福井大学
工学部　研究報告
第48巻　第1号
2000年3月

インターネットメールの脆弱性と新しいメール配送プログラムの提案

近藤 岳大*　西野 順二**　小高 知宏**　小倉 久和**

The Weakness of the Internet Mail System and a Proposal of New Mail Transfer Agent

Takehiro KONDOH, Junji NISHINO,
Tomohiro ODAKA and Hisakazu OGURA

(Received Feb. 29, 2000)

This paper presents a framework of implementing Mail Transfer Agent(MTA) for Small Office/Home Office and aims at the improvement of the security of e-mail server. We simplify the setting file of our MTA to reduce setting errors, because difficult settings cause setting errors and occur security holes. This MTA has limits relaying function of mails in order to prohibit SPAM mail relay. And we limit a size of received mail for the counter measures of mail bomb. Experimental results indicate that, this MTA has high security. And the MTA was evaluated that the settings of the MTA are very easy, and that this MTA is convenient in spite of the minimum functions.

Key Words : Mail Transfer Agent, Network Security

1　はじめに

本研究は、インターネットにおけるセキュリティを今一度見直すことを目的とし、特に小規模ネットワーク向けに、設定を容易にして設定ミスによるセキュリティホールの発生を押さえることを目的としたメール配送用ソフトウェア開発を検討した.

インターネット電子メールのサーバは、配送用ソフトウェアの歴史が古く広く使われているために、攻撃の対象となることが多く、同時に設定ミスからのセキュリティホールも発生しやすい. また、電子

*大学院情報工学専攻
**工学部知能システム工学科

メールのサーバに対する攻撃というのは、設定が困難なためか、大学の研究室などと言った専門の管理者がいないような小規模ネットワークにおいて、頻繁に見られている [1].

本研究では、メールを配送するためのソフトウェアとして最小限の機能を持たせるということを開発の主眼としている. 小規模ネットワークで運用したときに使われていないと思われる機能を削除し、メールを配送する機能のみを持たせるということである. これは、同時に 2 つの意味を持つ. まず 1 つ目には、機能を最小限にすることによって、当然の事ながら機能に関する設定項目が減り、設定を容易にすることができる. 設定が容易であれば設定ミスが減り、それから発生するセキュリティホールを抑えることが可能となる. 2 つ目としては、ソフトウェア全体が小さなプログラムとなるため、開発上でバグが入りにくくなる. サーバ用のソフトウェアのバグは、時に重大なセキュリティホールとつながることがある. さらに本研究では、メールを配送する本体を役割ごとに複数のプログラムに分割し、それぞれを独自の権限で動作させる. これにより、万が一セキュリティバグが発見された場合でも、権限が最小限であるため、その被害がシステムの他の箇所まで及ぶことがないようにする. このようにして、機能面、設定面、開発面からセキュリティを意識したメール配送用ソフトウェアを開発し、セキュリティレベルの高いメールサーバの構築・運用を目指す.

2 現行 MTA の問題点とその対応

電子メールは、MTA（Mail Transfer Agent）と呼ばれる電子メール配送用のプログラムによって送受信されている. その MTA として、もっとも有名であり多くのホスト上で稼働しているのが、sendmail である. この sendmail は歴史が古く、バージョンアップに伴う機能が柔軟で豊富である. また、よく使われているだけに書籍やホームページなどから、関連する情報を見つけることが容易である.
しかしながら、機能が豊富であるがゆえにそれが欠点となり、sendmail の設定はきわめて困難であると言うことがよく知られている. そして、その設定が一つのファイルに納められており、設定そのものが特有の言語で書かれているため、完全に理解することは設定すること以上に困難である.

この設定の困難性は設定ミスによるセキュリティホールを発生させることになり、そのため攻撃の対象となったり、攻撃の踏み台にされることが多い. その上、歴史が古く、よく使われているだけに、攻撃方法も良く知られており、攻撃の対象となりやすい. この攻撃の対象となるのが、SOHO（Small Offilce/Home Office）や大学の研究室などと言った、サーバをたてていながらも専門の管理者がいないような小規模のネットワークである. このようなネットワークでは、セキュリティに対する意識が甘いことが多く、バージョンアップが遅かったり、修正パッチを当てていないことも多い. また、sendmail の豊富な機能の中には、小規模のネットワークでは使われていない機能や全く知られていない機能まである. そのためこのような機能がセキュリティホールになった場合に発見するのが遅れ、攻撃の原因となりやすい.

そこで本研究では、ネットワークセキュリティの中でも電子メールのホストに関するセキュリティに関してとくに重点を置いた. 小規模のネットワーク向けに対して、機能を必要最小限に押さえることによって設定を容易にし、そこからセキュリティホールが発生しないような MTA を提案する.

2.1 電子メールにおける攻撃の種類と対処

2.1.1 SPAM メール

メールの中継機能を利用した攻撃に SPAM メールがある. SPAM メールとは、大量のねずみ講の勧誘などと言った迷惑メールの総称である. SPAM メールが騒がれるようになった 1995 年以前は、配送先／配送元によらず無条件にメールの中継を行っていた. しかし、このことが SPAM メールの横行を助けている可能性があるという見方がでるようになり、事実、SPAM メールを送る攻撃者は不正中継を利用している場合が多い（図 1）[3]. SPAM メールの中継ホストになると、まずホストのディスク容量や CPU と言った資源が不正に使用されることになる. また、SPAM メールを受け取ったターゲットが中継ホス

トに対して抗議のメールを送ったとき、それが多量である場合、次に説明するメールボムになることもある. 時には、SPAM メールを送るホストとしてブラックリストに載せられ、メールが送付を拒否されてしまう事態にも陥る場合もある（図2）. そのため現在では、本当に必要な中継以外を排除するのが一般的となりつつある. あるいは、前述のようにSPAMを送出し続けるホストを集めたブラックリストを作成し、それをもとにメールの受付を拒否する場合もある.

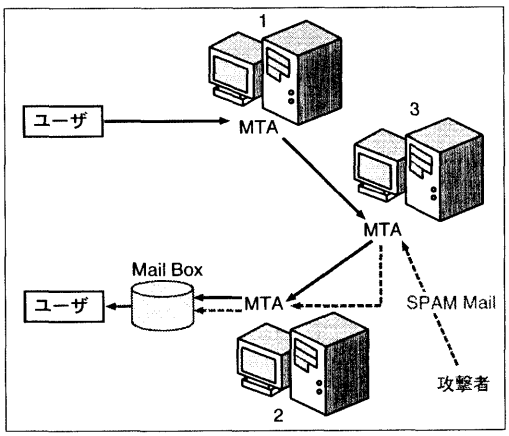

図 1: 不正中継を利用した SPAM メールの送信

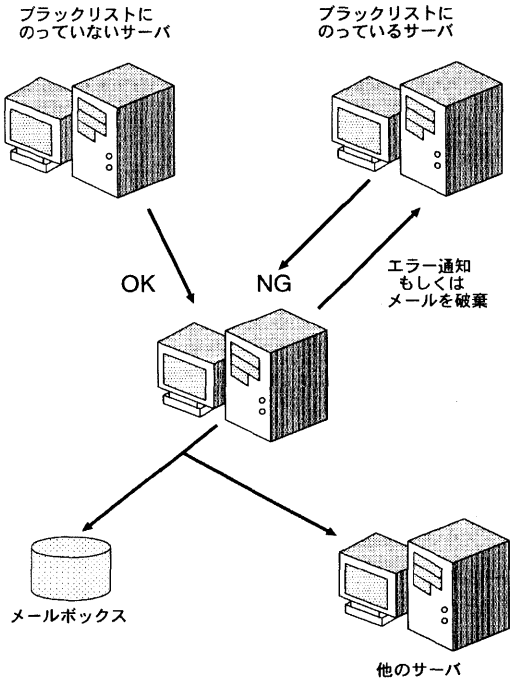

図 2: 特定ホストからのメールを拒否する

2.1.2 メールボム

sendmail は、基本的に受け取るメールのサイズに制限がない. したがって、意図的に巨大なメールが送信されたとき、メールスプールがあふれ、場合によってはホストがダウンしてしまう可能性がある. また、同一ホストもしくは複数ホストから、一つ一つのサイズは小さいが、何百通何千通という数のメールを送るもある. このようなメールをメールボムという.

2.1.3 攻撃への対処

メールの中継には、ドメイン内ホストからドメイン内ホスト及びドメイン外ホストへという経路と、ドメイン外ホストからドメイン内ホスト及びドメイン外ホストへという4つの経路が存在する. ここで、SPAM メールの中継に利用されるのは、ドメイン外ホストからドメイン外ホストの中継である. このことから、送信者及び送信先が自ドメインに関係しないメールに関しては一切中継を行わなければ、SPAM メールの中継を拒否することが可能となる.

メールボムに対して、まず、受信できるメールのサイズに制限を設けることによって、巨大なメール送付による攻撃には対処することが可能であると考えられる.

多数のメールによるメールボムへの対処法としては、例えば5秒間に20通以上のメールを送られてきたというような、一定時間中に一定量のメール送付があったホストからの受信拒否を自動化すること

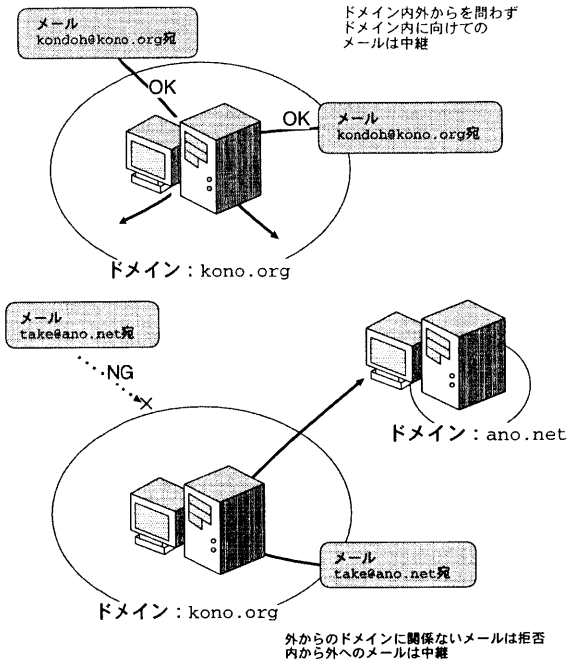

図 3: メールの中継を制限する

が考えられる.

2.2 本研究における MTA の仕様

本研究では、セキュリティを意識した上で MTA の設計・開発を行っている. そこで、どのような仕様に基づいているのかについて詳しく述べる.

本研究における MTA の仕様としては、3 つのポイントを上げることが出来る. まず一つ目として、大学の研究室や SOHO（Small Office/Home Office）などと言った小規模ネットワーク向けと言うことである. 攻撃の対象となるのが、サーバをたてていながらも専門の管理者がいないようなこれらのネットワークであることが多いからである. このようなネットワークでは、小さなところだから狙われるはずもない、とセキュリティに対する意識が甘いことが多く、また管理を専門にしていないために設定ミスなども引き起こしやすい. 本研究では対象を小規模ネットワークと限定することによって、機能を必要最小限におさえることができ、それによって設定を容易にすることが可能となり、専門の管理者でなくても設定ミスがでないようにする.

二つ目としては、MTA としてメールを送受信するための最小限の機能のみを実装することである. sendmail の豊富な機能の中には、小規模のネットワークでは使われていない機能や全く知られていない機能まである. その上、これらの機能に関しても設定をしっかり行う必要があり、そうしない場合はセキュリティホールになる可能性がある. そして、このような機能がセキュリティホールになった場合に発見するのが遅れ、攻撃の原因となりやすい. また、機能を豊富にすることによってプログラム的にも難しくなり、バグが増える可能性がある. 本研究においては、MTA としての最小限の機能のみを持たせることによって、設定項目を必要最小限に押さえ、設定を容易にする. また、機能が最小限であるためプログラム的にも簡単になり、バグを減らすことが出来る.

最後に、メールを配送する機能ごとに複数のプログラムに分割すると言うことである. sendmail が

MTA として全て 1 つのプログラムで、また root 権限で動作しているのに対し、本研究では SMTP 接続要求を処理する部分、メールをキュー登録する部分、配送する部分など、機能ごとに複数のプログラムに分割、それぞれを独自の権限で動作させることにした. このようにすることによって、開発上でバグが入るのを少しでも押さえることが出来るのではないかと考えている. また、権限をそれぞれのプログラムごとで制限することによって、万が一、MTA を通じて侵入されても root 権限の奪取やファイルの改竄などを防ぐことが出来るのではないかと考えている.

現行攻撃への対処法も開発段階で実装している. SPAM メールの中継サーバに利用されることへの対処法として、本 MTA はインターネットの末端に位置する小規模ネットワーク向けであるため、ドメイン内宛のメール及び、ドメイン内から外へのメール中継以外は行わない（図 3）.

サイズの大きいメールボムに対しては、受信できるメールのサイズを制限することによって対処した. 多数のメール送付によるメールボムへの対処法としては、発信ホストのブラックリストへの自動登録が考えられるが、活発なメーリングリストに登録しているユーザが複数人そのホストにいた場合、この方法を使うことはできず、実用には至っていない. そこで本 MTA では、受信を拒否するホスト名を手動で設定するようにしている.

2.3 他 MTA との比較

本研究での MTA と sendmail と qmail をセキュリティと関係させて比較すると表 1のようになる.

表 1: セキュリティ対策の比較表

	メールボム対策	SPAM メール対策	外部プログラム実行への対策	root 権限の保護
本 MTA	△	○	○	○
sendmail	×	×	△	×
qmail	△	△	○	○

まず、メールボムへの対策として、導入直後では研究での MTA や qmail は最初から上限を設けてあるが、メールを短時間に大量に送りつけるメールボムに対しては対策が出来ていないため△となっている. sendmail は、受け取るメールのサイズに上限がないため×となっている.

次に不正中継を利用した SPAM メールへの対策では、本研究での MTA は中継に関する設定は必要なく、不正中継に利用されることはないため○となっている. qmail は、導入直後では一切メールの中継を行わないが、実際の運用に当たってはドメイン内からのメール中継は行う必要があると考え、△とした. sendmail は中継に関する設定を行わないと、先ほどの例のように SPAM メールの中継サーバに利用されるため×となっている.

sendmail はバージョンによっては.forward に記述したり、telnet で直接 SMTP 交信を行うことによって外部プログラムを動作させられることがあるため△となっているが [2]、本研究での MTA や qmail はあらかじめその対策がとられていおり○となっている.

MTA の動作権限は、本研究での MTA や qmail は MTA 独自の権限をもつことから○となっているのに対し、sendmail は root 権限で動作し MTA 固有の権限を持たないため×となっている.

3 提案する MTA の構造

本章では、本研究における MTA のメール処理における特徴や、その設定方法について述べる. それにあたり、まず MTA がどのようにメールを処理しているか、それを図 4に示す.

まず、クライアントからのSMTP接続要求があり、接続が確立する. そして、SMTPにおけるコマンドのやり取りが行われ、メールを受信しキューに登録する. その後、キュー登録されたメールの宛先を判断し、それが自ホストのユーザ宛の場合はそのユーザのメールボックスに格納する. 宛先が他ホスト宛の場合は、そのホストもしくはそのホストに繋がる別のホストとSMTP交信を行いメールを配送する.

この基本的な動作はどのMTAでも同じであるが、本研究でのMTAは図の接続とメール受信の処理に着目し、工夫を凝らした.

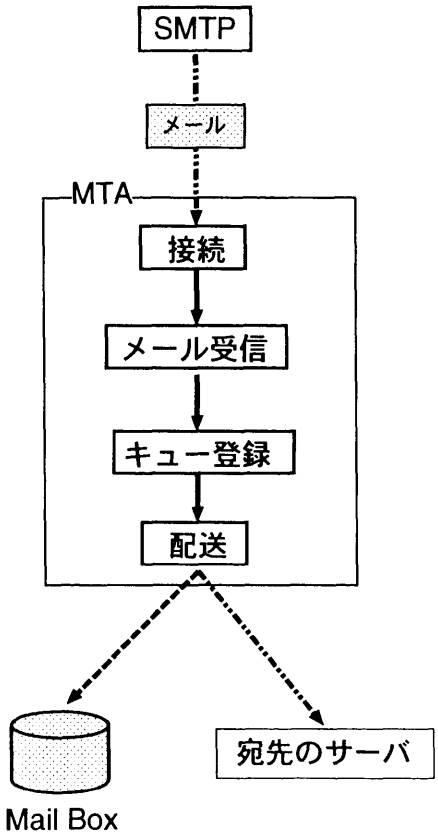

図4: メールの処理

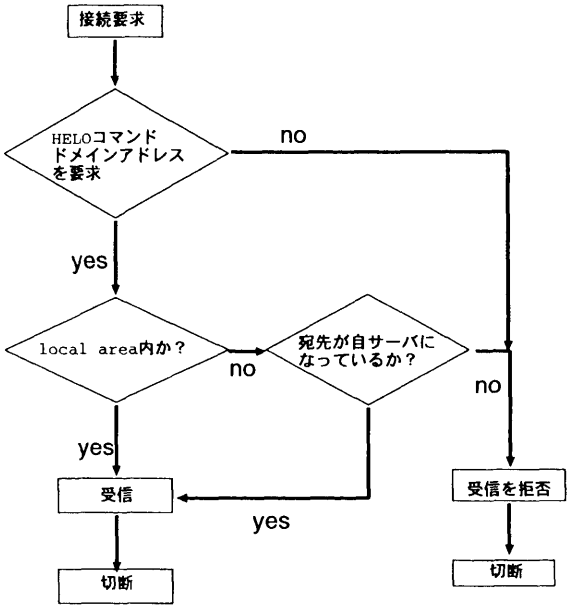

図5: smtpdの動作

3.1 内部動作

本研究におけるMTAの接続処理並びにメール受信処理について具体的に述べる. 本研究におけるMTAにおいては、smtpdというプログラムがこれらの処理を行う.

smtpdはクライアントから接続要求がありそれが確立すると、SMTPのHELOコマンドとクライアントのドメインアドレスを要求する. ここでこれらを受信できない場合は、そのクライアントからとの接続を切断する. HELOコマンドとドメインアドレスができた後は、そのドメインアドレスを判断する. クライアントのドメインアドレスが自ドメイン内の場合は、そのまま接続を続け、送信者、宛先、メール本体と受信し切断する. クライアントのドメインアドレスが自ドメインでない場合、つまり他ドメインのときには、次に送信者、宛先の順に受信し、その宛先を判断する. 宛先が自ドメイン内ホストである場合は、メール本体を受信し切断する. 宛先が自ドメイン内ではない場合は、受信を拒否し切断する（図5）. また、ブラックリストとしてホストやドメインが指定されていた場合、そこからの接続はドメイン

アドレスを受信した段階でメールの受信を拒否する.

smtpd が受理する SMTP コマンドとしては、RFC821 で必須とされるコマンド、及び RFC1123 で必須とされたコマンドを実装している [4][5]. ただし、VRFY コマンドはユーザ情報を求めるコマンドのため、実装してあるものの実際には、問い合わせのあったユーザがいるかどうかだけを返答する.

そのほかの機能は普段使われれておらず、またセキュリティ保護のためから実装していない. また、これらのコマンドを削除することでプログラムをより簡単に出来る.

3.2 設定方法

設定に関するファイルは、configfiles/というディレクトリに全て納められる. ここで必須とされるのは、servername、postmaster という 4 つのファイルである. この 2 つの他には、denyfrom というファイルを作り、設定することが出来る.

servername は、例えば mail.kono.org のように、MTA を稼働させるホストの完全修飾ドメインを書き込む. ここで指定されたホスト名は、以下のように用いられる.

- 送信者に対してエラーメールを送信する際のホスト名
- Message-ID 作成の際のホスト名
- SMTP 接続要求があったときの応答のとき
- remote が SMTP 接続したときのホスト名

postmaster は、メールが配送できなかったなどといったエラー発生時に、その旨を送信する管理者のメールアドレスを指定する. 通常の MTA ではこのような設定はないのだが、本研究における MTA では alias 機能を完全に排除したため、設定する必要がある.

denyfrom は、受信を拒否するホスト名を書き込むブラックリストの設定ファイルである. また、db.ano.net とホスト名を含めた FQDN で設定した場合は、db.ano.net からのメールのみを拒否し、mail.ano.net からのメールは受信する. ここで、ano.net と指定した場合は、そのドメインを含む全てのホストからのメールを拒否する. このファイルが空、もしくは存在しない場合は自ホスト宛のメールは拒否しない.

4 動作検証と評価

本研究で開発した MTA は、あらかじめ現行攻撃への対策が実装されているが、実際そのように動作するか検証を行った. また、開発した MTA を実際の SOHO 環境に導入し、設定方法や動作などについて評価を受けた.

4.1 メールの中継に関する検証

本研究における MTA は、SPAM メールの中継サーバとして利用されることがないように、メールの中継を制限している. そこで、開発した MTA が稼働しているホストをクライアントの SMTP サーバとして指定し、ドメインの外から他ドメインに向けてメールの送信を試みた. その結果、クライアントの MUA（Mail User Agent）には受信を拒否するエラー通知が表示され、ドメイン外から他ドメイン宛のメールは中継されなかったことを確認した.

これにより、他ドメインからさらに別のドメイン宛のメールの中継は行わず、SPAM メールの中継サーバに利用されることはない.

4.2 メールボムに対する検証

メールボムに対する検証は、サイズの大きいメール送付によるメールボムと多量のメールによるメールボムの二つについて検証を行った.

サイズの大きいメールを送付するメールボムの検証として、まずメール本文が600KByteあるメールの送付を試み、次にメールにファイルを一つ添付し、全体で約1MByteのメール送付を試みた. 結果としては、両方ともMTAに組み込まれているデフォルトの値である512KByteの大きさを超えているため、メールの受信を拒否した.

大量のメールを送付するメールボムの検証として、、約2秒間に100通のメールをクライアントから送付した. その結果として、MTAを稼働させているホストの動作が遅くなった.

これらの結果により、サイズの大きいメールに対しては問題がないが、大量のメールを送信させるメールボムに対しては、問題が生じることが分かった.

4.3 評価

本研究で開発したMTAを、実際にSOHOでメールサーバを管理しているシステム管理者に協力を依頼し、試験的に導入し評価を受けた. 導入期間は一週間である. 評価の対象としては、以下の通りである.

- 設定が簡単かどうか
- 動作は安定しているか
- 機能に関しては問題ないか

まず、導入後の設定は簡単であるという評価を受けた. sendmailを設定するときは、直接設定ファイルであるsendmail.cfを編集する方法、付属のcfというツールを使う方法、さらにCFというツールを使う方法がある. 直接編集したり付属のcfを使う場合は設定が難しく、CFを使う場合にはWWW上からダウンロードしてきた上で再構成が必要である. しかし本研究でのMTAは、別途のツールを必要とせず、簡単に設定が行えたということだった. 特に、sendmailやqmailで必要であるメールの中継に関する設定に関しては、導入段階でドメイン外から他ドメイン宛のメールを拒否するため設定する必要がないということが、好評であった. ただ、受信できるメールのサイズに上限があるのは良いが、できればそれを自由に設定できた方が良いという評価も同時に受けた.

メールの送受信は問題なく行われた. 稼働させたコンピュータは、Pentium133MHz、メモリ48MByte、OSがVineLinux1.1で、安定して動作したということだった. 導入期間内に送受信したメールの数は約1000通である.

ただ、機能があまりにも最小限であるという、否定的な意見も受けた. 特に、alias機能がないのは不便であるということだった. alias機能があれば、私的なメールと公的なメールを一つのアドレスで管理が出来るという意見だった. また、alias機能を実装することによって設定ファイルの一つである`postmaster`が削除できるのではないか、と言う意見も受けた.

また、ブラックリストに関する設定として、可能であるならば自動化して欲しいという要望も受けた. Web上において、SPAMメールを送信しているサーバがブラックリストして公開されているため、それにリストアップされているサーバを自動的に登録は出来ないものだろうか、という意見だった.

5 考察と今後の課題

本研究では、小規模ネットワーク向けに電子メールサーバ用ソフトウェアを開発した. また、専門の管理者でなくても管理ができるように、機能を必要最小限にして、設定をできる限り簡単にした.

5.1 導入に関する考察

現段階では、インストール方法や稼働しているMTAからの移行方法、実際の機能に関することといったマニュアル面が一切作成されていない. MTAの移行をスムーズに行うことによってサーバが停止する時間を最小限におさえ、またトラブル発生時に素早い対応ができるような、わかりやすいマニュアルを作成する必要がある.

5.2 機能及び設定に関する考察

本研究では、MTAとしての機能をメールを配送することだけに着目して、最小限の構成にした. そのため、現段階において、本研究でのMTAはalias機能をサポートしていない. ただ、alias機能を持たせることによって、設定のひとつである`postmaster`を削除できる. また、インストール用のスクリプトを作成し、その引数にホスト名を指定するか、起動時に引数としてホスト名を指定してやることによって、設定ファイルそのものをなくすことはできないかと考えている.

また、本研究でのMTAは受信メールのサイズに制限があり、それが固定されてしまっている. 受信側が承諾した上で、やるべきではないとしても、サイズの大きいメールを送信する可能性がある. この場合のことを考えると、受信サイズ制限を設定できた方がよいのかもしれない.

5.3 配送性能に関する考察

実験の結果から、自ホスト宛の10KByteのメールを配送するのに、1通あたり約23msの時間がかかることがわかった. これを1日あたりに換算すると、36万通のメールを処理できることになる. Internet Service Providerなどのようなユーザの多いホストの場合は、より多くのメールを処理する必要があるかもしれないが、ネットワークの末端である小規模ネットワークでは、十分であろうと考えられる. また、一般的なメールのサイズは2〜6KByteであることから、試算した数よりもより多いメールを処理できると考えられる.

リモートホストへの配送に関しては、ローカル配送のときと同様に1通あたり22ms〜23msで配送が出来た. しかしながら、ここでの配送実験は100Base-TXという高速LAN環境下での実験であり、実際にはより回線容量の小さい状況下で配送が行われ、さらにはリモートホストのサービス状況によって左右される可能性があり、配送速度は遅くなるものと考えられる.

また、現段階ではメールを配送するために、キュー登録やリモート配送用などのプログラムが1つだけ動作する. そこで、各処理のプロセスを同時に複数動作させることによって、配送効率を挙げることが可能である. その場合、プロセス管理やメモリ管理を今以上に厳しく行う必要がある.

5.4 セキュリティに関する考察

まず、SPAMメールの中継サーバにならないように、ドメイン外からのメールの中継を行わないことによって対処した. また、実際に他サーバ宛のメールを中継させ、それを拒否することを確認した. しかし、ドメイン内から外へのメール中継は行うため、IP-SpoofingのようにIPアドレスを偽造することによって、SPAMの中継サーバに利用される可能性がある. しかし、これはIPというプロトコルの問題であって、MTAそのものでは解決することは困難である.

メールボムに対しては、まず、サイズの大きいメールは、受信するメールのサイズを制限することによって対処し拒否することを確認した. ただ、配送実験の時には、BCC（Blind Carbon Copy）に同一アドレスを記述することによって50通もしくは100通としたが、場合によってはこれをメールボムとして利用される可能性もある. そこで、同一のMessage-IDを持つメールを、同一アドレスに2通以上届けるようになっている場合は、1通のみを届けるようにして対処することが出来る.

しかし、小さなメールを連続して何百何千通と送りつけてくるメールボムに対しては、現在のところ対応が練られていない. 手動でブラックリストに登録し受信を拒否するしかないのだが、これではこの

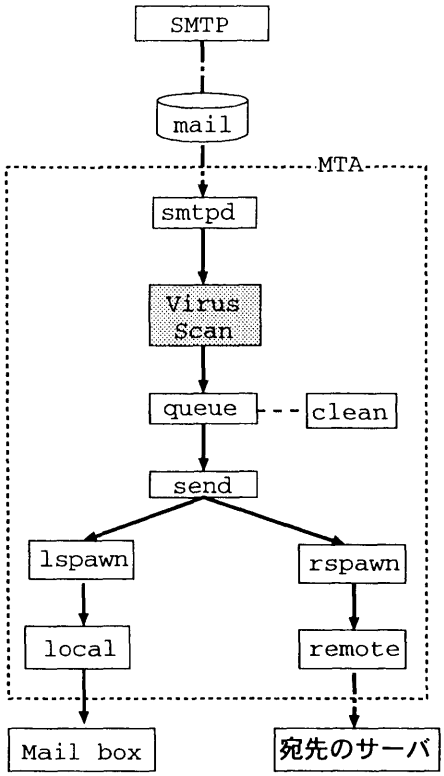

図 6: ウィルススキャンツールの追加

ようなメールボムを受けた後にしか対応することができず、そのころにはすでにサーバがリソースが消費されてしまっている. そこで、十数秒という短期間の間に、何百通というメールを送信してきたメールサーバを自動的にブラックリストに登録してしまうことも考えられる.

また、現段階では実装されていないが、メールを受信しキューに登録する間にウィルスを発見するツールをプラグイン的に組み込むことができないものかと考えている. 通常メールに添付されたファイルは何らかの方法で圧縮されていることが多いが、時折圧縮されずに添付されてくることもある. そのような場合は、添付されたファイルをスキャンすることでウィルスを発見することが可能である. また、圧縮されていてもファイルの内容を検査することは可能であり、このような場合にもウィルスを発見することが可能ではないかと考えられる. Melissa のようにメール本体に特定のコードを含んでいる場合は発見が容易である. そして、ウィルスを発見した場合は、そのユーザに対して警告のメールを送信し、ウィルスが含まれている危険性を示す. さらに、ウィルスを検出したメールの送信ホストを自動的にブラックリストに登録し、以後メールの受信を拒否するようにする. このようにして、メールを利用したウィルスの感染を防ぐことが出来るのではないかと考えられる.

謝辞

研究をすすめる上で、開発したMTAを導入し評価してくださいました河合睦子氏に深く感謝いたします。

参考文献

[1] すずきひろのぶ, 白橋明弘, 林毅著： インターネットマガジン 98/7 月号「最強インターネットディフェンス術」. インプレス, 1997〜1998, pp200〜226

[2] 石川英治, アカイコウジ 著：ハッカージャパン VOL.2. 白夜書房, 1998, pp.124-139

[3] やまだあきら 著：電子メールの配送と MTA の役割. SoftwareDesign 1998 年 10 月号. 技術評論社, 1998 年, pp.16-25

[4] Jonathan B. Postel 著：RFC821 SIMPLE MAIL TRANSFER PROTOCOL. August 1982

[5] R. Braden, Editor 著：RFC 1123 Requiements for Internet Hosts – Application and Support. Octorber 1989

[6] David H. Crocker 著： RFC822 STANDARD FOR THE FORMAT OF ARPA INTENET TEXT MESSAGES. August 1982

[7] J. De Winter 著: RFC1985 SMTP Service Extension for Remote Message Queue Starging. August 1996