approved?	Opted-In?	Delete?
	☑	☑	
admin	☑	☑	□
jane	☑	☑	□

Figure 5.6: Stream Opt-In Approval

If the "Approved?" column is checked, then the stream may opt in to spam scanning. If it is not checked, then the stream may not opt in to spam scanning.

If the "Opted-In?" column is checked, the stream is currently opted in to spam scanning. Otherwise, it is not.

To add a stream to the table, enter the stream name in the input box and set "Approved?" and "Opted-In?" appropriately. Then click **Submit Changes**.

To edit existing streams, adjust "Approved?" and "Opted-In?" appropriately and click **Submit Changes**. To delete a stream from the opt-in table, enable the **Delete?** checkbox on the appropriate row and click **Submit Changes**.

If the default setting is to permit anyone to opt in to spam scanning, you can nevertheless exclude particular streams from being able to opt in by entering them in the Stream Approval Table and turning off the “Approved?” checkbox.

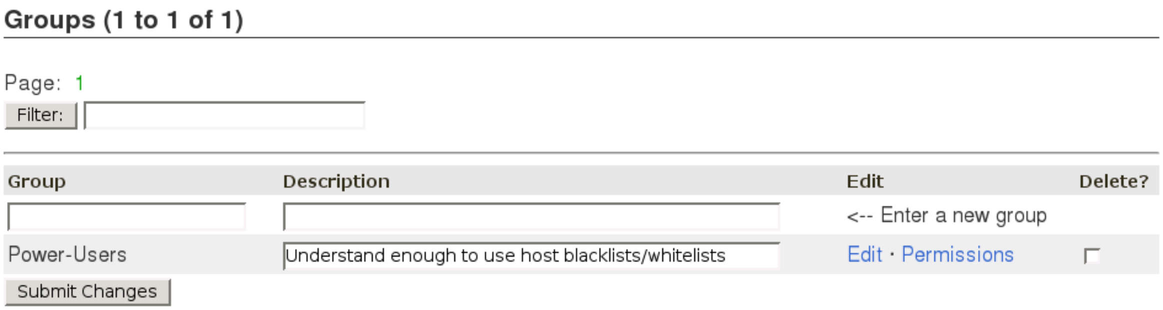
In order for spam-scanning to occur, a stream must be both approved and opted-in. If the stream is not found in the Stream Approval Table, then the defaults are taken from the Global Settings.

If you enter a string in the “Filter:” box, then CanIt-PRO limits the display to entries whose **Stream** column contains that string.

5.5 Groups

For the purpose of granting permissions, CanIt-PRO allows you to create *groups*. A group is simply a collection of users.

To edit groups, click on “Administration” and then “Groups”. The Groups Page appears:



Groups (1 to 1 of 1)

Page: 1

Filter: [ ]

Group	Description	Edit	Delete?
[ ]	[ ]	<-- Enter a new group	
Power-Users	Understand enough to use host blacklists/whitelists	Edit · Permissions	□

Submit Changes

Figure 5.7: Groups

5.5.1 Creating, Deleting and Editing Groups

To create a new group:

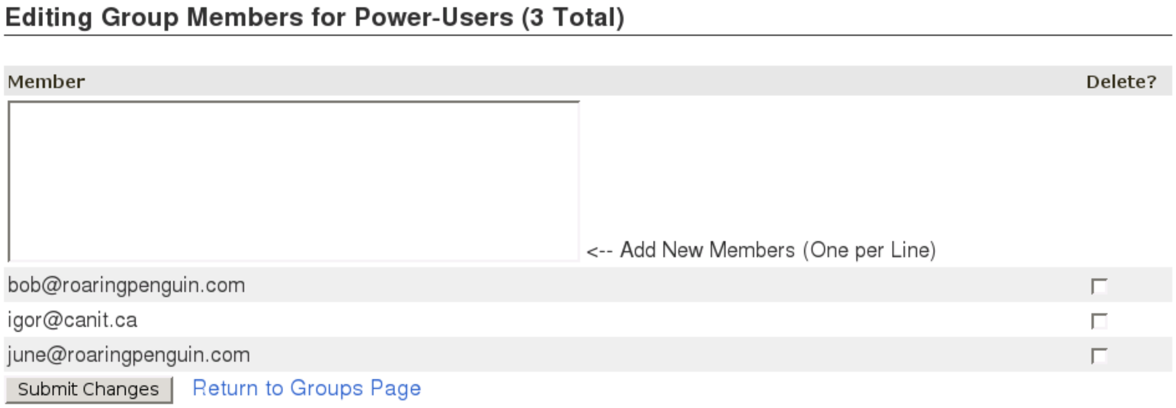
1. Enter the name of the group in the **Group** box.
2. Enter a description of the group in the **Description** box.
3. Click **Submit Changes**

To delete an existing group:

1. Enable the **Delete** checkbox for the group you want to delete.
2. Click **Submit Changes**

To edit a group:

1. Click on the “Edit” link next to the appropriate group. The Group Members page appears:



Editing Group Members for Power-Users (3 Total)

Member	Delete?
<-- Add New Members (One per Line)	
bob@roaringpenguin.com	☐
igor@canit.ca	☐
june@roaringpenguin.com	☐

Figure 5.8: Group Members

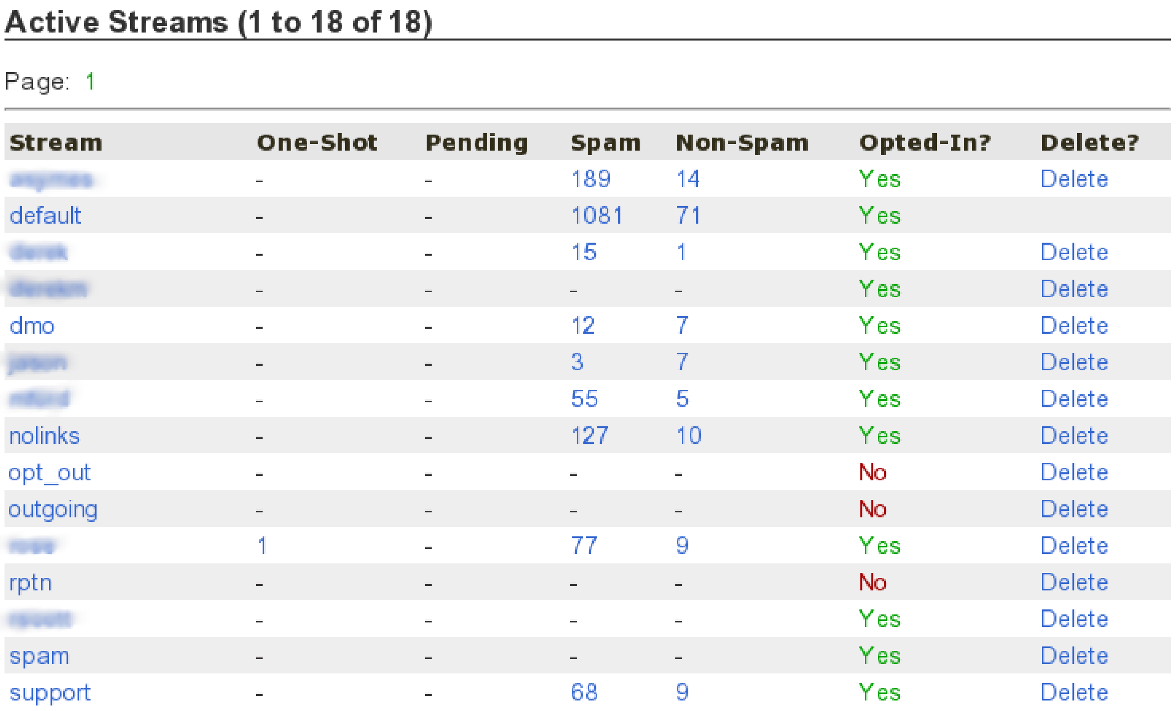
2. Enter new members (one per line) in the **Member** text area.
3. If you want to delete existing members, enable the appropriate **Delete** checkbox.
4. Click **Submit Changes**

Note: External authentication methods can affect group membership. See Chapter 6 for details.

In the Groups Page, click on “Permissions” to edit the permissions associated with the group. Permissions will be discussed in detail in Chapter 8.

5.6 Viewing Active Streams

The CanIt-PRO administrator can look at all the streams with entries in the incidents table. To do this, select “Administration” and then “See Active Streams”. The Active Streams Page appears:



Active Streams (1 to 18 of 18)

Page: 1

Stream	One-Shot	Pending	Spam	Non-Spam	Opted-In?	Delete?
[redacted]	-	-	189	14	Yes	Delete
default	-	-	1081	71	Yes	
[redacted]	-	-	15	1	Yes	Delete
[redacted]	-	-	-	-	Yes	Delete
dmo	-	-	12	7	Yes	Delete
[redacted]	-	-	3	7	Yes	Delete
[redacted]	-	-	55	5	Yes	Delete
nolinks	-	-	127	10	Yes	Delete
opt_out	-	-	-	-	No	Delete
outgoing	-	-	-	-	No	Delete
[redacted]	1	-	77	9	Yes	Delete
rptn	-	-	-	-	No	Delete
[redacted]	-	-	-	-	Yes	Delete
spam	-	-	-	-	Yes	Delete
support	-	-	68	9	Yes	Delete

Figure 5.9: Active Streams

A stream is considered “active” if it has at least one message in the trap (new, pending, spam or non-spam).

The columns in the display are:

Stream The name of the stream. Each stream name is a hyperlink; if you click on the link, you will switch streams to that stream.

One-Shot The number of one-shot messages in the stream’s trap.

Pending The number of pending messages in the stream’s trap.

Spam The number of spam messages in the stream’s trap.

Non-Spam The number of non-spam messages in the stream’s trap.

Opted-In? Set to **Yes** if the stream is both approved for anti-spam scanning and opted-in; set to **No** otherwise.

Delete A column of links for deleting streams.

If you enter a string in the “Filter:” box, then CanIt-PRO limits the display to entries whose **Stream** column contains that string.

5.6.1 Deleting a Stream

To delete a stream, click on the “Delete” link in the Active Streams page. Then click on “Yes, delete it!” to confirm deletion.

5.7 Filtering Outgoing Mail

Some organizations like to add boilerplate disclaimers to outgoing mail. CanIt-PRO can achieve this by streaming all outgoing mail to an “outgoing” stream, and adding boilerplate options for that stream.

To stream all outgoing mail to a particular stream, set up your domain mappings as follows:

- All of your own domains (that is, domains considered “internal”) should have mappings set up. The mappings could be **ChopDomain**, **Sendmail**, or whatever, as long as the mappings exist.
- The wild-card domain * should have a domain mapping of **Database**.
- The wild-card address * should have an address mapping mapping it to the stream **outgoing**. (You can name your outgoing stream however you like.)

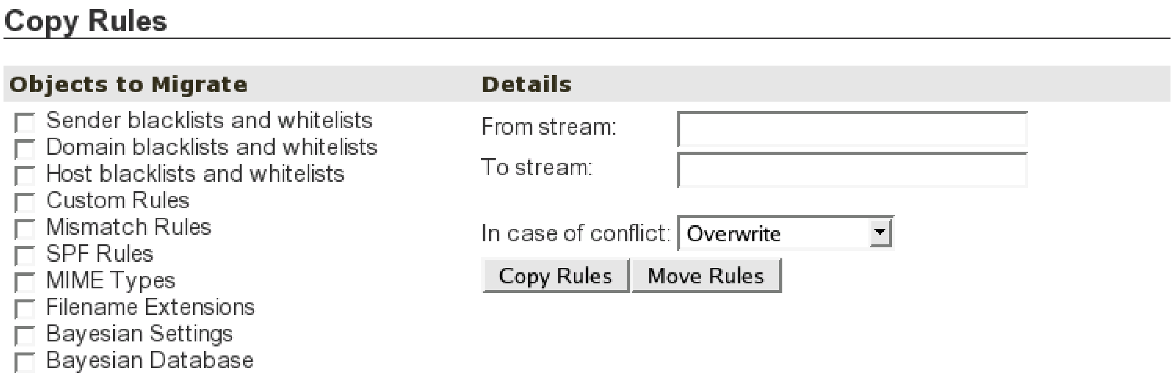
With these settings, mail for internal recipients will be streamed appropriately, and mail for external recipients will all be streamed to **outgoing**.

For the **outgoing** stream, enter the appropriate boilerplate to add to outgoing messages. You can also add custom body-matching rules if you want to trap mail containing certain words—for example, “Do Not Distribute Externally” Such rules on an **outgoing** stream may help prevent unauthorized distribution of confidential information.

See also Section 4.5 on page 39 for another way to force outgoing mail into a specific stream.

5.8 Copying Rules from One Stream to Another

Occasionally, it is useful to copy or move rules from one stream to another. To do this, click on “Administration” and then “Copy Rules”. The Copy Rules page appears:



Copy Rules

Objects to Migrate

□ Sender blacklists and whitelists
□ Domain blacklists and whitelists
□ Host blacklists and whitelists
□ Custom Rules
□ Mismatch Rules
□ SPF Rules
□ MIME Types
□ Filename Extensions
□ Bayesian Settings
□ Bayesian Database

Details

From stream:

To stream:

In case of conflict: Overwrite

Figure 5.10: Copying Rules

To copy rules:

1. Choose which rules you wish to copy by activating the appropriate check boxes under **Objects to Migrate**.
2. Put the name of the stream you want to copy *from* in the **From stream:** box.
3. Put the name of the stream you want to copy *to* in the **To stream:** box.
4. Select “Preserve Original” or “Overwrite” to handle the case of conflicting rules in the source and destination streams.
5. Click on **Copy Objects** to copy rules from the source stream to the destination stream. **Move Objects** is similar, but any rule that is successfully placed in the destination stream is deleted from the source stream.

Note: The rules for copying the Bayesian Database are different from other objects:

- Copying the Bayes data from one stream to another *always* overwrites the destination stream’s Bayes data.
- Moving Bayes data from one stream to another does *not* clear out the original stream’s Bayes data.
- Copying a large number of tokens can be very slow – be patient.

5.9 Secondary MX Hosts

Secondary MX hosts require special handling by CanIt-PRO. If e-mail is received from an IP address in the list of secondary MX hosts, CanIt-PRO modifies its behaviour as follows. Note that localhost (127.0.0.1) is *always* considered a secondary MX host for the purposes below:

- Suspect mail is not responded to with an SMTP temporary-failure code; instead, it is held locally in the CanIt-PRO database. (However, if you set the **Send tempfail indications for suspect messages** setting to **Always**, then CanIt-PRO will tempfail mail from secondary MX machines.)
- CanIt-PRO ignores any host-blacklists or host-whitelists for the secondary MX host. However, as a special exception, you can whitelist the local host (127.0.0.1). This whitelists locally-generated mail. For held-and-remailed messages, CanIt-PRO uses a special technique to remember the true IP address of the originating host, so whitelisting 127.0.0.1 is safe. It will not automatically whitelist held-and-remailed messages.
- Any mismatch rules are ignored.
- Real-time DNS blacklist lookups are suppressed.

- First-time sender checks are suppressed.
- Custom rules based on the “Relay” or “HELO” field are not evaluated.

Note that any machine under your control that you expect to forward mail to your machine should be considered a secondary MX host. For example, if a number of users have accounts on a machine that forward mail to your machine using `.forward` files, you should consider entering that machine as a secondary MX host.

Also, note that if CanIt-PRO is able to determine the “real” relay IP by parsing the **Received:** headers, and you have enabled this option, then CanIt-PRO runs all the host checks as usual, using the real relay IP address. However, these checks are (of necessity) delayed until after the DATA phase of the SMTP transaction, because CanIt-PRO does not have the required information at the MAIL FROM: or RCPT TO: phases.

5.10 Avoiding Backscatter

Under most circumstances, if CanIt-PRO rejects a message, it responds with an SMTP failure code. This generally causes the sending relay to mail a failure notification to the original sender.

However, because most spam and viruses have faked sender addresses, you may not want this behavior for messages relayed from a secondary MX host or, in the case of CanIt-PRO, for messages split into multiple streams. That’s because if a message is rejected after having been accepted by one of your mail servers, it’s the responsibility of the sending server to generate a failure Delivery Status Notification or DSN.

If (as is likely) the sender address is faked, that failure message may arrive at an unsuspecting third-party. This is what is known as *backscatter*.

It is a violation of RFC 821, and is generally considered bad behavior, to silently discard mail; however, many sites are beginning to lump hosts responsible for generating backscatter into the same category as spammers. Because of this, we now recommend leaving the **Avoid generating DSNs for rejected mail from our MX hosts** set to **Yes**, so that bounce messages are not generated.

Chapter 6

External Authentication

6.1 Introduction

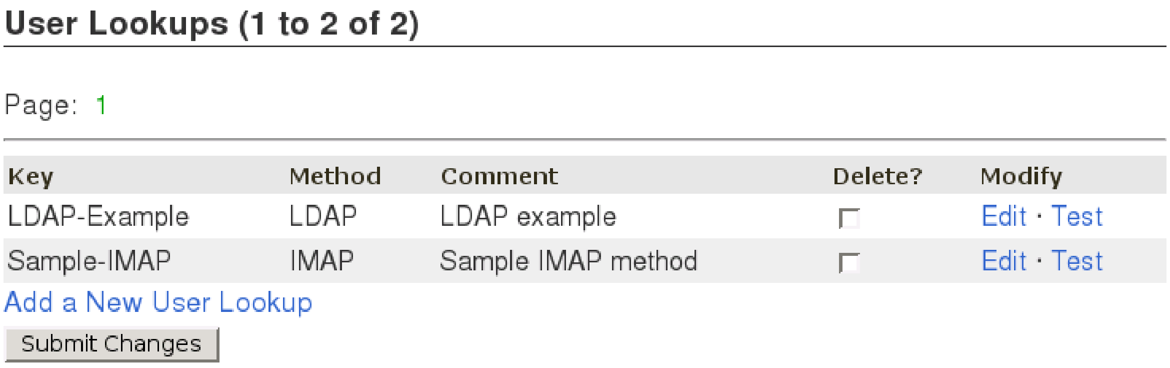
In addition to its built-in user list, CanIt-PRO can authenticate users using external mechanisms. To enable the use of external authentication mechanisms, these basic steps must be followed:

1. A *User Lookup* must be defined. A User Lookup describes to CanIt-PRO how to look up user information from an external source.
2. An *Authentication Mapping* must be created. An Authentication Mapping tells CanIt-PRO which User Lookup to user for a given domain. You can use different authentication mechanisms for different domains, which gives CanIt-PRO considerable flexibility.

6.2 User Lookups

To create a User Lookup:

- Click on “Setup” and then “User Lookups”. You will see the User Lookup list:



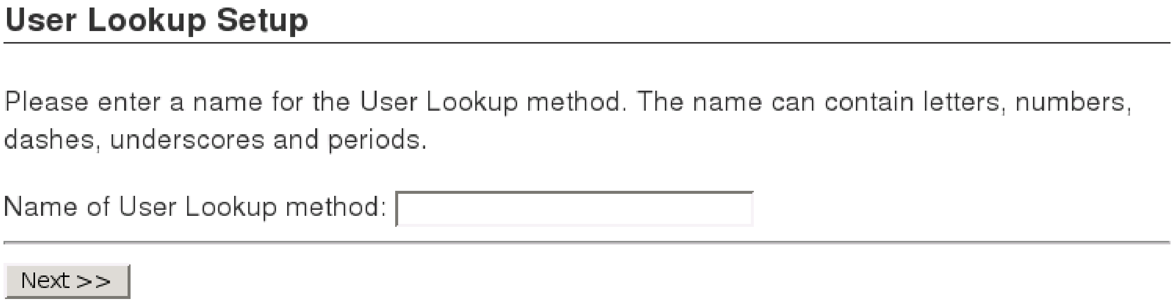
User Lookups (1 to 2 of 2)

Page: 1

Key	Method	Comment	Delete?	Modify
LDAP-Example	LDAP	LDAP example	□	Edit · Test
Sample-IMAP	IMAP	Sample IMAP method	□	Edit · Test

Figure 6.1: User Lookup List

- Click on “Add a New User Lookup”, and the User Lookup Wizard appears:



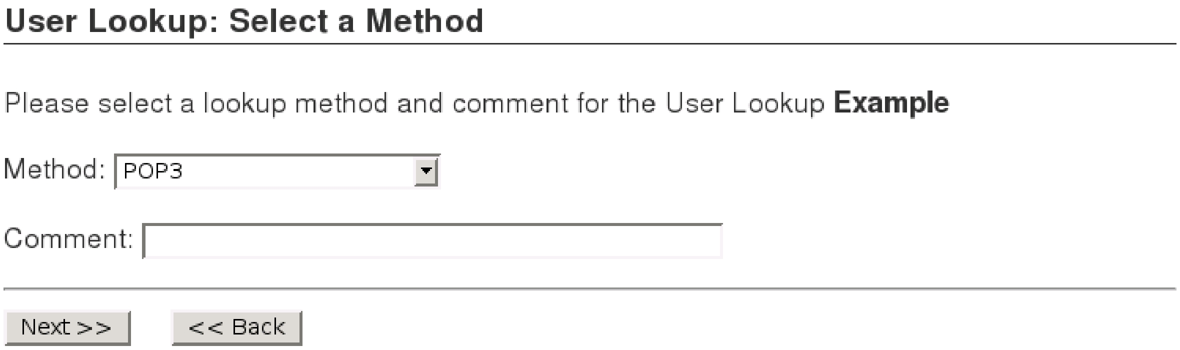
User Lookup Setup

Please enter a name for the User Lookup method. The name can contain letters, numbers, dashes, underscores and periods.

Name of User Lookup method:

Figure 6.2: User Lookup Wizard

- Pick a name for the User Lookup, and click “Next”. The User Lookup method selection screen appears:



User Lookup: Select a Method

Please select a lookup method and comment for the User Lookup **Example**

Method: POP3

Comment:

Figure 6.3: User Lookup: Method Selection

- Enter a comment for the lookup method. The comment can be anything you like; its purpose is to document the method so you remember what it does.

- Select a lookup method. CanIt-PRO supports the following methods:

  - **POP3**: CanIt-PRO authenticates users against a POP3 server.
  - **IMAP**: CanIt-PRO authenticates users against an IMAP server.
  - **LDAP (Generic)**: CanIt-PRO authenticates users against an LDAP server.
  - **LDAP (Active Directory)**: This is very similar to generic LDAP, but default values in the User Lookup creation wizard are more suitable for Active Directory lookups. Note that if you authenticate against Active Directory, you must turn off the “Enhanced Security Option” of Active Directory, because its non-standard Kerberos implementation does not interoperate with CanIt-PRO’s LDAP libraries.
  - **Program**: CanIt-PRO invokes a program (that you supply) to perform authentication.

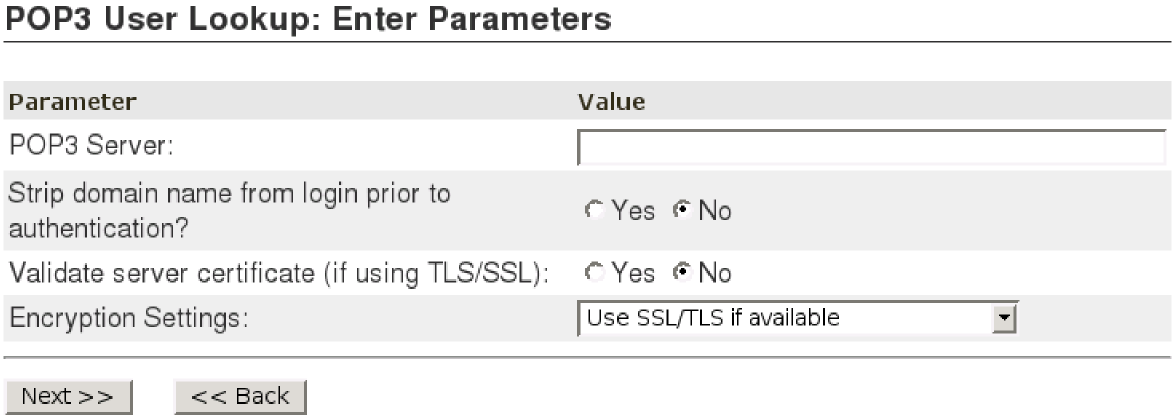
  - **Program (Legacy method)**: CanIt-PRO invokes external programs in the same way as older versions did (using the “Alternate Authentication” global setting that has since been removed.)

  Note also that the LDAP and Program methods can be used for streaming as well as authentication; details will be given in the following sections.

- Click “Next’.

6.2.1 IMAP and POP3 Authentication

If you selected IMAP or POP3 authentication methods, then the wizard looks like this:



POP3 User Lookup: Enter Parameters

Parameter	Value
POP3 Server:	
Strip domain name from login prior to authentication?	○ Yes ● No
Validate server certificate (if using TLS/SSL):	○ Yes ● No
Encryption Settings:	Use SSL/TLS if available

Figure 6.4: IMAP/POP3 User Lookup

To complete the setup:

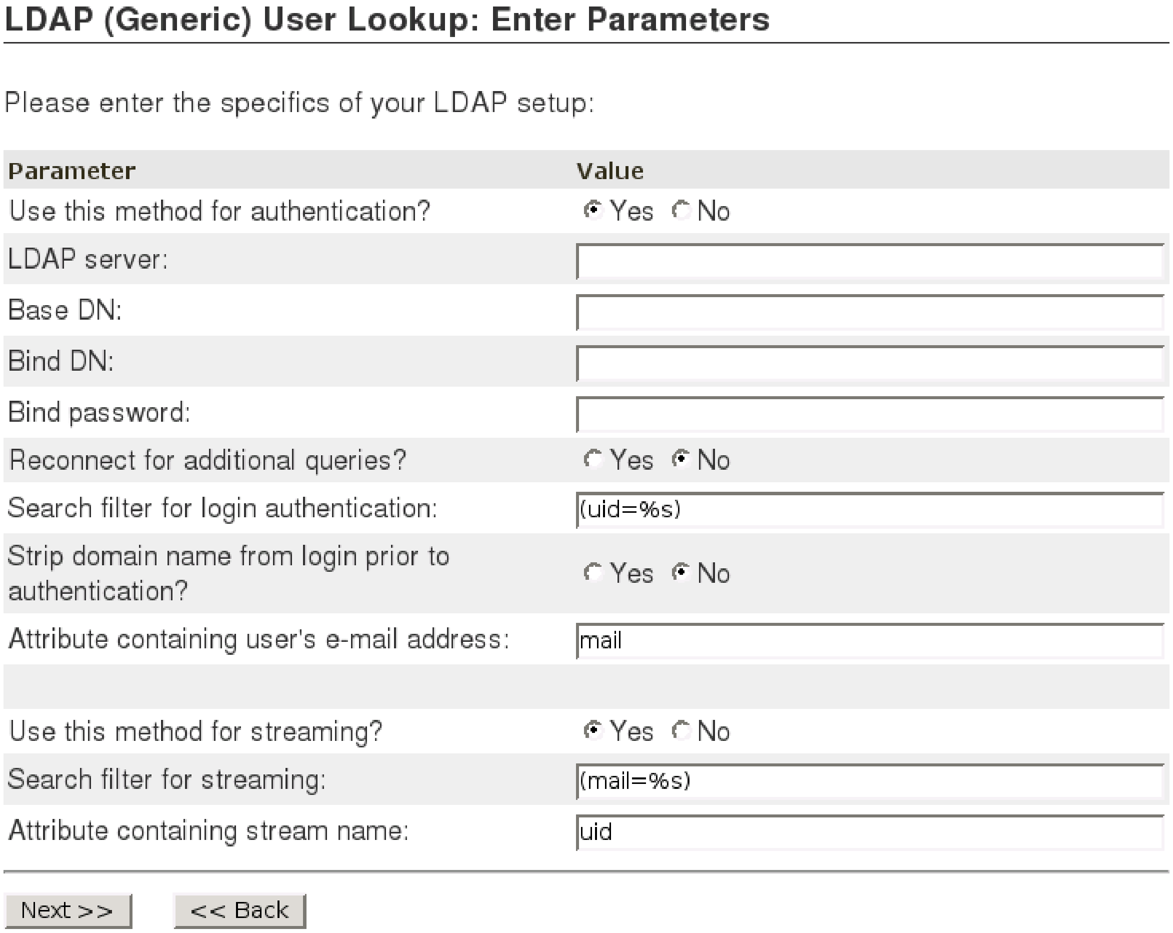
- Enter the IP address or host name of the IMAP or POP3 server. If the server is listening on a non-standard port, add a slash followed by the port number to the server name. For example, if you have an IMAP server listening on port 1143 on the host `magnesium`, you could enter `magnesium/1143` as the server.
- If you would like to strip the domain name from the login name before attempting authentication, set the “Strip domain name” setting to **Yes**. If someone logs in to CanIt-PRO as **`user@domain.net`** and this setting is **Yes**, then the username passed to the IMAP or POP3 server is simply **`user`**.
- If you want CanIt-PRO to validate the SSL certificate of the server (assuming SSL or TLS is used), set “Validate server certificate” to **Yes**.
- Pick the appropriate encryption settings for CanIt-PRO to use when communicating with the POP3 or IMAP server.
- Click “Next” to see a summmary of your settings.
- If all of the settings are correct, click “Finish” to create the POP3 or IMAP User Lookup.

6.2.2 LDAP Authentication

There are two types of LDAP user lookups possible within CanIt: LDAP (Generic) and LDAP (Active Directory). Both of these methods are very similar; the only difference is the defaults that are offered by the User Lookup Wizard when you create the User Lookup.

LDAP user lookups can be used for one or both of user authentication and stream mapping. When used for stream mapping, the LDAP lookup method will also validate incoming email addresses against the LDAP server, allowing rejection of invalid recipients immediately at the CanIt gateway.

If you select one of the LDAP methods, you will see the LDAP User Lookup Wizard:



LDAP (Generic) User Lookup: Enter Parameters

Please enter the specifics of your LDAP setup:

Parameter	Value
Use this method for authentication?	● Yes ○ No
LDAP server:	
Base DN:	
Bind DN:	
Bind password:	
Reconnect for additional queries?	○ Yes ● No
Search filter for login authentication:	(uid=%s)
Strip domain name from login prior to authentication?	○ Yes ● No
Attribute containing user's e-mail address:	mail
Use this method for streaming?	● Yes ○ No
Search filter for streaming:	(mail=%s)
Attribute containing stream name:	uid

Figure 6.5: LDAP User Lookup

To complete the setup:

- If you wish to use this User Lookup for authentication, set “Use this method for authentication?” to **Yes**.
- In the “LDAP server(s)” box, enter the IP address or host name of your LDAP server. You can enter a comma-separated list of servers if you have more than one LDAP server. As with the

IMAP and POP3 User Lookups, if a server listens on a non-standard port, enter a slash followed by the port number after the server name. For example, if you have two LDAP servers **serverA** and **serverB**, and the second listens on non-standard port 3389, enter the following into the server box:

```
serverA, serverB/3389
```

- Normally, CanIt-PRO tries the LDAP servers in order. If you would like it to try them in a random order (for load-balancing), set "Load-balance LDAP servers" to **Yes**.

- Enter the Base DN of your LDAP tree in the "Base DN" box.

- Typically, CanIt-PRO needs to bind to the LDAP directory before it can search it. Enter the Bind DN in the "Bind DN" box. If a password is required, enter it in the "Bind password" box.

- Some LDAP servers require CanIt-PRO to disconnect and reconnect and re-bind between queries. (Active Directory requires this.) If your LDAP server requires this, set the "Reconnect" setting to **Yes**.

- Enter the search filter for login authentication. The string `%s` will be replaced by the user's login name. For most UNIX LDAP servers, a search filter of `(uid=%s)` is appropriate. For Active Directory, it might be `(sAMAccountName=%s)`.

- If you would like to strip the domain name from the login name before attempting authentication, set the "Strip domain name" setting to **Yes**. If someone logs in to CanIt-PRO as **`user@domain.net`** and this setting is **Yes**, then the username passed to the LDAP server is simply **`user`**.

- To use the Locked Addresses feature, CanIt-PRO needs to know the e-mail address of a logged-in user. In most UNIX LDAP servers, this is stored in the `mail` attribute, while in many Active Directory servers, this is stored in the attribute `proxyAddresses`.

- If you wish to control group membership using LDAP, enter the name of an LDAP attribute in the **Attribute containing group names** box. This attribute should contain a comma-separated list of group names. When a user authenticates, he/she will be considered to be a member of all of the groups listed in this attribute.

- If you wish to use the LDAP server to stream addresses as well as authenticate, set "Use this method for streaming" to **Yes**.

- For streaming, CanIt-PRO needs to look up an e-mail address in the LDAP server. For most UNIX servers, the appropriate search filter is `(mail=%s)`, while for Active Directory, it is probably `(proxyAddresses=smtp:%s)`. In the search filter, the string `%s` is replaced with the e-mail address. `%u` is replaced with the local part of the e-mail address (everything before '@') and `%d` is replaced with the domain part of the address (everything after the '@'.)

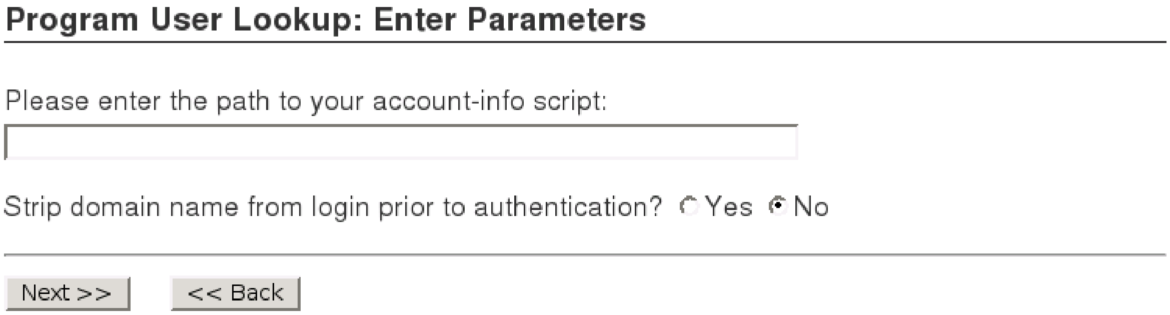
- CanIt-PRO needs to know which LDAP attribute contains the stream name. For most UNIX servers, the appropriate attribute is `uid`, while for Active Directory, it is probably `sAMAccountName`.

- If you would like CanIt-PRO to force stream names (as determined by the LDAP lookup) to lower-case, set “Force stream name to lower-case?” to **Yes**. (This is the default.) If you want to preserve mixed-case stream names, set this setting to **No**.
- You can change the connect timeout from the default value of 120 seconds to any value from 2 to 120 seconds. Note that this timeout *only* applies to streaming lookups by the Perl filters. It does not apply to authentication, because PHP (used for the Web interface) does not have a way to specify an LDAP connect timeout.

Once you have entered the LDAP parameters, click “Next” to review your entries, and “Finish” to create the User Lookup.

6.2.3 Program Authentication

With the Program User Lookup method, CanIt-PRO invokes an external program to authenticate users and map addresses to streams. If you select **Program** as your User Lookup type, the Program User Lookup Wizard appears:



Program User Lookup: Enter Parameters

Please enter the path to your account-info script:

Strip domain name from login prior to authentication? ○ Yes ● No

Figure 6.6: Program User Lookup

To configure the Program User Lookup:

- Enter the full path to your “account-info” script. This is an executable script or program that you must supply. The path you supply must be an absolute path name. If you are running a CanIt-PRO cluster, this script must exist (and be identical!) on all scanning servers and the Web server.
- If you would like to strip the domain name from the login name before attempting authentication, set the “Strip domain name” setting to **Yes**. If someone logs in to CanIt-PRO as **`user@domain.net`** and this setting is **Yes**, then the username passed to the program is simply **`user`**.

How the Program User Lookup is Invoked

- For *authentication*, the program is invoked as follows:

`/path/to/script --authenticate`

The program is then expected to read two lines from its standard input: The first line is a login name, and the second line is a password. The program must then validate the login name and password, and exit with one of the following exit codes:

- **0** — Authentication was successful.
- **1** — Authentication failed.

- For *obtaining user information*, the program is invoked as follows:

  `/path/to/script --info username`

  Here, the program is passed the successfully logged-on user name as a command-line argument. It should print a series of key=value lines to its standard output, and exit with an exit status of 0. (Note that the script doesn't *have* to produce any output, but it *can* produce output if you want to pass extra information to CanIt-PRO.)

  The key/value pairs currently used by CanIt-PRO are:

  - `home_stream=stream-name` — sets the user's home stream to *stream-name* instead of her login name. One possible use could be to convert a login name to all lower-case on systems that permit case-insensitive authentication. This ensures that no matter how the person logs in, she is directed to the correct stream name.
  - `groups=group1,group2,...,groupN` — when the user logs in, add her to all of the groups listed in the comma-separated list.
  - `mail=email-address` — set the user's e-mail address to *email-address*.

- For *mapping an e-mail address to a stream*, the program is invoked as follows:

  `/path/to/script --info-email address`

  Here, *address* is an e-mail address that must be streamed. The script should write key=value lines to its standard output, and exit with one of the following exit codes:

  - **0** — the address exists and was successfully streamed.
  - **1** — there was a temporary failure streaming the address. The mail will be tempfailed.
  - **67** — the address is not valid. CanIt-PRO will fail the SMTP RCPT command with a "User unknown" failure code.

  If the address was streamed successfully, the script must print the following line to standard output:

  `stream=stream-name`

  This causes *address* to be mapped to *stream-name*. If no `stream=stream-name` line is emitted, but the script exits with a zero status, then CanIt-PRO falls back to database lookups, as described in Section 2.5 on page 22.

Sample Program for the Program User Lookup Method

The following is a very simple Bourne shell script illustrating how the Program User Lookup method works. Real scripts would obviously be more complex and probably written in a more appropriate language like Perl.

```
#!/bin/sh
do_auth () {
    read user
    read pass
    # In reality, we would do a directory lookup against LDAP or similar
    if test "$user" = "foo" -a "$pass" = "bar" ; then
        exit 0
    fi
    exit 1
}

do_info () {
    user="$1"
    # In reality, we would do a directory lookup against LDAP or similar
    if test "$user" = "foo" ; then
        echo "home_stream=foobar";
        echo "mail=foo@roaringpenguin.com";
    fi
    exit 0
}

do_info_email () {
    email="$1"
    # In reality, we would do a directory lookup against LDAP or similar
    if test "$email" = "foo@roaringpenguin.com" ; then
        echo "stream=foobar-stream";
    fi
    if test "$email" = "nouser@roaringpenguin.com" ; then
        # No such user
        exit 67
    fi
    exit 0
}

# Main program
case "$1" in
    --authenticate)
        do_auth
        ;;
    --info)
        do_info "$2"
        ;;
    --info-email)
        do_info_email "$2"
        ;;
    *)
        exit 1;
        ;;
esac
```

6.2.4 Program Authentication (Legacy Method)

If you select this User Lookup method, then CanIt-PRO falls back to behavior compatible with previous versions:

- If a program called `/etc/mail/canit/account-info` exists and is executable, CanIt-PRO invokes it as if it were the script supplied for a **Program** User Lookup method.
- Otherwise, CanIt-PRO invokes `/etc/mail/canit/authenticate-user` to authenticate users and `/etc/mail/canit/address-to-stream` to convert an e-mail address to a stream. These scripts have been in use since CanIt-PRO 2.0 and are deprecated; you should convert to the new **Program** User Lookup method.

6.2.5 The `account-info` Script

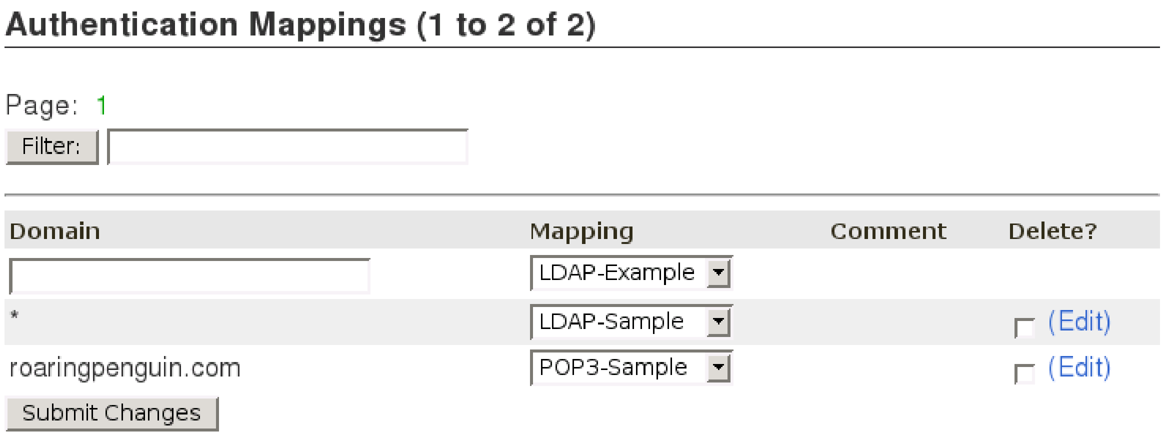
Some User Lookup methods (such as POP3 or IMAP) as well as a lookup in the built-in user database are not capable of passing extra information back to CanIt-PRO. For that reason, if any User Lookup method other than Program or LDAP is used, CanIt-PRO still attempts to execute:

```
/etc/mail/canit/account-info --info username
```

to obtain extra attributes (`mail`, `groups` and `home_stream`) after a user logs in. If you need to set users' e-mail addresses or home streams, but have them authenticate against an IMAP or POP3 server, simply supply an appropriate `account-info` script.

6.3 Authentication Mappings

Once you have set up your User Lookup methods, you need to tell CanIt-PRO which method to invoke for each domain. To do this, click on "Setup" and then "Authentication Mappings". The Authentication Mappings page appears:



Authentication Mappings (1 to 2 of 2)

Page: 1

Domain	Mapping	Comment	Delete?
	LDAP-Example		
*	LDAP-Sample		□ (Edit)
roaringpenguin.com	POP3-Sample		□ (Edit)

Figure 6.7: Authentication Mappings

To create a new authentication mapping:

1. Enter the domain name in the **Domain** field.
2. Select the User Lookup from the **Mapping** field.
3. Click on **Submit Changes**

In Figure 6.7, we see that anyone who logs in as *`user`*`@roaringpenguin.com` will be authenticated using the `POP3-Sample` User Lookup. Anyone logging in with a different domain (or no domain at all—simply *`user`*) will be authenticated using the `LDAP-Sample` User Lookup.

If you enter a string in the “Filter:” box, then CanIt-PRO limits the display to entries whose **Domain** or **Mapping** columns contain that string.

Chapter 7

Bayesian Filtering

7.1 Introduction to Bayesian Filtering

Bayesian filtering is a statistical technique whereby CanIt-PRO assigns a *spam probability* based on training from users. Bayesian filtering can greatly improve the accuracy of CanIt-PRO, and makes it harder for spammers to evade filtering.

Please consult the CanIt-PRO User's Guide for additional details on using Bayesian filtering. This guide only contains information relevant when setting up and administering CanIt-PRO.

7.2 Unauthenticated Voting

Normally, to vote if a message is spam or not spam, a user must log in. You can configure CanIt-PRO to permit unauthenticated voting; this can make life easier for end-users who can just click on a link without worrying about entering a user name and password.

Note: Think carefully about permitting unauthenticated voting. If voting links ever escape your organization (as part of a forwarded message, for example), and your CanIt-PRO Web interface is externally accessible, outsiders can cast votes. We strongly recommend permitting unauthenticated voting only if access the the CanIt-PRO Web interface is controlled in some other way.

To permit unauthenticated voting, you need to make the following change:

- Under "Global Settings", set **Permit unauthenticated voting** to **Yes**

7.3 The Bayes Journal

Bayesian training can be slow because it involves many database updates. For that reason, when you train a message, CanIt-PRO simply makes a note of the fact that the message is to be trained in a special table called the *Bayes Journal*. Periodically, the CanIt-PRO ticker process goes through the Bayes Journal and actually updates the Bayes data.

For this reason, if you train some messages, these results will not immediately appear in the Bayes Settings page. The Bayes Journal is run every 10 minutes or so, so your training should appear within 10-15 minutes.

7.4 RPTN

RPTN stands for the Roaring Penguin Training Network, and is a mechanism whereby multiple CanIt-PRO installations can share Bayes votes. RPTN contains two main parts:

1. In the *reporting* phase, CanIt-PRO installations send reports about whether or not mail they have seen is spam. A report essentially consists of a list of tokens in the mail message and a **spam** or **not-spam** flag, depending on how the incident was disposed of.

   The RPTN server aggregates all of the reports it receives and builds a database of Bayesian statistics from the reports.

2. In the *download* phase, a CanIt-PRO installation downloads the aggregated data and installs it in its database. This data can subsequently be used for Bayesian analysis.

To set up RPTN, click on **Setup** and then **Wizards**. Choose the **RPTN Setup Wizard**. The wizard leads you through the following steps:

1. You are asked if you would like to download Bayes data from RPTN.
2. If you answered **Yes** in Step 1, you are given an opportunity to limit when RPTN data is downloaded. Downloading RPTN data can place a fair amount of load on the server, so you should limit RPTN downloads to off-peak hours. Be sure to leave at least a four-hour download window, because RPTN checks are made every two hours. If the download window is too short, you may miss a download.
3. You are asked if you would like to submit reports to RPTN.
4. If you answered **Yes** in steps 1 or 3, you are prompted for your download username and password. You cannot submit RPTN reports or download RPTN data unless you supply a valid username and password.
5. Your settings are summarized, and you are prompted to click **Finish** to save the changes.

RPTN data are downloaded into a stream called `@@RPTN`. If you would like to use RPTN data in Bayesian analyis, you must include `@@RPTN` in the stream setting “Inherit Bayes training history from these streams”. If you want all streams to inherit Bayes data from `@@RPTN`, then set the “Inherit Bayes training history from these streams” setting in the `default` stream.

Note: To download RPTN data, the CanIt-PRO server must be able to make outgoing HTTPS connections (over TCP port 443) to the machine `server.rptn.ca`. To submit RPTN reports, the server must be able to make outgoing HTTPS connections to `server.rptn.ca` and also be permitted to send outgoing e-mail to `rptn-server@rptn.ca`. If you have a firewall in front of the CanIt-PRO server, please ensure that the firewall rules permit the RPTN traffic.

Chapter 8

Permissions

8.1 Introduction

In addition to the fairly coarse-grained settings described in Section 5.3.1, "User Privilege", on page 55, CanIt-PRO allows you to implement fine-grained control over access to various parts of the Web-based interface.

CanIt-PRO has two kinds of permissions:

1. *Stream Permissions* control access to CanIt-PRO features that affect the filtering of e-mail. For example, the ability to whitelist or blacklist senders, create custom rules, and so on are all Stream Permissions. Stream Permissions depend on *both* the user *and* the stream; a given user may have different permissions in different streams.
2. *User Permissions* control access to various parts of the CanIt-PRO user-interface not directly connected to filtering mail. For example, access the different GUI preferences and the ability to do WHOIS lookups are all User Permissions.

CanIt-PRO can associate permissions with *users* and with *groups*. Any user can be a member of zero or more groups. CanIt-PRO always grants a user the *union* of all his user-specific permissions and all his group permissions. Adding a user to a group, therefore, can only ever grant additional permissions. It cannot take away permissions.

8.2 Stream Permissions

Every stream has associated with it an ordered list of *stream classes*. When CanIt-PRO looks up stream permissions, it first calculates the list of stream classes associated with a particular user and stream. Here is how CanIt-PRO computes the list of stream classes:

1. The name of the stream always comes first. Thus, for example, if you are viewing a stream called `mystream`, then the list of stream classes starts with `mystream`.

2. If mystream happens to be your “home stream” (Section 3.5), then @@HOME is added to the list of stream classes.
3. If you have write-access in mystream, then @@WRITABLE is added to the list of stream classes.
4. If you have read-access in mystream, then @@READABLE is added to the list of stream classes.
5. Finally, the wildcard value * is added to the end of the list of stream classes.

When CanIt-PRO determines what permissions you have in a particular stream, it uses the following procedure:

1. It looks for permissions granted in the actual stream name. If it finds any, it stops searching the stream classes.
2. Otherwise, it checks the the stream classes and adds all permissions found to the set of granted permissions.

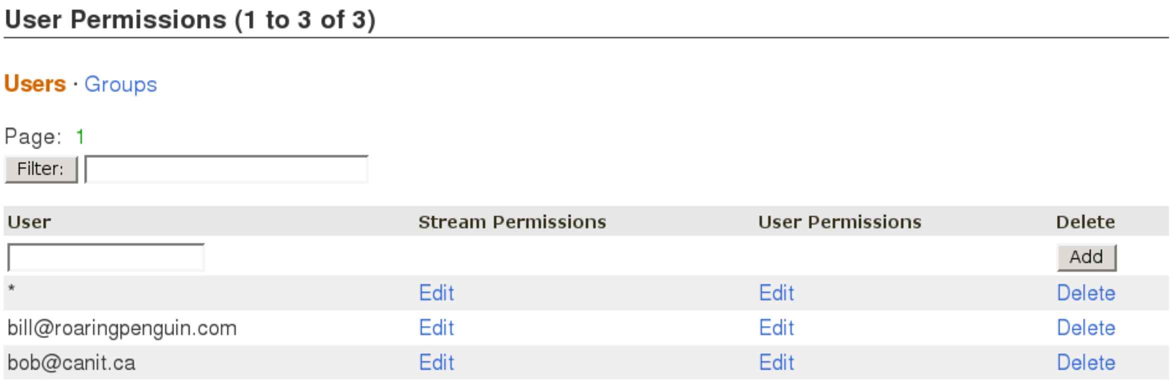
8.3 Determining Permissions

To determine a particular user’s permissions, CanIt-PRO performs the following steps:

1. First, it gathers all permissions associated with the particular user’s login ID.
2. Next, it adds all permissions granted to all the groups to which the user belongs.
3. If there was no entry in the permissions table for the particular user (that is, if Step 1 found no entries), then CanIt-PRO performs the following steps:
   (a) If the user has *root* privileges, then CanIt-PRO adds all permissions granted to the pseudo-user *root*.
   (b) If the user does not have root privileges, or the previous step yielded no entries, then CanIt-PRO adds all permissions granted to the wild-card user *.

8.4 Granting Permissions

To grant or deny permissions, click on “Administration” and then “Permissions”. The Permissions Page appears:



User Permissions (1 to 3 of 3)

Users · Groups

Page: 1

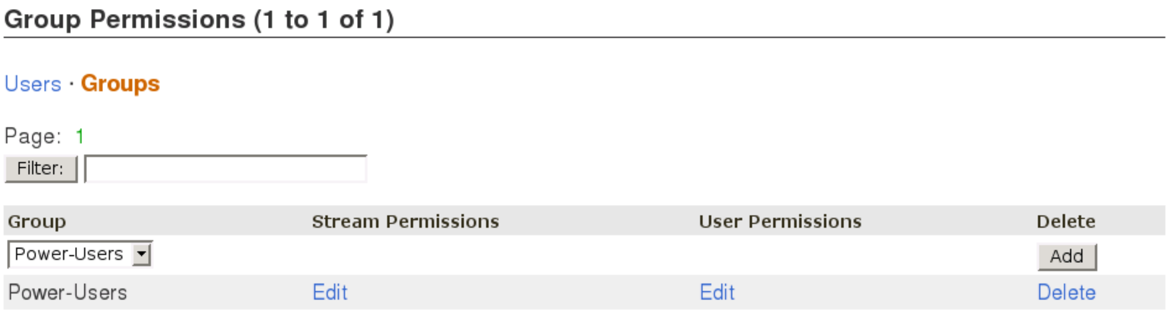
Filter:

User	Stream Permissions	User Permissions	Delete
			Add
*	Edit	Edit	Delete
bill@roaringpenguin.com	Edit	Edit	Delete
bob@canit.ca	Edit	Edit	Delete

Figure 8.1: Permissions Page

If you enter a string in the "Filter:" box, then CanIt-PRO limits the display to entries whose **User** column contains that string.

If you want to edit permissions for groups rather than users, click on the "Groups" link:



Group Permissions (1 to 1 of 1)

Users · Groups

Page: 1

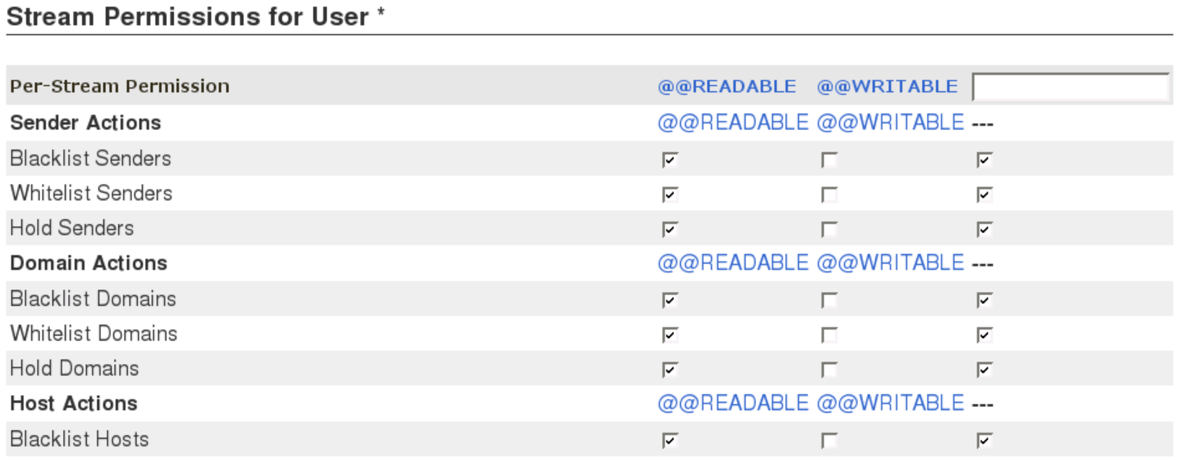
Filter:

Group	Stream Permissions	User Permissions	Delete
Power-Users			Add
Power-Users	Edit	Edit	Delete

Figure 8.2: Permissions Page

8.4.1 Granting Stream Permissions

To grant stream permissions, click on the "Edit" link in the **Stream Permissions** column. The Stream Permissions page appears:



Stream Permissions for User *

Per-Stream Permission	@@READABLE	@@WRITABLE	□
Sender Actions	@@READABLE	@@WRITABLE	---
Blacklist Senders	☑	□	☑
Whitelist Senders	☑	□	☑
Hold Senders	☑	□	☑
Domain Actions	@@READABLE	@@WRITABLE	---
Blacklist Domains	☑	□	☑
Whitelist Domains	☑	□	☑
Hold Domains	☑	□	☑
Host Actions	@@READABLE	@@WRITABLE	---
Blacklist Hosts	☑	□	☑

Figure 8.3: Stream Permissions Page

- To enable a stream permission in a particular stream or stream class, enable the checkbox in the appropriate row and column.
- To enter the name of a stream or stream class, enter it into the text box in the **Per-Stream Permission** row. Note that when you enter permissions for a new user, you *must* enter the stream class in the text box, or your changes will be discarded.
- To delete all permissions for a particular stream or stream class, click the "Delete" link at the bottom of the appropriate column.
- To view permissions only for one stream or stream class, click on the stream or stream class name.
- To make your changes take effect, click **Submit Changes**.

The Stream Permissions are:

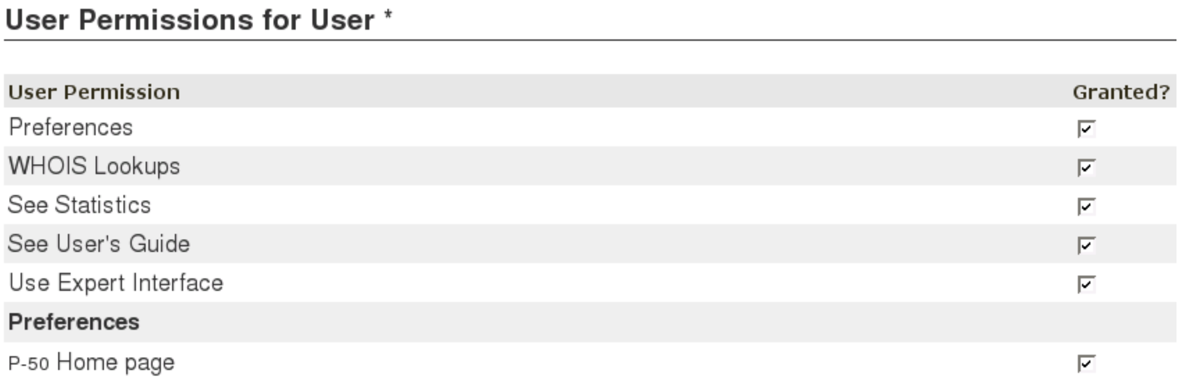
- **Blacklist Senders** – The user is permitted to blacklist senders.
- **Whitelist Senders** – The user is permitted to whitelist senders.
- **Hold Senders** – The user is permitted to add a hold rule for senders.
- **Blacklist/Whitelist/Hold Domains** – These permissions are similar to the Sender Action permissions, but they apply to domain rules.
- **Blacklist/Whitelist/Hold Hosts** – These permissions are similar to the Sender Action permissions, but they apply to host rules.
- **Reject/Accept/Hold MIME Types** – These permissions are similar to the Sender Action permissions, but they apply to MIME type rules.

- **Reject/Accept/Hold Filename Extensions** – These permissions are similar to the Sender Action permissions, but they apply to filename extension rules.
- **Custom Rules** – The user is permitted to create custom rules.
- **Mismatch Rules** – The user is permitted to create mismatch rules.
- **SPF Rules** – The user is permitted to create SPF rules.
- **RBL Rules** – The user is permitted to create RBL rules.
- **Bayes Settings** – The user is permitted to edit Bayes scoring rules.
- **Blacklisted Recipients** – The user can blacklist recipients.
- **Valid Recipients** – The user can enter recipients into the Valid Recipients Table.
- **See One-Shot/Pending/Non-Spam/Spam Message** – The user can see the specified message type in the trap. Note that these permissions are normally *off* for `@@READABLE` streams; otherwise, the user could see `default`'s spam trap.
- **Add Alternate Addresses to Streams** – The user can add aliases to his/her stream.
- **Opt In/Out** – The user can opt in or out of spam-scanning.
- **Adjust Notification Settings** – The user can adjust his or her notification settings.
- **See Per-Stream/Global Reports** – The user can see the specified reports.
- **Stream Settings** – Every stream setting has an associated permission. The user can only see a stream setting if its corresponding permission has been granted. The user can only change a stream setting if the permission has been granted and the user has write-access in the stream.

Note: If a user does not have write-access in a stream, then permissions such as **Custom Rules**, **Whitelist Senders**, etc. merely permit the user to *see* the rules. He or she still cannot change them.

8.4.2 Granting User Permissions

To grant user permissions, click on the "Edit" link in the **User Permissions** column. The User Permissions page appears:



User Permissions for User *

User Permission	Granted?
Preferences	☑
WHOIS Lookups	☑
See Statistics	☑
See User's Guide	☑
Use Expert Interface	☑
Preferences	
P-50 Home page	☑

Figure 8.4: User Permissions Page

The following User Permissions may be granted:

- **Preferences** – Unless this permission is granted, the user will not have access to the "Preferences" menu or any of its sub-menus.
- **WHOIS Lookups** – If this permission is granted, the user will be allowed to do WHOIS lookups.
- **See Statistics** – Allows the user to see the "Reports : Statistics" page.
- **See User's Guide** – Enables the link to the user's guide.
- **Use Expert Interface** – Grants the user access to the expert interface.
- **Preferences** – Each preference setting has an associated permission. A user can only change those settings for which permission has been granted.

Chapter 9

Streams, Inheritance and the Simple GUI

9.1 Simplification

CanIt-PRO is extremely versatile, allowing end-users to set many parameters such as blacklists, whitelists, custom rules, and so on. For many users, this is intimidating—the users may be unsophisticated, and just want to “make spam stop.”

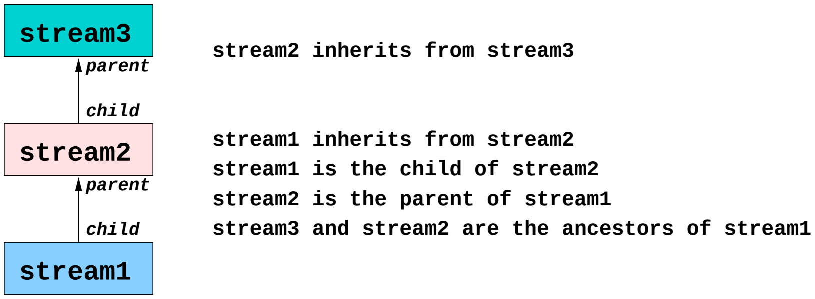
CanIt-PRO allows the administrator to set up special streams with pre-configured settings. Unsophisticated users then see a very simple interface which allows them to choose from one of these settings. CanIt-PRO achieves this with *stream inheritance* and *special streams*.

9.2 Stream Inheritance

Streams in CanIt-PRO *inherit* rules and settings from other streams. By default, all streams inherit rules and settings from the `default` stream.

If a stream `stream1` inherits from another stream `stream2`, we refer to `stream2` as the *parent* of `stream1`. Conversely, we call `stream1` the *child* of `stream2`.

Furthermore, suppose that `stream2` inherits from `stream3`. We then call `stream3` and `stream2` the *ancestors* of `stream1`. These terms are illustrated in Figure 9.1:



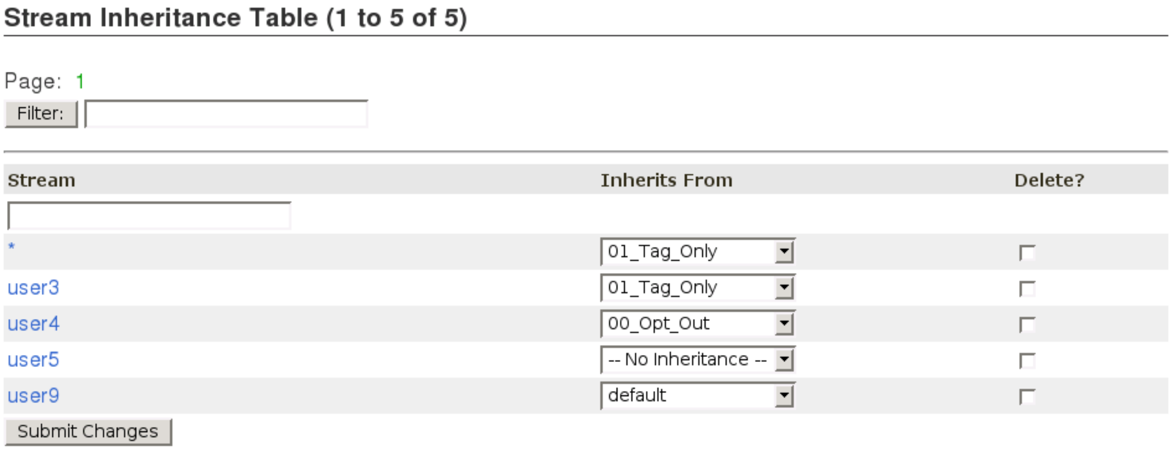
```
stream2 inherits from stream3

stream1 inherits from stream2
stream1 is the child of stream2
stream2 is the parent of stream1
stream3 and stream2 are the ancestors of stream1
```

Figure 9.1: Stream Inheritance Terminology

In addition to the default inheritance, streams can be configured to inherit rules and settings from *Special Streams* (discussed next in Section 9.3.)

To determine a stream's inheritance, CanIt-PRO consults the Stream Inheritance Table. To see this table, click on "Administration" and then "Inheritance":



Stream Inheritance Table (1 to 5 of 5)

Page: 1

Stream	Inherits From	Delete?
*	01_Tag_Only	□
user3	01_Tag_Only	□
user4	00_Opt_Out	□
user5	-- No Inheritance --	□
user9	default	□

Figure 9.2: Stream Inheritance Table

To determine a stream's parent, CanIt-PRO first looks up the stream in the inheritance table. If there is an entry, then that entry is used to determine the parent. If there was no entry, CanIt-PRO looks up the key "*" in the inheritance table. If such an entry exists, it is used to determine the parent.

In the example in Figure 9.2:

- `user3` inherits from `01_Tag_Only`.
- `user4` inherits from `00_Opt_Out`.
- `user5` does not inerit from any other stream.
- `user9` inherits from `default`.

- All other streams (except for `default`) inherit from `01_Tag_Only`, because of the wildcard entry.

If you enter a string in the “Filter:” box, then CanIt-PRO limits the display to entries whose **Stream** or **Inherits From** columns contain that string.

9.3 Special Streams

A *Special Stream* is a normal stream with two extra behaviors:

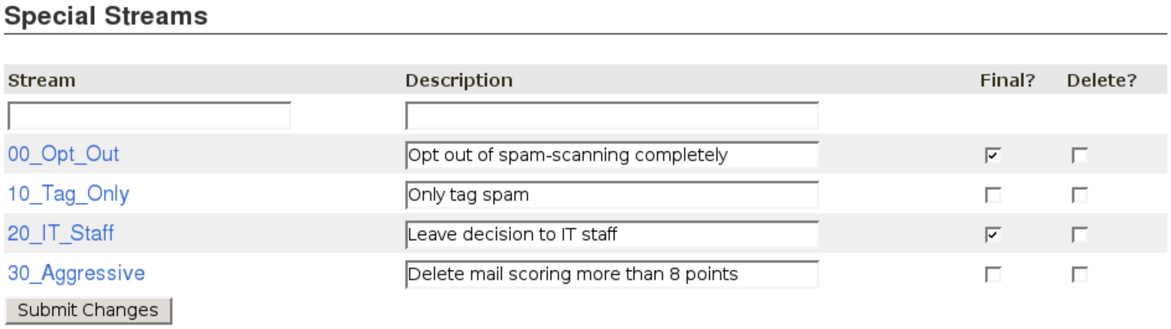
- Other streams are allowed to inherit from special streams. Normally, a stream can only have `default` as its parent. If you add special streams, however, other streams are allowed to make the special streams their parents.

- If a stream inherits from a special stream, then mail for the child stream is trapped in the *parent*’s trap. That is, by inheriting from a special stream, a stream “loses” its trap, giving responsibility for any trapped mail to the special stream.

9.3.1 Final Streams

A special stream may be marked *final*. If a special stream is marked final, then children of that stream may not override the special stream’s rules or settings. If a stream inherits from a final special stream, it’s as if the stream has given *all* control over to the special stream.

To see special streams, click on “Administration” and then “Special Streams”. The Special Stream Table appears:



Special Streams

Stream	Description	Final?	Delete?
00_Opt_Out	Opt out of spam-scanning completely	☑	□
10_Tag_Only	Only tag spam	□	□
20_IT_Staff	Leave decision to IT staff	☑	□
30_Aggressive	Delete mail scoring more than 8 points	□	□

Figure 9.3: Special Stream Table

9.3.2 Creating Special Streams

To create a special stream, enter the name of the stream in the **Stream** text box, and a user-friendly description in the **Description** box. Then click **Add Special Stream**.

In the example, the four streams 00_Opt_Out, 10_Tag_Only, 20_IT_Staff and 30_Aggressive have been created. (Special streams are presented to end-users in order of the stream name, so we named the streams beginning with numbers so they would sort from least to most aggressive. We leave gaps between the stream numbers so we can insert more streams in between if required.)

Once you have created the special streams, configure them appropriately. For example, for 00_Opt_Out, you'd switch into that stream, and then under "Preferences : Opt In/Out", you'd opt that stream out. (For convenience, you can click on a stream name in the Special Stream Table to switch into that stream.) For 30_Aggressive, you might change the stream settings to auto-discard anything scoring 8 or more on the spam scale. For 20_IT_Staff, you could have CanIt-PRO hold suspect spam, and have a member of your IT staff check 20_IT_Staff's trap and release false-positives.

Note that 00_Opt_Out and 20_IT_Staff are marked final. This means that rules and settings in streams inheriting from these two special streams are ignored; only the special streams' settings and rules are used. On the other hand, streams inheriting from 10_Tag_Only and 30_Aggressive may define their own rules, settings, whitelists and blacklists.

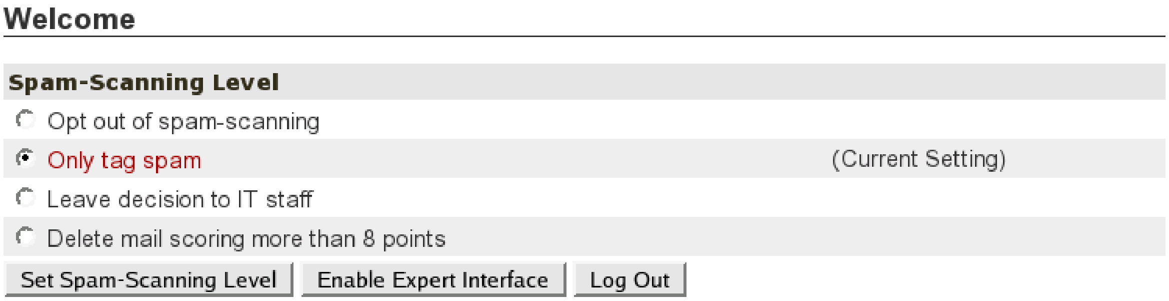
You can define as many special streams with as many different settings as you deem appropriate. Note that all special streams (by default) inherit from the default stream.

9.3.3 Deleting Special Streams

To delete a special stream, enable the checkbox in the **Delete?** column for the appropriate stream. Then click **Submit Changes**. *Warning:* If you delete a special stream, then all inheritances from that stream are deleted. Please see Section 9.2 for more details.

9.4 The Simplified GUI

If the CanIt-PRO administrator enabled **Users authenticated by alternate means default to simple GUI?** (Section 5.1), then such users only see the Simplified Interface:



Welcome

Spam-Scanning Level

○ Opt out of spam-scanning

● Only tag spam (Current Setting)

○ Leave decision to IT staff

○ Delete mail scoring more than 8 points

Figure 9.4: Simplified Interface

The simplified interface simply lists the possible Special Streams. The currently-inherited special stream is highlighted in bold red print.

To inherit from a different stream, the user simply clicks on the appropriate radio button and clicks **Set Spam-Scanning Level**. This adjusts the entry in the inheritance table.

To log out, the user clicks on **Log Out**.

If the user clicks on **Enable Expert Interface**, then he or she will have access to the usual CanIt-PRO interface. He or she can then turn off inheritance (via "Preferences : Set Default Stream") and take control over his or her own blacklists, whitelists, rules and spam trap.

Note: If you have set the global setting **Switching to expert mode cancels stream inheritance** to **Yes**, then the act of clicking **Enable Expert Interface** cancels any inheritance that was in force, making the stream inherit from `default` again.

To get back to the simple GUI, click on "Simple Interface" top-level menu entry. Note that this menu entry does not appear until at least one special stream has been defined.

9.5 Inheritance from Non-Final Streams

If a stream inherits from a non-final stream, CanIt-PRO uses the following procedures to resolve rules. In these examples, we assume that stream `john` inherits from the non-final stream `10_Tag_Only`

- For sender, domain and host blacklists and whitelists, and for MIME type and mismatch rules, CanIt-PRO first looks for a rule associated with the original stream (in our example, `john`.) If no such rule is found, it then tries the parent stream (in our example, `10_Tag_Only`) and then the parent of the parent, and so on up the inheritance chain.

- For custom rules, CanIt-PRO uses all the rules associated with the original stream *in addition* to rules associated with the ancestor streams.

- Bayes data is associated with the original stream (`john`) and not the parent stream (`10_Tag_Only`).

9.6 Inheritance from Opted-Out Streams

If a stream or *any* of its ancestors is opted-out of spam-scanning, then no spam scanning is performed.

Chapter 10

Locked Addresses

10.1 Introduction to Locked Addresses

Locked Addresses are designed to solve the following problem: You want to give out your e-mail address to someone, but you don't trust that person or organization not to turn around and give or sell it to others. You want an address that can only be used by the person or organization you give it to, and not by anyone else.

CanIt-PRO has a complete solution to this problem. However, it does require some administrative overhead before users can take advantage of the feature.

10.2 Preparing to use Locked Addresses

Before end-users can use locked addresses, you need to perform the following steps.

10.2.1 Create a new domain

Choose a new domain, specifically for locked addresses. This domain should be a subdomain of your "real" domain. For example, if you own the domain roaringpenguin.com, you might choose to place all your locked addresses in la.roaringpenguin.com. The domain you use for locked addresses should contain *only* locked addresses and should not be used for any "real" e-mail addresses.

10.2.2 Configure mail for the new domain

The next step is to configure the CanIt-PRO machine to receive mail for the new domain. Obviously, the first thing you need to do is publish an MX record for the domain. For example, if your locked address domain is la.roaringpenguin.com and your CanIt-PRO server's name is canit.roaringpenguin.com, you might add a DNS record that looks like this:

```
la.roaringpenguin.com.   1d   IN   MX   1 canit.roaringpenguin.com.
```

Also, you need to configure the CanIt-PRO machine to accept *and discard* all mail for the locked domain. (Mail should never be delivered to addresses in the locked domain, but just in case, there should be a mechanism to discard them.)

Configuring Sendmail to accept mail for the locked domain is easy: Just add an entry in the access database. In our example, it would be:

```
To:la.roaringpenguin.com    RELAY
```

The easiest way to configure Sendmail to discard mail for the locked domain is to make use of the `virtusertable` feature. Add an entry like this in `virtusertable`:

```
@la.roaringpenguin.com           devnull@canit.roaringpenguin.com
```

and ensure that mail to `devnull@canit.roaringenguin.com` gets discarded (by making an alias from `devnull` to `/dev/null`.)

(Of course, you'd substitute your own locked address domain for `la.roaringpenguin.com` and your own CanIt-PRO server name for `canit.roaringpenguin.com`.)

10.2.3 Inform CanIt-PRO about the locked address domain

CanIt-PRO needs to know the domain you're using for locked addresses, so it can treat any such addresses specially. In the Web interface, click on "Administration" : "Global Settings" and enter the locked address domain into the global setting "Domain for Locked Addresses"

10.2.4 Associate each login name with an e-mail address

CanIt-PRO can only generate locked addresses if it has a *real* e-mail address for each logged-in user. For users in CanIt-PRO's built-in user table (Section 5.3 on page 55), simply ensure that you enter an e-mail address for each user.

For users authenticated via external means, the User Lookup method must return the user's e-mail address upon login. For some User Lookup methods such as POP3 or IMAP that cannot return the e-mail address, you need to create an `account-info` script (Section 6.2.4 on page 74) and ensure that a `mail=`*`email-address`* attribute is always emitted for each login that should be permitted to use locked addresses.

Once all of these steps in Sections 10.2.1 through 10.2.4 have been performed, the Locked Address feature is ready to use. Please consult the CanIt-PRO User's Guide for details about how to use a Locked Address.

Chapter 11

Attachment Handling

CanIt-PRO can handle file attachments in a number of different ways. Messages can be delayed, rejected or held based on the attachment's type. They can be scanned for viruses and held or rejected using one or more configured virus scanners. If desired, attachments can also be removed from the message and discarded, or held for access via a web-based system.

11.1 General Filename and MIME Type Rules

Whole messages can be rejected or held on a per-stream basis using the **Filename Extensions** or **MIME Types** rules. See the section entitled **Blacklists, Whitelists and Rules** in the CanIt-PRO Users Guide for full details.

11.2 Delaying Attachments

On a site-wide basis, it is sometimes useful to delay certain attachment types temporarily, without placing them in a stream's trap area. By delaying these attachments for a short period of time, you can give your virus scanners and RBLs time to catch up with new virus and spam content.

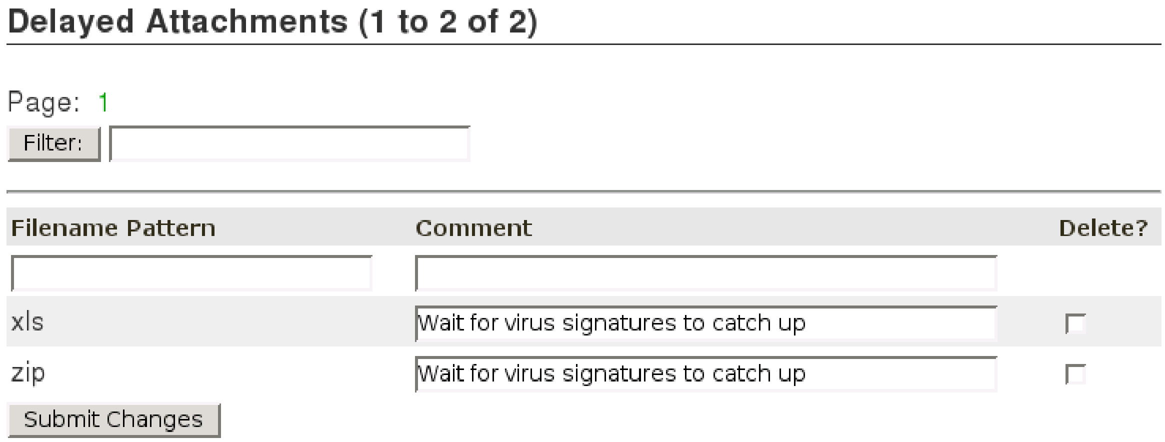
11.2.1 Enabling the Feature

First, the feature must be enabled via the Web GUI. Log in as an admin user, and enable **Delay Attachments** on the **Setup : Features** page.

Next, configure the time delay, by modifying **Time in hours to delay messages with Delayed Attachments** under **Global Settings**.

11.2.2 Creating Delay Rules

To create a delay rule, click on **Administration** and then **Delayed Attachments**. The Delayed Attachments screen appears:



Delayed Attachments (1 to 2 of 2)

Page: 1

Filename Pattern	Comment	Delete?
xls	Wait for virus signatures to catch up	□
zip	Wait for virus signatures to catch up	□

Figure 11.1: Delayed Attachments

To add a rule:

1. Enter a filename pattern in the **Filename Pattern** box. A filename pattern is normally interpreted as a filename extension. For example, `exe` will match a file with the extension `.exe`. Note that the pattern should *not* contain a period. If a filename pattern begins with `ˆ`, then it matches an entire filename. For example, the pattern `ˆbad.exe` matches (only) the filename `bad.exe`.
2. Enter a comment in the **Comment** box. This will help you remember why you are delaying the given filename pattern
3. Click **Submit Changes** to add the rule.

Note: Attachment-delaying is global. It *cannot* be adjusted on a per-stream basis.

11.2.3 How It Works

As an administrator, you may configure any number of file extensions or full filenames to be delayed. When a message arrives matching that filename or extension, it will be held in a special `@@DELAYED` stream for the number of hours specified in the **Time in hours to delay messages with Delayed Attachments** configuration.

Once that time is elapsed, the message is automatically released from the `@@DELAYED` trap, proceeding through the CanIt-PRO filtering process where normal scanning will proceed as if that mail had just arrived.

Should it be necessary for a message to be released from `@@DELAYED` early, the `admin` user (or other user with appropriate permissions) may manually release it. Note, however, that a message released from `@@DELAYED` may be re-trapped in its normal stream because of spam-scoring rules. That is because messages released from `@@DELAYED` are scanned by CanIt-PRO as if they had never been seen before; CanIt-PRO does not correlate what it believes to be a brand new message with anything in the `@@DELAYED` stream.

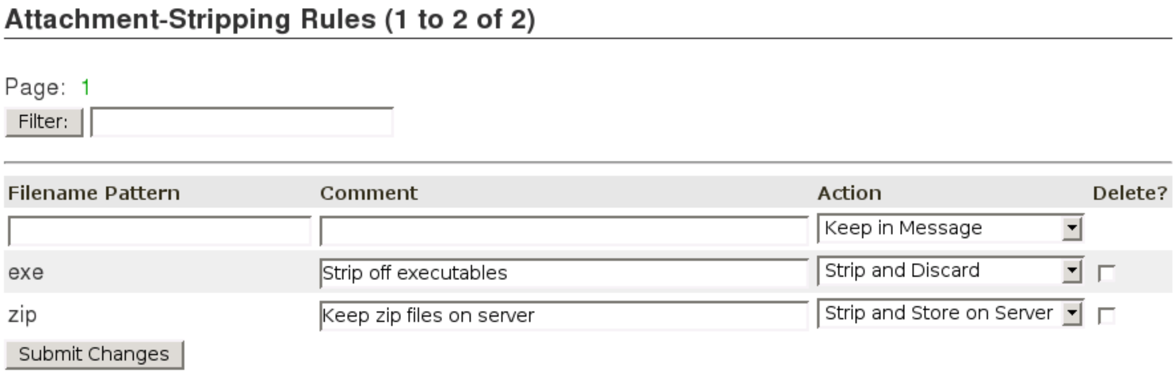
11.3 Stripping Attachments

In addition to delaying, holding or rejecting mail based on characteristics of attachments, CanIt-PRO can strip attachments out of messages before forwarding the message. You can configure CanIt-PRO to strip out attachments and store them for retrieval via the Web interface, or simply to strip them out and discard them.

Attachment-stripping rules *can* be set per-stream, but only the CanIt-PRO administrator can create or edit attachment-stripping rules; normal users cannot. In addition, all streams inherit `default`'s attachment-stripping rules, even if the "Inherit rules from 'default' stream" setting is set to **No**.

To create attachment-stripping rules:

1. Click on **Rules** and then **Attachment Stripping**. You see the Attachment Stripping Screen:



Attachment-Stripping Rules (1 to 2 of 2)

Page: 1

Filename Pattern	Comment	Action	Delete?
		Keep in Message	
exe	Strip off executables	Strip and Discard	☐
zip	Keep zip files on server	Strip and Store on Server	☐

Figure 11.2: Attachment-Stripping Rules

2. Enter a filename pattern in the **Filename Pattern** box. This pattern is interpreted exactly as for Delayed Attachments.

3. Enter a comment in the **Comment** box.

4. Choose an **Action** setting to determine how CanIt-PRO handles the filename pattern:

   - **Keep in Message** indicates that CanIt-PRO should not strip the attachment out. This setting can be used in a particular stream to override settings in `default`.
   - **Strip and Store on Server** indicates that CanIt-PRO should remove the attachment and store it in the PostgreSQL database. CanIt-PRO will also add a message indicating that the attachment was stripped, and provide a link whereby the message recipient can retrieve the attachment.
   - **Strip and Discard** indicates that CanIt-PRO should remove and discard the attachment. CanIt-PRO will add a note to the message indicating that the attachment was discarded and cannot be retrieved.

5. Click **Submit Changes** to create the rule.

Chapter 12

Tips

Managing spam requires constant attention, but there are many things you can do to reduce the workload of the spam-control officer. This chapter offers advice for fine-tuning CanIt-PRO and making it more effective.

12.1 Hit-and-Run Spam

In the past, spammers would use open SMTP relays to send spam. With the advent of inexpensive residential broadband, many spammers use special software to send bulk mail directly from their PC's.

Because spammers want wide distribution, they want each message to be sent as cheaply as possible. Some spam software, therefore, ignores SMTP errors if a message cannot be delivered. This is the motivation behind the "One-Shot" message category.

CanIt-PRO can deal very effectively with hit-and-run spam software by sending a temporary failure indication at the MAIL FROM: or RCTP TO: SMTP command when mail from an unknown sender arrives.

If you set the **Tempfail unknown senders on first transmission** setting to **Global**, then at the MAIL FROM: command, CanIt-PRO sends a temporary-failure indication if mail is from an unknown sender. If the message is retried, CanIt-PRO permits the SMTP dialog to continue.

If you set the setting to **Global-Plus-IP**, then CanIt-PRO forces a given sender to retry from the same class-C network as the initial attempt. This is to thwart spammers who amass a worldwide list of open relays or proxies.

If you set the setting to **Per-Recipient**, then at the RCPT TO: command, CanIt-PRO sends a temporary failure notification if the sender has never attempted to send mail to the recipient before. This is a very powerful weapon against hit-and-run spamware.

Finally, the setting **Per-Recipient-Plus-IP** is the most stringent: It forces each sender address to retry for each recipient address from the same class-C network.

The table of "known senders" is purged based on the **Expire old data after this many days** setting. If you set this to 45 days, for example, then any known sender who hasn't sent mail for the last 45 days is purged from the table. However, entries in the table which have attempted only one transmission are purged after the shorter **Mark one-shot messages as spam after this many days** setting, which is typically 7 days.

There are some down-sides to using the anti-hit-and-run features. Valid mail from new senders may be delayed by anywhere from 15 minutes to four hours, depending on the retry interval on the sending relay. You can avoid this delay by setting up a secondary MX record. In fact, you can simply give the CanIt-PRO machine a virtual interface with another IP address and publish this other IP address as a secondary MX record. In this way, when proper SMTP relays receive a temporary failure indication on the primary MX machine, they immediately try to send to the secondary MX machine. Hit-and-run spamware won't retry.

On a similar note, CanIt-PRO will not issue temporary failures for messages relayed from any server listed as a Secondary MX Relay (see section 2.7.2 on page 26) or for a server in a Known Network with **Skip Hit-and-Run** configured (see section 4.5 on page 39). If a message is received by such a server, hit-and-run detection will not be used. In some cases, this can cause hit-and-run statistics to be skewed. For example, if mail is initially received by a CanIt-PRO server and marked as hit-and-run, then is received by a secondary MX server and either relayed to the CanIt-PRO server, or to an internal mail server, the message will appear in the CanIt-PRO statistics as ”hit-and-run”, even if it was received and processed.

Also, if you use the **Per-Recipient** setting, the size of the known-senders table grows to the product of number of senders by number of recipients. In an organization with thousands of recipients, this table could easily grow to millions of entries and consume several tens of megabytes of disk space.

In general, however, we find that setting **Tempfail unknown senders on first transmission** to **Per-Recipient** is a cheap and effective way to combat hit-and-run spam.

Hit-and-run statistics are added to the “One-Shot” column in the statistics display after two days. (This is because we need to wait a while to ensure the connection attempts really are hit-and-run.) Therefore, the “One-Shot” column is only accurate for rows older than two days.

WARNING: Some mailing list programs use “disposable” sender addresses which always change. These lists do *not* work well with the anti-hit-and-run scheme. To work around the problem, you should whitelist the domain of the mailing list sender.

CanIt-PRO tries to detect disposable-address schemes. It ignores everything in the sender address following a plus sign or a dash followed by a digit. These rules catch most common methods for generating disposable addresses, but they are not exhaustive.

12.2 Don’t Trust Sender Addresses

Many spammers use one-time disposable sender addresses. Many addresses are not even valid. So we do not recommend blacklisting addresses unless you receive many different spam addresses from the same address. Therefore:

> Blacklisting individual addresses is usually not effective. Whitelisting known good addresses (for example, mailing-list sending addresses) can be very effective. The sender report may, however, highlight a persistent spam sender address which is worth blacklisting.

12.3 Don't Trust Sender Domains

Just as sender addresses are often fake, sender domains are too. However, some domains are known spammers and these can be profitably blacklisted. The tip:

> Blacklisting entire domains can be effective under limited circumstances. Whitelisting known good addresses (for example, mailing-list sending addresses) can be very effective. Holding all mail from free e-mail services like Hotmail and Yahoo can be effective if you use it in conjunction with whitelisting of known good senders from those services. Use the domain report to help make these decisions.

12.4 You May Trust Relay Hosts

It is rather difficult to fake the IP address of the SMTP relay host, so this attribute can usually be trusted. We recommend using a DNS-based blacklist service in your Sendmail configuration file to reject the most obvious offenders. However, if you receive multiple spam messages from a given relay host, it can be effective to block the host:

> Blacklisting a repeat-offender relay host is effective. Whitelisting known good hosts such as internal hosts is effective and recommended. Use the host report to determine which hosts are persistent spam relays.

12.5 Custom Rules

12.5.1 General Recommendations

There are a few custom rules which are quite effective:

1. If you know that your CanIt-PRO server only accepts inbound mail from the Internet, then no server should ever claim to be in your domain in the HELO command. If your CanIt-PRO server is called `canit.mydomain.tld`, a custom rule to add 5 points if HELO ends with `mydomain.tld` can be very effective. In fact, you might want to make high-scoring rules which automatically reject messages with obviously-fake HELO arguments.

2. Similarly, no machine should ever put an IP address as the argument of HELO. Some spammers use random IP addresses here to confuse spam-reporting tools. A custom rule which "regexp-matches" HELO against `^\d+\.\d+\.\d+\.\d+$` can be quite effective.

3. Custom rules which specify Sender contains "offer", "bounce", "return" and "noresponse" can often trap spam. You should use only moderate scores on these rules, because some legitimate mail comes from such senders. However, adding a rule which scores 5 for these patterns can help catch a lot of spam which might otherwise sneak under the scoring threshold.

4. Subject-matching rules for the most obnoxious spams are very effective. For example, Subject regexp-match rules against `v\Sagra` and `(increase|enlarge).*penis` are very effective.

12.5.2 Things to avoid

Be very careful when writing custom rules, especially rules that can match on the message body. For example, a straightforward rule that contains “cum” in the body will match mail containing mail containing “document”, “cumulative”, “modicum” and at least 64 other common English words. Similarly, “sex” will match “sexton”, “Essex” and others.

If you want to match words in a message body, we recommend that you use a regular-expression match, and use Perl’s word-boundary operators. For example, the Perl regular expression `\bcum\b` matches the word “cum”, but not “document”, “cumulative” or “modicum”.

12.6 Group High-Scoring Messages Together

We recommend that you set the default sort order to sort by Score, Descending. This groups high-scoring messages at the beginning and low-scoring messages at the end of the pending list. This makes it easier for the spam-control officer to dispose of the messages.

> Reduce the spam-control officer’s work by sorting message summaries by Score, Descending. This lets the officer use the interface more effectively.

12.7 Roaring Penguin Best-Practices

At Roaring Penguin Software Inc., we’ve spent quite a bit of time analyzing spam and spammers. You may wish to try out some of our anti-spam rules to see if they work well for you. Here is a quick summary of the rules we use; they may inspire you to develop your own anti-spam rules.

- We use custom rules to add 5 to any message whose **Sender** contains “offer”, “noresponse”, “remove”, “marketing” or “promo”. These rules may be a touch aggressive for very busy sites, but are quite effective for smaller sites.
- Another custom rule adds 1.2 to any **Relay** containing “[” (left square bracket.) This indicates a reverse-DNS failure on the sending host, which is mildly correlated with spamming.
- We add 2,000 to messages whose **Subject** contains various offensive terms or obvious spam expressions.
- We use mismatch rules for the domains **aol.com**, **hotmail.com**, **ibm.com** and **yahoo.com**. Our rules reject the messages outright; this may be too aggressive for many sites.
- We use a **Spam threshold** of 4.6, because we find the default of 5 is somewhat conservative.
- We use a discard threshold of 20; this seems quite safe.
- We set **Tempfail unknown senders on first transmission** to **Per-Recipient**. Again, this may be too aggressive for large sites.

12.8 General Anti-Spam Tips

12.8.1 Use Receive-Only Addresses on your Web Site

Spammers love to extract e-mail addresses from Web sites, and not only do they use them for the obvious purpose of spam targeting, but also they use them as fake sender addresses.

Therefore, we recommend a general policy of publishing only generic e-mail addresses on your Web site, like `info@roaringpenguin.com` and `sales@roaringpenguin.com`. When you reply to inquiries, always use a real, personal e-mail address like `dfs@roaringpenguin.com`. This has two benefits:

1. If someone sends e-mail purporting to come from `info@roaringpenguin.com`, you know immediately that it is spam, and you can reject it. You can blacklist all your generic addresses inside CanIt-PRO.
2. If someone complains about receiving e-mail from one of the generic addresses, you can point to your policy and assure the recipient that the sender address was faked.

12.8.2 Do Not Reply to Spam

Do not ever reply to spam e-mail; such replies simply serve to validate your e-mail address. Similarly, do not visit Web sites purporting to offer opt-out services; they also serve to validate your address for further spamming.

Chapter 13

Security

Running a secure CanIt-PRO installation is relatively straightforward, but there are many issues you have to watch out for. This chapter gives you guidance on how to secure your CanIt-PRO installation.

13.1 Don't Run as Root

The most basic security principle is to run as little software as *root* as possible. Therefore:

- *Always* create the Sendmail *smmsp* user and group, and do not run Sendmail suid-*root*. Instead, the permissions on the Sendmail executable should look like this:
  `-r-xr-sr-x root smmsp sendmail`
  That is, the `sendmail` binary should be owned by *root*, group *smmsp* and have mode 2555.

- *Always* create the MIMEDefang *defang* user and group, and run MIMEDefang as *defang*. In `/etc/mail/canit/mimedefang.conf`, enable `MX_USER=defang`.

13.2 Ownership and Permissions

All system configuration directories like `/etc` and their descendants should be owned by *root* and writeable only by *root*. Here are suggested ownership and permissions for various files and directories. Note that where we use group *root*, your system may use *wheel* or some other group for *root*-owned files.

File or Directory	**Owner**	**Group**	**Mode**
`/etc/mail/canit` and ancestors	*root*	*root*	0755
`/etc/mail/canit/db-settings`	*apache*	*defang*	0640
`/var/spool` and ancestors	*root*	*root*	0755
`/var/spool/MIMEDefang`	*defang*	*defang*	0700
The PHP files in Apache's Web space	*root*	Apache's group	0644

13.3 PostgreSQL Security

By default, PostgreSQL trusts any connection coming from the local host. Therefore, if you use PostgreSQL on your CanIt-PRO server with the default access rules, *do not allow normal users to have shell accounts on the CanIt-PRO server.* This cannot be emphasized strongly enough: If you allow normal users shell access on the CanIt-PRO server with PostgreSQL's default setup, anyone can access or change the spam database.

If you must allow shell accounts on the CanIt-PRO server, then you *must* password-protect your PostgreSQL installation. See the PostgreSQL documentation ("Authentication Methods" section) for details. You must also protect your database passwords:

- The file `/etc/mail/canit/db-settings` must be owned by *apache* and group *defang*. Both the *defang* user and the *apache* user need read-access to these files, which should have mode 0640. (We assume your Web server runs as user *apache*; if not, substitute the Web server user as appropriate.)

For best security, we strongly recommend that you do not allow ordinary users to have shell accounts on your mail server. If the CanIt-PRO database server is on a different machine, you should not permit shell accounts on that machine either.

13.4 PHP Security

PHP has a parameter called `register_globals`, which automatically sets global variables based on GET, PUT or COOKIE variables. This setting may be a security risk, and CanIt-PRO does *not* require it. We strongly recommend that you set `register_globals` to `off`.

13.5 Network Security

When you log on to CanIt-PRO, your username and password are transmitted in cleartext. While you interact with CanIt-PRO, your browser passes a session cookie back so CanIt-PRO can keep track of your session. Both your password and the cookie are vulnerable to network sniffing. If you interact with CanIt-PRO over an untrusted network, or a network whose traffic may be sniffed, you should use https and SSL encryption. Setting this up is beyond the scope of this manual, but CanIt-PRO should operate with no changes over https.

13.6 Backups

The daily CanIt-PRO cron job dumps a text backup of the spam database to the file `/var/spool/Canit-Spam-DB-Backup/SPAM-DATABASE-BACKUP`. You should back this file up regularly in case the CanIt-PRO server suffers a hardware or other problem. You should also make sure the file is not readable by normal users.

If you are using the Berkeley DB back-end for Bayes data, you should also back up the entire directory tree rooted at `/var/spool/MD-Bayes`.

Some CanIt-PRO settings are stored in `/etc/mail/canit`; you should back up that directory any time that you change a file in it.

Appendix A

A Testing Topology for CanIt-PRO

A.1 Introduction

The best way to evaluate CanIt-PRO is to route real-world mail through it. However, you may be hesitant to place CanIt-PRO in production without testing it first. So we'll show you how to set up CanIt-PRO for test purposes, and then how to put it into production in a safe way. The test topology makes it very easy to back out of CanIt-PRO if you decide to do so.

A.2 Assumptions

We make the following assumptions about your current e-mail setup:

- You already have a mail server which is your primary MX record, and you control that server and its network. The existing mail server may run Sendmail, but it doesn't have to—it could run Netscape Messenger, Microsoft Exchange, or any other mail server software of your choice.
- You have a spare Intel-architecture server for installing Linux and CanIt-PRO. This server should have sufficient horsepower to handle all of the mail for your domain or domains. While you can use other supported UNIX operating systems for CanIt-PRO, the instructions in this paper are specific to Linux. If you are an experienced UNIX and Sendmail system administrator, you can probably translate them for your own system.
- You control your DNS settings and can publish MX records for your domains.

A.3 Network Setup

Figure A.1 illustrates the assumed existing network setup followed by the new network setup. Note that your actual setup may be more complex and may include firewalls, demilitarized zones, etc. Conceptually, however, we assume you have an existing mail server which is the primary MX machine for your domains, and which is connected to the Internet.

The test network shows how the CanIt-PRO server is configured to accept mail from the Internet and relay it to your actual mail server.

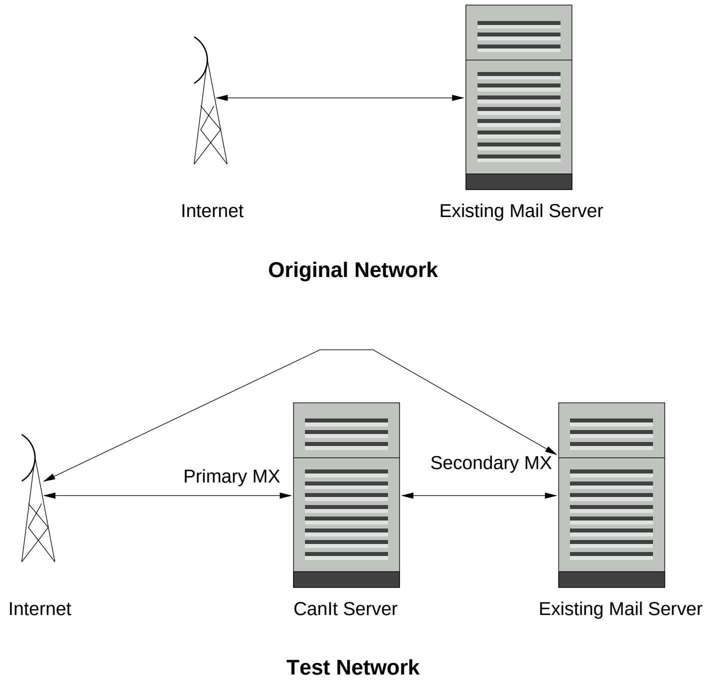


Figure A.1: Network Configurations

A.4 Build the CanIt-PRO Server

To build the CanIt-PRO server, install Linux on an Intel Architecture server. Be sure to install Apache, PHP and PostgreSQL, which are included with most Linux distributions.

Install CanIt-PRO according to the instructions in the manual. Be sure to run the Red Hat “prepare-system” script, because this script creates a skeleton Sendmail configuration required for mail relaying.

A.5 Configure the CanIt-PRO Server to Relay Mail

You’ll need to edit two files on the CanIt-PRO server to configure Sendmail to relay mail. Make a list of all the domains for which your existing mail server accepts mail. Let’s suppose you own the domains *example1.com* and *example2.net*, and accept mail for both on the machine *mail.example1.com*. Finally, we’ll assume the CanIt-PRO server is called *canit.example1.com*.

A.5.1 Enable Relaying

First, you must enable relaying for the domains you control. To do this, edit the file `/etc/mail/access` and add a line for each domain, something like this:

```
To:domainname.tld  RELAY
```

In our example, we'd add two lines to `/etc/mail/access`:

```
To:example1.com  RELAY
To:example2.net  RELAY
```

A.5.2 Configure Forwarding Relays

Next, you have to tell CanIt-PRO where to relay mail for the domains. Edit the file `/etc/mail/mailertable` and add a line for each domain, something like this:

```
domainname.tld  esmtp:[relay.domainname.tld]
```

In our example, recall that *mail.example1.com* handles mail for both domains, so our mailertable would look like this:

```
example1.com  esmtp:[mail.example1.com]
example2.net  esmtp:[mail.example1.com]
```

A.5.3 Rebuild Sendmail Databases

Finally, you need to rebuild Sendmail's internal databases to reflect these changes. Simply execute the following Linux commands as *root*:

```
cd /etc/mail
make
```

A.6 Route Test Mail

Up until this point, your existing mail server has continued to act as it always does. The CanIt-PRO machine, although "live" and on the network, is not handling any mail traffic. Now comes the time to route mail through the CanIt-PRO server. There are two options to route test mail through the CanIt-PRO server:

A.6.1 Direct Injection

The least disruptive method is to directly inject test messages into the CanIt-PRO server. Run an SMTP client and send messages via the CanIt-PRO server. Verify that they are received and that spam messages are held.

You can use an e-mail client such as Mozilla or Microsoft Outlook for testing purposes. Simply set the outgoing SMTP machine to be the CanIt-PRO relay (in our example, *canit.example1.com* and send messages to people in your organization.

Alternatively, you can use a UNIX or Linux machine with its own DNS server. Create an MX record for your domain pointing to the CanIt-PRO server and send messages. Remember, only the test machine thinks that CanIt-PRO is your mail relay; the rest of the Internet still uses your existing mail server.

A.6.2 Create a Test Subdomain

Another option is to create a test subdomain, such as *test.example1.com*. Configure your regular mail server to accept mail for that domain, and don't forget to modify the CanIt-PRO server's access and mailertable files to relay mail for that domain. Then publish an MX record for *test.example1.com* pointing to *canit.example1.com*. You can then send mail from anywhere in the Internet to someone at *test.example1.com* and it will be relayed through the CanIt-PRO server. Existing mail to your proper domain, however, will still travel via your old mail server.

A.7 Route Real Mail

Once CanIt-PRO has passed the initial tests, it's time to route real e-mail through it. The safest way to do this is to add an additional MX record for your domains. This record should have the highest priority, and point to the CanIt-PRO server.

For example, let's suppose your existing MX records look like this:

```
example1.com.  1d IN MX 10 mail.example1.com.
example1.com.  1d IN MX 15 m2.example1.com.
```

Simply add another MX record like this:

```
example1.com.  1d IN MX 5 canit.example1.com.
```

and propagate the DNS changes. Mail for your domain will now be routed through the CanIt-PRO machine. In an emergency, if you need to take the CanIt-PRO machine offline, simply kill Sendmail on the CanIt-PRO server. Relays attempting to deliver mail to your domain will first try the CanIt-PRO server and immediately get a "Connection refused" error. They will fall back very quickly to the remaining MX records, and mail will flow as usual.

Note: This test setup is *not* a viable topology for stopping spam. Because CanIt-PRO sends temporary-failure codes for suspect mail, if your real mail server has an MX record, the sender will simply relay the spam directly to it. For production use, all of your public records should either:

- Be running CanIt-PRO, or
- Relay to a machine running CanIt-PRO.

The actual internal mail server should be hidden (no MX record) and ideally firewalled off, so only the CanIt-PRO relay can connect to it.

A.8 Outgoing Mail

If you want to pass outgoing mail through CanIt-PRO, configure your mail server to use the CanIt-PRO server as a “SmartHost”. This is a host to which all non-local mail will be sent. The details of SmartHost configuration differ among mail servers; consult your mail server documentation for details.

Appendix B

CanIt-PRO Architecture

B.1 Introduction

CanIt-PRO is based on the Sendmail Milter API, described at `http://www.sendmail.com/partner/resources/development/milter_api/`. Milter is a scalable API for doing site-wide filtering of e-mail.

Figure B.1 shows how CanIt-PRO interfaces with Milter.

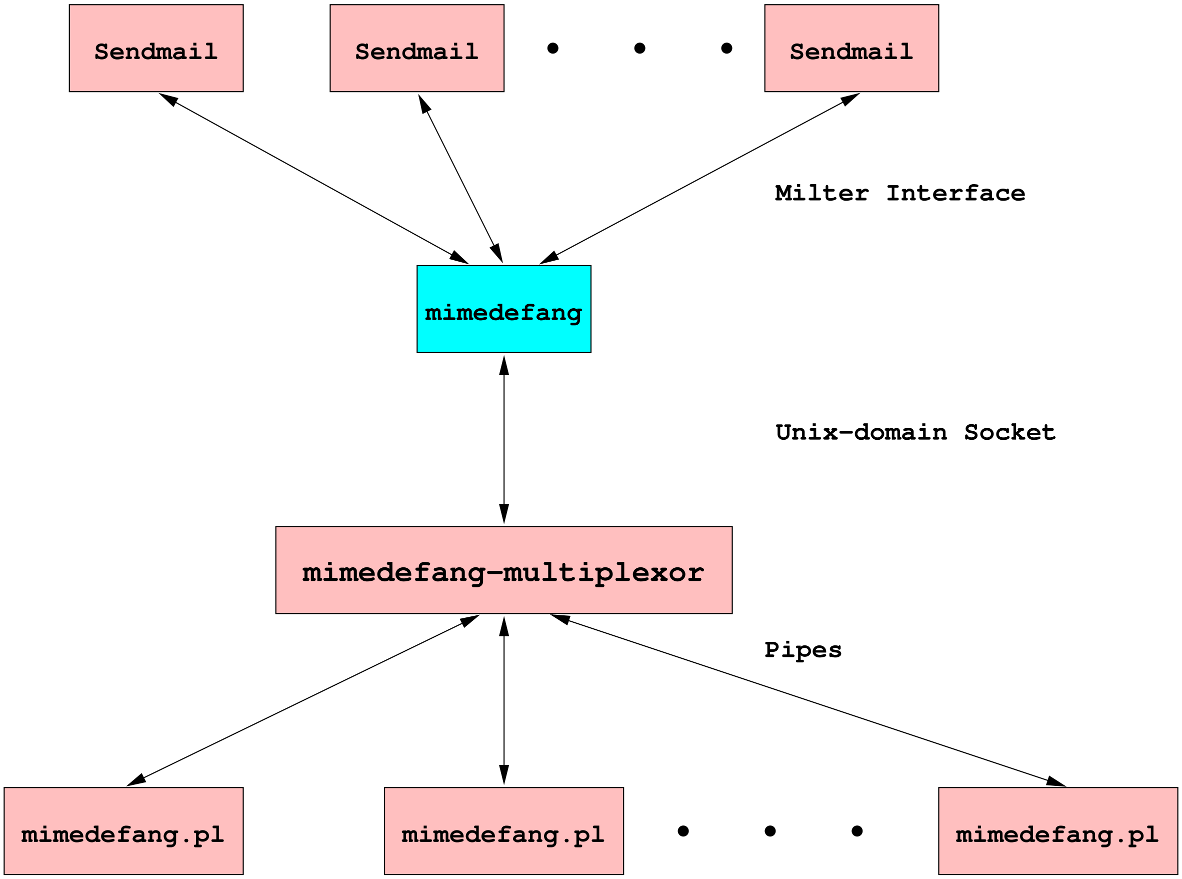


Figure B.1: CanIt-PRO Architecture

B.2 CanIt-PRO Architecture

In Figure B.1, we show multiple `sendmail` processes communicating with a single `mimedefang` process. The `mimedefang` executable uses the Milter reference library, and is therefore multi-threaded. The `mimedefang` process is shown in cyan because it is the only multi-threaded process in CanIt-PRO; all others are single-threaded. The interface between `mimedefang` and `sendmail` may be a local (UNIX-domain) socket or a TCP socket.

`mimedefang` takes care of accepting e-mail headers and bodies from `sendmail` and writing them to a temporary spool directory (typically, `/var/spool/MIMEDefang`). It then sends short commands to `mimedefang-multiplexor`.

`mimedefang-multiplexor` listens on a UNIX-domain socket and manages a pool of Perl processes which do the actual filtering. The multiplexor has the following responsibilities:

1. It listens for requests from `mimedefang` and assigns them to one of the Perl processes.
2. It starts more Perl processes (up to a configured limit) if load increases.

3. During times of low load, it kills off Perl processes (down to a configured limit.)
4. It kills Perl processes which have processed a configured number of messages. This is done to avoid potential memory leaks.
5. It kills Perl processes which take too long to scan a message or which stop responding to requests.

`mimedefang.pl` is the actual Perl filtering program. It listens for requests (from the multiplexor) on its standard input, and writes results to its standard output. The commands and results exchanged are quite short; any modifications to the e-mail message are done in the spool directory.

Because the multiplexor manages several Perl processes, the Perl filters do not have to be thread-safe. In addition, the "pool-of-preforked-processes" architecture scales very well on SMP systems, and is efficient, robust and reliable.

B.3 Controlling CanIt-PRO

CanIt-PRO is started by a script called `/etc/mail/canit/mimedefang-ctrl`. This script is invoked with a single argument; possible arguments are:

`start` Starts CanIt-PRO.

`stop` Stops CanIt-PRO.

`restart` Equivalent to `stop` followed by `start`.

`reread` If CanIt-PRO is running, signals `mimedefang-multiplexor` to kill all idle perl filters, and to terminate busy filters once they become idle. In effect, this forces a reread of any configuration files.

`reload` A synonym for `reread`.

`status` Checks if CanIt-PRO is running. Exits with an exit code of 0 if CanIt-PRO is running, or 1 if it is not.

`/etc/mail/canit/mimedefang-ctrl` reads a shell script called `/etc/mail/canit/mimedefang.conf`. This script contains variable assignments. The meanings of the variables are described below. Boolean variables can take the values `yes` or `no`, while other variables are integers or strings.

MULTIPLEXOR (boolean) should always be set to `yes`.

START_TICKER (boolean) should be set to `yes` on the machine running the ticker, and `no` on all other machines.

MX_USER (string) should be set to the user ID of the `mimedefang` processors. Typically, this is a dedicated user called `defang`.

MX_RELAY_CHECK (boolean) enables filtering of relay IP addresses during SMTP connection.

MX_SENDER_CHECK (boolean) enables checking of the sender address in the SMTP "MAIL FROM:" command.

MX_RECIPIENT_CHECK (boolean) enables checking of the recipient address in the SMTP "RCPT TO:" command.

MX_LOG (boolean) enables logging of messages. This should always be set to `yes`.

MX_REQUESTS (integer) specifies how many requests each Perl filter will handle before being killed. The filters are killed after this number of requests to eliminate any possibility of problems due to memory leaks.

MX_MINIMUM (integer) specifies the minimum number of Perl filters to keep running, even if the system is idle.

MX_MAXIMUM (integer) specifies the maximum number of Perl filters to run concurrently, no matter how busy the system is.

Note that each Perl filter requires a database connection. The default installation of PostgreSQL permits only 32 simultaneous database connections. If you need more than this, you should increase the number of PostgreSQL back-ends with the "-N" and "-B" `postmaster` options when you start the database. Please see the **postmaster**(1) and **pg_ctl**(1) man pages for details.

MX_IDLE (integer) specifies how long in seconds a Perl process should be idle before it is killed off. After a period of heavy load, idle processes eventually get killed off until there are `MX_MINIMUM` Perl filters running.

MX_BUSY (integer) specifies how long in seconds a Perl filter is allowed to process a message. If the filter takes longer than this, it assumed to have hung up and is killed, and the message is tempfailed.

MX_CMD_TIMEOUT (integer) specifies how long in seconds to wait for commands and results to be transferred between `mimedefang` and `mimedefang-multiplexor`.

MX_SLAVE_DELAY (integer) specifies how long to wait after starting each Perl filter. If the system is idle, but fewer than the minimum number of filters are running, a new filter is started each `MX_SLAVE_DELAY` seconds.

MX_MIN_SLAVE_DELAY (integer) specifies that the multiplexor must not start slaves more quickly than the specified delay, no matter what. Even if the system is busy, a new filter will not be started more often than every `MX_MIN_SLAVE_DELAY` seconds. Setting this to 1 or 2 seconds may help your machine withstand a sudden surge in e-mail; it helps smooth out sudden load increases. However, it may cause delays as some mail is tempfailed.

MX_MAX_RSS (integer) specifies the maximum resident-set size in kB of each Perl filter process. On systems which support this limit, a Perl filter which exceeds this limit is killed.

MX_MAX_AS (integer) specifies the maximum virtual address space in kB of each Perl filter process. On systems which support this limit, a Perl filter which exceeds this limit is killed.

MX_STATS (boolean) specifies that the multiplexor should log statistical information in `/var/log/mimedefang/stats`.

MX_FLUSH_STATS (boolean) specifies that the multiplexor should flush `/var/log/mimedefang/stats` each time it writes a line to the file.

MX_STATS_SYSLOG (boolean) specifies that the multiplexor should log statistical information using Syslog.

MX_SOCKET (string) specifies the full path to the UNIX-domain socket used for communication between `mimedefang` and `mimedefang-multiplexor`.

LOG_TIMES_TO_SYSLOG (boolean) specifies whether or not to log filter times using syslog. If you set this to **yes**, then CanIt-PRO will log lines similar to this in your mail log:

```
gBNEeeI9004056: Filter time is 231ms
```

USE_MKDIR_MUTEX (boolean) specifies whether or not to protect temporary-directory creation with a mutex. Most systems should *not* require this. However, if you notice error messages in your mail log to the effect that MIMEDefang could not create a working directory for scanning, try setting this to **yes**.

MX_EMBED_PERL (boolean) specifies whether or not the multiplexor should use an embedded Perl interpreter. Normally, when a Perl slave is needed, the multiplexor `fork`s and the child `exec`s a Perl program. If you set this to **yes**, then the multiplexor uses an embedded Perl interpreter that reads the Perl filters only once. When a new slave is needed, only a `fork` is done. The overhead of the `exec` and the Perl interpreter initialization is avoided.

On some systems, it is not possible to embed a Perl interpreter. If you set this flag to **yes** on such a system, a warning is logged to syslog and CanIt-PRO continues as if the flag were **no**.

On some systems, it is possible to embed a Perl interpreter, but not to safely destroy it and create another interpreter in the same process. On such systems, a warning is logged if you force a filter reread. This will not affect the operation of CanIt-PRO, but if you edit the actual Perl filter file, you will need to do a (more expensive) `mimedefang-ctrl restart` rather than the cheaper `mimedefang-ctrl reread`.

B.4 Filter Settings

A few filter settings are stored in the file `/etc/mail/canit/filter-settings.pl` rather than in the PostgreSQL database. You should edit this file for your site; it is a Perl script, so you should follow Perl syntax. The important settings are:

$AdminAddress The e-mail address of the CanIt-PRO administrator. Enclose it in single-quotes, or the “@” sign will give Perl trouble.

$AdminName The full name of the CanIt-PRO administrator.

$DaemonAddress The e-mail address from which CanIt-PRO notifications appear to come. Enclose it in single-quotes.

$TempfailIfDatabaseDown If this is set to 1 (the default), then CanIt-PRO will tempfail mail if the PostgreSQL database server is non-responsive. If timely delivery of e-mail is more important to you than guaranteed scanning, set **$TempfailIfDatabaseDown** to 0. In that case, if the database is down, mail will be delivered un-scanned with a warning added in the `X-Spam-Score:` header.

$VirusHandlingIfDatabaseDown If you set **$TempfailIfDatabaseDown** to 0, this parameter controls how viruses are handled while the database is down. It must be set to one of **Reject**, **Discard** or **Accept**.

$WindowsExecutablesIfDatabaseDown If you set **$TempfailIfDatabaseDown** to 0, this parameter controls how Windows executables are handled while the database is down. It must be set to one of **Reject**, **Discard** or **Accept**.

B.5 Tuning CanIt-PRO

Tuning CanIt-PRO is a bit like tuning Sendmail: A black art. Nevertheless, we can offer some guidelines which should help improve the performance of your CanIt-PRO installation.

B.5.1 Memory

You CanIt-PRO server should have sufficient memory. As a rule of thumb, you should have about 16MB of memory for each concurrent Perl filter. If you set the maximum number of Perl filters to 16, for example, your machine should have about 256MB of physical memory.

Your CanIt-PRO server should also have sufficient swap space that a sudden flood of e-mail does not cause exhaustion of virtual memory. An additional 32MB of swap space for each Perl filter is probably a good rule of thumb.

B.5.2 Disk

You should have fast, reliable disks on your CanIt-PRO server. In particular, the CanIt-PRO spool directory (`/var/spool/MIMEDefang`) is heavily used, and it may be worth putting it on its own disk. Even better, put the spool directory on a RAM disk, assuming you have sufficient memory. A RAM-based CanIt-PRO spool directory is a large win, especially on systems like Solaris with relatively conservative file systems.

To calcluate the amount of RAM you'll need for the spool, multiply the size of the largest message you'll accept by the maximum number of concurrent filters, and then multiply by 3 as a safety factor for CanIt-PRO processing. For example, if you accept messages up to 3MB, and you'll have at most 8 Perl filters running, then your `/var/spool/MIMEDefang` space should be at least 72MB. If you use a RAM disk for the spool directory, add this memory to the memory requirements in the previous section.

B.5.3 Solaris-Specific tmpfs Note

Solaris is very conservative about committing writes to disk. On a busy Solaris server, consider it *mandatory* to put `/var/spool/MIMEDefang` on a RAM-based `tmpfs` file system. The performance improvement will be dramatic.

B.5.4 CPU

Spam-scanning is quite CPU-intensive, but in modern computers, the CPU is unlikely to be the bottleneck. If the CPU does prove to be a bottleneck, you should consider a faster machine, or even a multiprocessor machine.

B.5.5 Sendmail

Tuning Sendmail is quite complex; for a review of some of the issues involved, we recommend “Sendmail Performance Tuning” by Nick Christenson, Addison-Wesley, ISBN 0-321-11570-8.

B.6 Dealing with Overload

Normally, the resources which first become overloaded in a mail server are disk or network bandwidth. However, a server with CanIt-PRO installed is more likely to run out of CPU power or memory, simply because content-scanning is relatively expensive. If your CanIt-PRO machine becomes overloaded to the point that very little mail is flowing and the machine is struggling, here are tuning tips to help you recover.

B.6.1 Tune CanIt-PRO and Sendmail

In addition to the tuning tips in Section B.5, two parameters are particularly helpful in letting the CanIt-PRO server deal with overload: In `/etc/mail/canit/mimedefang.conf`, set `MX_MAXIMUM` to a fairly low number, around 5 or 6. On most hardware, this should limit the impact of scanning on CPU and memory. It will allow the CanIt-PRO machine to process incoming mail smoothly until the overload conditions abate.

In conjunction with `MX_MAXIMUM`, it is very useful to set Sendmail’s `ConnectionRateThrottle` option. If you set this to 3, for example, Sendmail will accept at most 3 SMTP connections per second. Again, this lets your machine process mail smoothly until overload conditions abate.

So if your server becomes overloaded, follow these recovery steps:

- Set `MX_MAXMIMUM` to 5, and `ConnectionRateThrottle` to 3. (If you use M4 to generate the sendmail configuration file, the M4 parameter is called `confCONNECTION_RATE_THROTTLE`.)
- Watch the load carefully. If your machine appears to have idle time and free memory on its hands, cautiously increase the parameters until throughput seems to be maximized.

B.6.2 Network Architecture

A good way to deal with temporary overload conditions is to have a secondary MX machine that simply relays mail without doing any scanning. It will queue messages that the primary machine cannot handle, and then deliver them serially to the primary machine, smoothing out the load. The disadvantage of this scheme is that some relay-IP tests do not work as effectively, and the secondary MX machine may have to generate bounce messages.

If your CanIt-PRO machine is overloaded a lot of the time, we suggest setting up a second equal-weighted MX machine with CanIt-PRO installed. The two CanIt-PRO machines can share the same PostgreSQL database, since database access is rarely the bottleneck. Having two equal-weighted MX records will spread the load over both machines.

Appendix C

CanIt-PRO HOWTOS

C.1 Restoring a Database from a Dump

The CanIt-PRO cron job makes a text dump of the entire database every night; the database is dumped into `/var/spool/Canit-Spam-DB-Backup/SPAM-DATABASE-BACKUP`. You should back this file up to ensure the integrity of your spam database.

If, for some reason, you need to restore the database from the text file, follow this procedure. Note that you may need to supply the full path to the PostgreSQL utilities like `pg_dump`, `psql`, `createuser`, etc.

All of these examples assume that the PostgreSQL superuser is named `postgres`. This is likely to be true on Linux and Solaris, but some platforms use `pgsql` instead (this is the setting in FreeBSD's port of PostgreSQL.)

1. Stop CanIt-PRO, the ticker, Sendmail and the CanIt-PRO Web interface. (You can disable the Web interface without stopping Apache by touching the file `/etc/mail/canit/disabled`.)

2. Dump your existing database, just to be safe. Be sure to do this in a directory with sufficient space:

   ```
   $ pg_dump -U postgres spam > spam-dump-file.txt
   ```

3. Drop the database:

   ```
   $ dropdb -U postgres spam
   ```

4. Create an empty database:

   ```
   $ createdb -U postgres -E sql-ascii spam
   ```

5. Restore the database contents from the nightly dump file:

   ```
   $ psql -U postgres -d spam < SPAM-DATABASE-BACKUP
   ```

6. Analyze the database to update statistics for the query optimizer:

   ```
   $ psql -U postgres -d spam -c 'ANALYZE VERBOSE'
   ```

Do not omit the ANALYZE step or your database will be very slow.

7. Restart CanIt-PRO, Sendmail and the CanIt-PRO Web interface. Remove the file `/etc/mail/canit/disabled` if you created it.

C.2 Moving CanIt-PRO to a Different Machine

If you need to move CanIt-PRO to a different machine, follow these instructions.

On the existing machine:

1. Stop CanIt-PRO, the ticker, Sendmail and the CanIt-PRO Web interface. (You can disable the Web interface without stopping Apache by touching the file `/etc/mail/canit/disabled`.)
2. Dump your existing database. Be sure to do this in a directory with sufficient space:

   ```
   $ pg_dump -U postgres spam > spam-dump-file.txt
   ```

3. Copy the *entire* directory tree rooted at `/var/spool/MD-Bayes` to the new machine, being sure to preserve ownership and permissions. There are various ways to do this depending on your operating system. However, in the common case in which the old and new machine both have `rsync` and `ssh` installed, one way to achive the copy is to run this command on the old machine:

   ```
   # rsync --archive -essh /var/spool/MD-Bayes new_machine:/var/spool
   ```

   You may wish to add the **--verbose** and **--progress** flags if you have a lot of data to copy.

On the new machine:

1. Install CanIt-PRO as usual, but stop *just before* you would normally run the `init-database` script.
2. Copy the `spam-dump-file.txt` file from the old machine.
3. Create the spam user using PostgreSQL's `createuser` command:

   ```
   $ createuser -U postgres spam
   ```

4. Create an empty spam database:

   ```
   $ createdb -U postgres -E sql-ascii spam
   ```

5. Restore the database contents from the dump file you copied over:

   ```
   $ psql -U postgres -d spam < spam-dump-file.txt
   ```

6. Analyze the database to update statistics for the query optimizer:

   ```
   $ psql -U postgres -d spam -c 'ANALYZE VERBOSE'
   ```

   Do not omit the ANALYZE step or your database will be very slow.

7. Continue with the `init-database` script. It will detect the existing spam database, and update the schema if necessary. (The schema may be updated if you install a newer version of CanIt-PRO on the new machine than the one running on the old machine.)

C.3 Using `canit-cmd`, the CanIt-PRO Command-Line Tool

CanIt-PRO includes a tool called `canit-cmd`, which lets you manipulate certain tables in the database from the command line. This lets you script things like addition or deletion of users, address mappings, domain mappings, and so on.

`canit-cmd` is invoked as follows:

```
$ canit-cmd command [args...]
```

The *command* specifies which action you want to take. The additional *args* may or may not be required, depending on the *command*.

For a full list of the available *command*s, invoke `canit-cmd` without any arguments:

```
$ canit-cmd
```

Appendix D

CanIt-PRO Logging

D.1 General Information

CanIt-PRO logs messages regarding its operation using syslog. By default, these are logged using the `mail` syslog facility to keep them together with Sendmail's logs. This is recommended, but if for some reason you wish to change it, you can do so by modifying the `$SyslogFacility` variable in `/etc/mail/canit/filter-settings.pl`.

In general, a CanIt-PRO log entry will consist (after the standard syslog preamble of date, host, process name, and process ID) of the word `CanIt:` followed by the 14 character Sendmail queue ID (or the text `NOQUEUE`) followed by another colon. After this comes the message-specific information for that log type.

Several types of log message are generated, at different log levels:

Debugging messages Debugging messages provide very verbose, detailed information regarding the internal workings of CanIt-PRO. These are logged using syslog's `debug` facility, and are turned off by default in shipped versions of CanIt-PRO.

You will probably never need to enable debug logging, but if you need to do so, you must edit the CanIt-PRO filter file (`/etc/mail/canit/canit-pro-filter`) and add the line:

```
CanIt::Logger::set_debuglevel( CanIt::Logger::DEBUG_ON() );
```

to the `filter_initialize()` function, and restart the CanIt-PRO service.

When enabled, debug logging provides extra debugging information. After the general log entry info mentioned above, a debug message consists of `DEBUG:`, the message itself, and then in parentheses, the line, file, function, and caller information for each debug message.

Note: Enabling debug logging is not recommended on a heavily loaded production server, as the extra syslog traffic will slow things down, and greatly increase the disk space required for your logs.

Regular log messages Regular log messages provide information about the normal operation of CanIt-PRO and are logged at the 'info' level.

Event messages Event log messages provide information about the normal operation of CanIt-PRO in a format that is both human readable and machine parseable. These are logged at the ’info’ level.

Warning messages Warning messages indicate that an undesirable, but non-fatal, condition has occurred. These are logged at the ’warning’ level.

Error messages Error messages indicate that a failure has occurred within CanIt-PRO and should be attended to immediately. These are logged at the ’error’ level.

D.2 Event Log Format

Event messages are logged in a format designed to be both human-readable and machine-parseable. This format consists of comma-separated key=value pairs, where the key consists of entirely lowercase alphabetic characters, and the value consists of arbitrary text appropriate for that key, with problematic characters such as newlines and commas replaced with a `%` followed by their two-digit hexadecimal value.

With the exception of `what`, which always appears first, and `subject`, which will appear last if present, the key/value pairs cannot be assumed to occupy any specific position in the log line. Depending on where and why the message was logged, different keys will be present.

An example log message is:

```
Jan 01 13:10:31 oxygen mimedefang.pl[9813]: CanIt: j4CHAVtu009864:
what=accepted, nrcpts=1, relay=192.168.10.8, score=2.5,
sender=user1@someremotehost.tld, stream=user1,
subject=Yes%2C this is an example
```

(We have wrapped the output for readability; in reality, the log message would appear on a single line.)

Here we see the standard date, time, hostname, process name, and process ID from syslog, the name `CanIt:`, the sendmail queue ID for the message being processed, and a number of key-value pairs separated by commas.

The keys that can appear in an “event” log line are:

what This field provides the first indication of what happened to the message. The ’reason’ and ’detail’ fields provide further information

Valid values for ’what’ are:

accepted Message was accepted and relayed through. The ’reason’ field may contain one of: approved, sender-whitelisted, domain-whitelisted, host-whitelisted, unscanned-toobigskip-spam-scan, opt-out, or no reason at all if none of those cases apply.

rejected Message (or sender, or recipient) was rejected and the sending relay was given a 5xx failure code. The ’reason’ field may contain one of: auto-reject, auto-reject-no-incident, blacklisted-recipient, domain-blacklisted, exe, ext, host-blacklisted, invalid-recipient, mime, mismatch-blacklist, rbl-blacklisted, sender-blacklisted, too-large, or virus.

tagged Message was tagged and relayed through. what=tagged log lines will not contain a ’reason’ field.

discarded Message was discarded silently. The 'reason' field can be auto-reject, auto-reject-no-incident, exe, ext, mime, virus.

greylisted Message was greylisted with a 4xx code. what=greylisted lines will not contain a reason field.

reason This provides secondary information (the "why" to the "what" above) regarding the disposition of an incoming connection. Valid values are:

approved Message was manually approved from the spam trap interface.

auto-reject Message was rejected. An incident is available and is indicated by the value for the `incident` key.

auto-reject-no-incident Message was automatically rejected due to spam score, and no incident was created.

blacklisted-recipient The specified recipient was blacklisted

domain-blacklisted The domain of the sender's address was blacklisted in the specified stream.

domain-whitelisted The domain of the sender's address was whitelisted in the specified stream.

exe The message contained a file with an extension considered executable on Microsoft operating systems. `detail` will contain the extension name.

ext The message contained a file with a blocked extension. `detail` will contain the extension name.

host-blacklisted The relay host was blacklisted in the specified stream.

host-whitelisted The relay host was whitelisted in the specified stream.

invalid-recipient The specified recipient was not valid.

mime The message contained a file with a blocked MIME type. `detail` will contain the actual MIME type found.

mismatch-blacklist The message triggered a mismatch rule.

opt-out The stream containing this message is configured to opt out of spam scanning.

rbl-blacklisted The relay sending this message was blocked by an RBL entry.

sender-blacklisted The sender address was blacklisted in the specified stream.

sender-whitelisted The sender address was whitelisted in the specified stream.

skip-spam-scan The originating relay was in a Known Network marked with "Skip Spam Scan"

too-large The message was rejected because it was over the configured maximum size for messages received. The `detail` key will contain the actual size of the message.

unscanned-toobig The message was not scanned for spam because it was over the configured maximum size for scanning. The `detail` key will contain the actual size of the message.

virus The message contained a virus payload. The `detail` key will contain the name of the virus found.

detail This provides further detail if necessary (and available) from certain tests. For example, if `what=discard` and `reason=virus`, the detail key will contain the name of the virus found.

incident The numeric ID of the incident, if available. An incident ID will be available only if an incident is associated with this message, either because it was created, or because the message matched an existing incident.

nrcpts The number of recipients for the given message. In general, rather than listing the individual recipients (which, in some cases could number in the hundreds), we use this key to provide only the number. The exception is when a particular single recipient is affected. In that case, we use the `recipient` key to log the actual address.

recipient If an envelope recipient is rejected for some reason, the recipient address is logged with this key.

relay The IP address of the sending relay. If parsing of Received: headers is enabled, this contains the address retrieved from the headers. Otherwise, the actual connecting relay IP is logged.

score The score for the message, if scoring rules were applied.

sender The envelope sender of the message.

subject The subject line of the message, if available. This key always appears last in the log message.

stream The name of the stream being applied to the message at the time.

Appendix E

Additional Scripts

CanIt-PRO ships with additional scripts that you may find useful. Please note that these scripts are *not* officially supported by Roaring Penguin Software Inc.

E.1 send-trapped-report.pl

The script `/etc/mail/canit/send-trapped-report.pl` is designed to run from `cron`(8). It sends periodic reports to all recipients letting them know which messages of theirs are in the trap.

To use the script, edit it and make the following changes:

- Set `$StatusList` as appropriate—determine whether you want reports sent for new and pending messages, or new, pending and spam messages.
- Set `$ReportDays` to a number of days approximately equalling the `cron` interval for running the script. For example, if you plan to run the script weekly, set `$ReportDays` to 7.
- To exclude certain addresses from receiving the report, adjust `$DontEmailMe`.
- If you only want certain people to receive the report (and no others), adjust `$OnlyEmailMe`.
- Adjust the remaining variables as indicated by the comments in the script.

To run the script, create an entry in *root*'s crontab. For example, to run it on Monday morning at 2:00am, use this cron entry:

```
0 2 * * 1  /etc/mail/canit/send-trapped-report.pl
```

Note that the trapped report includes message subjects, and may appear highly spam-like to CanIt-PRO. We recommend that you whitelist the host 127.0.0.1 if you use this script; otherwise, the reports themselves might end up getting trapped.

E.2 reset-password.pl

The script `/etc/mail/canit/reset-password.pl` lets you reset the administrator password if you forget it. To run the script, simply type:

```
# /etc/mail/canit/reset-password.pl
```

and follow the prompts.

Appendix F

Bayes Database Back-Ends

F.1 PostgreSQL Bayes Data Storage

By default, versions of CanIt-PRO prior to 3.2.0 store Bayesian statistics in the PostgreSQL database in a table called `bayes`. At a large site, Bayesian lookups can cause considerable database traffic and substantial load on the database machine. CanIt-PRO has a mechanism to store Bayesian statistics in Berkeley database files. These files are local to each scanner. Lookups are extremely fast, and involve no database traffic and no load on the PostgreSQL database. Similarly, updates do not involve the PostgreSQL database, which can greatly improve performance.

Note: As of CanIt-PRO version 3.2.0, the Berkeley DB back-end is the default storage mechanism for Bayes data. However, if you were running an older verision of CanIt-PRO and have upgraded, the upgraded version will continue to use the same Bayes back-end as the older version. We strongly recommend that all CanIt-PRO installations switch to the Berkeley DB back-end.

Note: As of CanIt-PRO version 3.3.0, the PostgreSQL back-end is no longer supported, and cannot be used.

F.2 Berkeley Database Bayes Storage

The Berkeley database storage of Bayes data operates as follows:

- The *master* database files are stored on the machine running the ticker. Each stream has its own database file under the directory `/var/spool/MD-Bayes/DB`.
- Bayes training is performed by the ticker. It updates the master Berkeley database files. If you are running a cluster, the ticker then copies the updated database files to each scanning machine.

As a consequence of the way the Berkeley database files work, you must be aware of the following:

- You must have sufficient room under `/var/spool/MD-Bayes/DB` for all of your Bayes data on the ticker machine *and* on each scanner.

- If you want to back up your Bayes data, you must back up `/var/spool/MD-Bayes` on the ticker machine *as well as* backing up the nightly database dump.
- The ticker machine must have a way to copy the database files to each scanning machine (see Section F.4.)

F.3 Switching to Berkeley Database Bayes Storage

To switch to the Berkeley database back-end, follow these steps:

1. On each scanning machine and on the ticker, ensure that the `defang` user account has the following properties:
   - Home directory is `/var/spool/MD-Bayes`.
   - Shell is a real shell (typically `/bin/sh`).
2. Decide if you wish to preserve the Bayes data currently stored in PostgreSQL.
   - If you do decide you want to keep it, you need to export the PostgreSQL data to Berkeley database files. To do this, run the `bayes-pg-to-dbfile.pl` script as root:

     ```
     # /etc/mail/canit/bayes-pg-to-dbfile.pl
     ```

     This script can take a considerable amount of time to run. You can run it while the system is live, however. You may lose some training data if you run it against a live system, but the amount of loss should be small and tolerable.

     Once the script has run, fix up the permissions. *Do not skip the `chown` command below!*

     ```
     # chown -R defang /var/spool/MD-Bayes/DB
     ```
   - If you do *not* care about keeping the Bayes data in PostgreSQL, simply move on to the next step.
3. If you are running a cluster, see Section F.4
4. As the administrative user, click on “Setup” and then “Bayes Database Wizard” in the Web interface.
5. Select the **Berkeley DB** format for Bayes storage.
6. Click **Next** to review your changes.
7. Click **Finish** to finish the wizard and make the changes take effect.
8. Once you are happy with the Berkeley DB back-end, you may delete the old Bayes data from the PostgreSQL database by connecting to the database as the ’spam’ user with:

   ```
   psql -U spam spam
   ```

   and executing the following SQL query:

   ```
   DELETE FROM bayes WHERE word != ’:TOTAL:’;
   ```

F.4 Cluster Considerations

Once you have dumped the Bayes data to Berkeley DB files, and set the permissions on `/var/spool/MD-Bayes/DB` appropriately, you need to copy the files to all your scanning machines. If you have `rsync` and `ssh` installed, the following commands can be used to copy the data over. They should be run as root on the ticker machine; we assume `$SCANNERS` is a list of all your scanners.

```
for mach in $SCANNERS ; do
    rsync -essh --archive --progress --verbose /var/spool/MD-Bayes/DB \
                                         $mach:/var/spool/MD-Bayes
done
```

F.4.1 Propagating Updates

Because the ticker can only update Berkeley databases locally on the ticker machine, a mechanism is required to copy updated files to all scanning machines. Whenever a Berkeley database file is updated, CanIt-PRO looks for a script called `/etc/mail/canit/sync-berkeley-db-multi` (On Solaris, this script might be called `/opt/RPSI/canit/bin/sync-berkeley-db-multi`). This script is executed with two arguments:

1. If the first argument is `delete`, then:
   - The second argument will be the absolute pathname of a Berkeley DB file.
   - The script should ensure that this file is deleted on all cluster members.
2. If the first argument is `copy`, then:
   - The second argument will be the absolute pathname of a Berkeley DB file.
   - The script should copy this file to the same location on all other cluster members.
3. If the first argument is `copyfrom`, then:
   - The second argument will be a filename containing a *list* of files to copy. Each line in the file will consist of the absolute pathname of a Berkeley DB file.
   - The script should copy *all* of the listed files from the ticker machine to all other cluster members, in as efficient a manner as possible.

We have provided a sample script called `sync-berkeley-db-multi-example`. This script uses `rsync` over `ssh` to copy the files, and `rm` and `ssh` to delete them. To use this script:

1. Ensure that each scanning machine and the ticker machine have the same machine architecture and same operating system. Berkeley DB files may *not* be compatible across different processor architectures, or even across different operating systems on the same processor.
2. Ensure that each scanning machine and the ticker machine have the `rsync` and `ssh` programs installed. Ensure that each scanning machine permits SSH access from the ticker machine.

3. On each scanning machine and on the ticker, ensure that the `defang` user account has the following properties:
   - Home directory is `/var/spool/MD-Bayes`.
   - Shell is a real shell (typically `/bin/sh`).
4. On the ticker machine, generate an SSH public/private key pair. Make sure *not* to use a passphrase. Install the private key in `defang`'s `.ssh` directory on the ticker machine. Install the public key on each scanning machine, as `defang`'s `.ssh/authorized_keys` file.
5. On the ticker machine, become the `defang` user and manually SSH into each scanning machine. The purpose of this is to populate the `known_hosts` file on the ticker, so that subsequent SSH sessions can proceed without user intervention.
6. Copy the `sync-berkeley-db-multi-example` file to `sync-berkeley-db-multi` and make it executable. Edit the file and set `@SCANNERS` to a space-separated list of scanning machines. (Do *not* include the ticker machine in this list.) Each time a Berkeley DB file is updated, the ticker will copy it to each machine listed in `@SCANNERS` using rsync-over-SSH.

If you do not want to use the sample script to copy the Berkeley DB files (perhaps because you don't want passphrase-less SSH sessions, or because the scanners and ticker machine are not all the same architecture), you can write your own script. It must have the following properties:

- It is invoked as `sync-berkeley-db-multi` *`action filename`*, where *action* is on of `copy`m `copyfrom` or `delete` and *filename* is the name of the file to copy or delete (or in the case of `copyfrom`, the file containing the list of files to copy.)
- For *each* scanner machine:
  - The script should first create any needed directories for the destination Berkeley database file.
  - The script should copy the file to a *new* file on the remote machine, in the same directory as the existing Berkeley database file (if any).
  - After successfully copying the file, the script should atomically rename the newly-created file to the proper destination filename.
- The script must *not* print anything to standard output. Anything that appears on standard output will be interpreted as master/slave communication by the MIMEDefang multiplexor, and will likely result in termination of the ticker slave.

F.5 Switching back to PostgreSQL Bayes Storage

As of CanIt-PRO 3.3.0, it is not possible to switch back to the PostgreSQL storage module for Bayes data.

Appendix G

The CanIt-PRO License

READ THIS LICENSE CAREFULLY. IT SPECIFIES THE TERMS AND CONDITIONS UNDER WHICH YOU CAN USE CANIT-PRO

This license may be revised from time to time; any given release of CanIt-PRO is licensed under the license version which accompanied that release.

CanIt-PRO is distributed in source code form, but it is not Free Software or Open-Source Software. Some CanIt-PRO components are Free Software or Open-Source, and we detail them below:

The following files may be redistributed according to the licenses listed here. An asterisk (*) in a file name signifies a version number; the actual file will have a number in place of the asterisk.

File	License
`src/DB_File-*.tar.gz`	Perl License
`src/Digest-MD5-*.tar.gz`	Perl License
`src/Digest-SHA1-*.tar.gz`	Perl License
`src/File-Spec-*.tar.gz`	Perl License
`src/HTML-Parser-*.tar.gz`	Perl License
`src/HTML-Tagset-*.tar.gz`	Perl License
`src/IO-stringy-*.tar.gz`	Perl License
`src/MIME-Base64-*.tar.gz`	Perl License
`src/MIME-tools-*.tar.gz`	Perl License
`src/Mail-SPF-Query-*.tar.gz`	Perl License
`src/Mail-SpamAssassin-*.tar.gz`	Apache License, Version 2.0
`src/MailTools-*.tar.gz`	Perl License
`src/Net-CIDR-Lite-*.tar.gz`	Perl License
`src/Net-DNS-*.tar.gz`	Perl License
`src/DBI-*.tar.gz`	Perl License
`src/DBD-Pg-*.tar.gz`	Perl License
`src/Time-HiRes-*.tar.gz`	Perl License
`src/URI-*.tar.gz`	Perl License
`src/clamav-*.tar.gz`	GPLv2
`src/mimedefang-*.tar.gz`	GPLv2

ALL REMAINING FILES IN THIS ARCHIVE (referred to as ”CanIt-PRO”) ARE DISTRIBUTED

UNDER THE TERMS OF THE CANIT LICENSE, WHICH FOLLOWS:

THE CANIT LICENSE

1. CanIt-PRO is the property of Roaring Penguin Software Inc. (”Roaring Penguin”). This license gives you the right to use CanIt-PRO, but does not transfer ownership of the intellectual property to you.

2. CanIt-PRO is licensed with a limit on the number of allowable protected mailboxes. The limit on the number of mailboxes is referred to as ”the Usage Limit”.

   CanIt-PRO usage may be purchased on a yearly basis, or you may purchase a perpetual license.

3. You may use CanIt-PRO up to the Usage Limit you have purchased. If you have purchased yearly usage, you may continue to use CanIt-PRO until your purchased usage time expires, unless you purchase additional time. If you have purchased a perpetual license, you may continue to use CanIt-PRO indefinitely, providing you do not violate this license.

   If you have purchased yearly usage, you may exceed your purchased mailbox limit by up to 10% until the yearly renewal date, at which time you must purchase a sufficient limit for the increased number of mailboxes.

   If you have purchased a perpetual license, or wish to increase your usage more than 10% above your paid-up limit, you must purchase the additional usage within 60 days of the increase.

4. You may examine the CanIt-PRO source code for education purposes and to conduct security audits. You may hire third-parties to audit the code providing you first obtain permission from Roaring Penguin. Such permission will generally be granted providing the third-party signs a non-disclosure agreement with Roaring Penguin.

5. You may modify the CanIt-PRO source code for your own internal use, subject to the restrictions in Paragraph 9 below. However, if you do so, you agree that Roaring Penguin is released from any obligation to provide technical support for the modified software. If you wish your modifications to be incorporated into the mainstream CanIt-PRO release, you agree to transfer ownership of your changes to Roaring Penguin.

6. You may make backups of CanIt-PRO as required for the prudent operation of your enterprise.

7. You may not redistribute CanIt-PRO in source or object form, nor may you redistribute modified copies of CanIt-PRO or products derived from CanIt-PRO.

8. If you violate this license, your right to use CanIt-PRO terminates immediately, and you agree to remove CanIt-PRO from all of your servers.

9. Restrictions on modification:

   (a) Notwithstanding Paragraph 5, you may not make changes to CanIt-PRO or your software environment which would allow CanIt-PRO to run without a valid License Key as issued by Roaring Penguin. You also agree not to set back the time on your server to artificially extend the validity of a License Key, or do anything else which would artificially extend the validity of a License Key.

(b) You may modify the Web-based interface only providing you adhere to the following restrictions:

(c) At the bottom of every CanIt-PRO web page, the following text shall appear, in a size, color and font which are clearly legible:

Powered by CanIt-PRO (Version x.y.z) from Roaring Penguin Software Inc.

where x.y.z is the product version. In addition, "CanIt-PRO" shall be a clearly-marked hypertext link to `http://www.roaringpenguin.com/powered-by-canit.php`

(d) You may not include elements on the CanIt-PRO Web interface that require plug-ins (such as, but not limited to, Macromedia Flash, RealPlayer, etc.) to function.

(e) You may not include Java applets on the CanIt-PRO Web interface.

(f) If you include JavaScript on the Web interface, you shall ensure that the interface functions substantially unimpaired in a browser with JavaScript disabled.

(g) You shall not include browser-specific elements on the Web interface. You shall ensure that the Web interface functions substantially unimpaired on the latest versions of the following browsers:

- Internet Explorer for Windows
- Mozilla for Windows
- Mozilla for Linux
- Konqueror for Linux

(h) You may not include banner ads on the CanIt-PRO Web interface.

10. Restrictions on reselling services:

Unless you purchased CanIt-PRO as a service provider on the ISP rate plan, you may not use CanIt-PRO to provide spam-scanning services to third parties. You may use CanIt-PRO only for your employees and contractors accounts on your own corporate servers.

11. Disclaimer of Warranty (Virus-Scanning)

NOTE: ALTHOUGH CANIT-PRO IS DISTRIBUTED WITH CLAM ANTIVIRUS, WE DO NOT MAKE ANY REPRESENTATIONS AS TO ITS EFFECTIVENESS AT STOPPING VIRUSES. ROARING PENGUIN HEREBY DISCLAIMS ALL WARRANTY ON THE ANTI-VIRUS CODE INCLUDED WITH CANIT-PRO, OR WHICH INTERFACES TO CANIT-PRO. WE ARE NOT RESPONSIBLE FOR ANY VIRUSES THAT MIGHT EVADE A VIRUS-SCANNER INTEGRATED WITH CANIT-PRO.

Index