オープンソースソフトウェアによる電子メール管理コンピュータの安全性対策

Computer Security Measures in Email Administration using Open Source Software

川　口　雄　一
Yuuichi KAWAGUCHI

Anti-spam and anti-viruse systems are very important measures for e-mail security. Although the author finds it practical to install anti-virus software on every personal computer, he also feels that not all users are willing to install such software. It is important to develop an anti-spam system and an anti-virus system on computers for e-mail administration. This papaer describes a method of developing such systems by using open source software. The official documents enabled the successful development of the system. As a result of testing after development, we found that systems are running without any problem.

電子メールの安全性対策として、アンチスパム、アンチウィルスはたいへん重要である。利用者個々のコンピュータにアンチウィルスが導入されることが常識となりつつあるように思われるが、同じく著者の感覚として、すべての利用者において、セキュリティ意識がアンチウィルス導入に肯定的となっているわけでもない。本稿ではこのような背景のもと、電子メール管理コンピュータ上にアンチウィルスとアンチスパムを構築することは安全性対策として有効であると考え、オープンソースソフトウェアを利用して具体的に構築する手順につい述べる。構築にあたっては、解説文書にほぼそのまま従うことで何も問題はなかった。構築試験の結果、アンチウィルスとアンチスパムはうまく動いていることがわかった。

Key words: Ubuntu（ウブンツ）
Postfix（ポストフィクス）
Amavisd-new（アマヴィス）
ClamAV（クラムエイヴィ）
SpamAssassin（スパムアサシン）

天使大学 看護栄養学部 教養教育科　（2009年１月20日受稿、2009年４月14日 審査終了受理）

Ⅰ．はじめに

１．安全対策構築の背景

周知のとおり、インターネット上を流れる電子メールの多くはスパムである。また、コンピュータウィルス(以後、「ウィルス」と略す)に感染する場合の多くは電子メールによる。したがって、電子メールの安全性対策はたいへん重要である。

グレイリストを用いると、配信されるスパムそのものの件数は激減する。例えば文献[1)]では、利用者に配信されるスパムの件数は、グレイリスト導入後は導入前に比べ1/4 = 25%に減少することが報告されている。しかし、全く無くなるわけではなく、知的なスパムなどについてはグレイリストを抜けて、利用者へ配信されてしまう。例えば、文献[1)]では、グレイリスト導入後でも利用者ひとり当り50(通/日)のスパムは配信されるとある。

ウィルスについては、利用者個々のコンピュータにアンチウィルスが導入されることが常識となりつつあるように思われる。そうであれば、定義ファイルの更新さえ間に合えば、電子メールによるウィルスに感染することはない。しかし、定義ファイルの更新が間に合わない場合もあるし、本著者の実感として、すべての利用者のセキュリティ意識がアンチウィルス導入に肯定的となっているわけでもない。

本稿ではこのような背景のもと、電子メール管理コンピュータ上にアンチウィルスとアンチスパムを構築することは安全性対策として有効であると考え、その具体的な構築手順につい述べる。本稿は、第 II 章１節で述べる「解説文書」をある程度理解できて、具体的な構築事例をもう少し詳しく知りたいと希望する読者を想定する。

２．基盤環境

安全対策の構築にあたり、電子メール環境を構築するもととなるコンピュータハードウェアの概略を示す。

CPU: AMD Turion X2, 2.20GHz
主記憶: 512MB
HDD: 20GB
OS: Linux, Ubuntu 8.10 (Server, amd64)

なお、この環境は Windows Vista (主記憶: 4GB) で動く VMware の上に構築した。公式サイト[*1]からダウンロードした iso イメージを使いインストールした。

Ⅱ．安全対策の構築

１．解説文書

Ubuntu の公式ドキュメントサイト(英語)[*2]にて「Ubuntu 8.10」を選ぶと、見出し「Other Documentation」のところに文書「Ubuntu Server Guide」があり、この中に章「13. Email Services」がある。以降では、ここに示される文書を「解説文書」と呼ぶ。

今回の環境構築にあたり、ほぼ解説文書どおりに進めることで完成できた。しかし、解説文書と実際の設定とにはいくつかの差異がある。本稿では、この差異を中心に説明する。逆に、解説文書の記載で充分な部分については説明をある程度割愛する。

２．Postfix の構築

今回の環境では MTA[*3] に Postfix を利用する。したがって、本節では「Postfix」を解説文章として構築を進める。

Postfix はパッケージとして Ubuntu 8.10 にインストールされているので、解説文書中のコマンド

```
$ sudo apt-get install postfix
```
[*4]

は不要である。しかし、念のため、アップデート版の有無を確認し、有るときにはインストールするために、

```
$ sudo apt-get update
$ sudo apt-get upgrade
```

を実行した。これは、Postfix だけではなく、インストールされているすべてのパッケージに効果がある。

なお、Ubuntu にインストールされているパッケージの一覧は、次のコマンドにより取得するこ

＊1 http://www.ubuntulinux.jp/
＊2 http://help.ubuntu.com/
＊3 Mail Transfer Agent

とができる。

```
$ dpkg --list
```

次のコマンドにより Postfix を設定する。

```
$ sudo dpkg-reconfigure postfix
```

いくつかの質問が示され、それに対して回答を与えることにより設定は終了する。回答のために事前に情報収集する必要がある。今回の事例では、いくつかについて、解説文書の例示と違う内容を回答した。以下に実際の回答と簡単な説明を、質問の順番に示す。下線部は解説文書の例示と異なる部分である。

頭に「(def.)」とある項目は、質問時にデフォルトの回答として示されたものを、そのまま選択したことを示す。

質問1: Internet Site

電子メールの送受信方法を指定する。メールハブやスマートホストなどで集中管理しているときは、それに応じて指定する。

質問2: **ubuntu.local**

電子メールを送信するときにアドレスのドメイン部[*4]を省略した場合に補完されるドメインを指定する。今回は、特に DNS を運用していないので、mDNS(Avahi) における登録名を指定した。

質問3: (def.) (空欄)

Postmaster と root に宛てられた電子メールの転送先アカウントを指定する。今回は procmail を利用する(cf. 質問7) ので、**/etc/aliases** を手で編集する。

質問4: **tenshi.ac.jp**, **ubuntu.local**, **localhost**

このサーバに宛てられたものとして受信するアドレスのドメイン部を指定する。今回は、既定値から localhost を残して他を削除し、本著者の電子メールアドレスのドメインと質問2で指定したドメインとを加えた。

質問5: (def.) No

今回は、とりあえず、既定値に従った。

質問6: (空欄)

リレー(relaying) を許可する IP アドレスを指定する。今回は空欄にした。空欄は、実際に接続されているインターフェイスから自動的にIPアドレスのリストを作ることを意味する。

質問7: (def.) **Yes**

ローカルの配送に procmail を使うかどうか。質問3でも書いたように、今回は利用する(Yes)。なお、この質問に関する記述は解説文書中になかった。

質問8: (def.) 0
質問9: (def.) +
質問10: (def.) all

今回は、質問8〜10 について、とりあえず、既定値に従った。

これらの回答は **/etc/postfix/main.cf** に反映される。

質問3と8の説明で述べたように、今回はローカル配送に procmail を利用する。Postmaster と root に宛てられた電子メールを転送するアカウントをファイル **/etc/aliases** に記載する。今回は、著者のアカウント(**yuuichi**) を指定した。

```
root: yuuichi
postmaster: root
clamav: root
```

今回この中には、後で構築するアンチウィルスのアカウント(clamav) へ送られる電子メールの転送先(root) もあらかじめ加えた。

ファイルの編集後、次のコマンドにより記載した内容を有効化する。

```
$ sudo newaliases
```

解説文書には SMTP 認証(Authentication)についての記載がある。しかし、今回はこれを利用

*4 「@」の右側部分

しないので設定しない。

最後に、Postfix を再起動[*5] する。

3．Mail Filter の構築

次に、本節では、アンチウィルスとアンチスパムを構築し、Postfix と連携させる設定を説明する。「Mail Filtering」を解説文書として構築を進める。

解説文書に倣い、アンチウィルスには Clam AV を、アンチスパムには SpamAssassin を利用し、これらと Postfix を連携させるために Amavisd-new を利用する。

■パッケージのインストール

解説文書に従い、必要なパッケージをコマンド **$ sudo apt-get install** によりインストールした。これらはすべて Ubuntu 8.10のデフォルトではインストールされていない。パッケージ名を示す。

§主となるもの

```
amavisd-new
spamassassin
clamav-daemon
```

§解説文書に倣い、とりあえずイントールするもの

```
dkim-filter
python-policyd-spf
```

これらのパッケージについてインストールはするが利用しない。

§ Spamassassin が内部で利用するもの

```
pyzor
razor
```

§電子メールの添付ファイルを処理するために使うもの

```
arj
cabextract
cpio
lha
nomarch
pax
rar
unrar
unzip
unzoo (*)
zip
zoo
```

パッケージ unzoo (*) をインストールするときに「パッケージ unzoo は無い(Couldn't find package unzoo)」というエラーになる。パッケージ zoo は unzoo の機能を持つので、これで代用する。

■設定の方針

設定の基本方針を次のとおりとした。

§送信者への通知　ウィルス、スパムともに、送信者へは DSN[*6] などで通知しない。

§ウィルス　単純に削除する。

例えば添付ファイルがウィルスに感染したときには、(1) MIME の Multipart のうち添付ファイルの部分だけを削除し、(2) 削除した記録に置き換え、(3) 感染していない本文などはそのままで受信者へ配信する方法を検討した。しかし、未だ実現できていない。

次善の方法として、ウィルスに感染した電子メールが配信されたことだけを連絡することも検討したが、これも実現できていない。

§スパム　ヘッダ(Subject) に印を付けて受信者へ配信する。

SpamAssassin の誤判定を無くすことはできない。したがって、スパムと判断された電子メールを単純に削除すると、必要なメールを失なう可能性がある。このため、すべての電子メールを受信者へ配信する。そして、配信されたメールを、

*5 **$ sudo /etc/init.d/postfix restart**

*6 Delivery Status Notification

MUA[*7] 側でヘッダ情報を利用した自動振り分けなどで処理する。

■各パッケージの設定

§ ClamAV

次の設定だけで Amavisd-new を経由して Postfix と連携できる。

```
$ sudo adduser clamav amavis
```

なお、本学の場合はプロキシ経由でインターネットに接続する。このため、ClamAV の定義ファイルを更新する freshclam について、**/etc/clamav/freshclam.conf** に次の内容を追加する。

```
HTTPProxyServer ホスト名
HTTPProxyPort ポート番号
```

§ Spamassassin

解説文書に倣い、**/etc/default/spamassassin** に次の内容を記述する。実際には、**0** となっている部分を **1** と変更するだけである。

```
ENABLED=1
```

この設定を有効にするために、spamassassin を再起動[*8] する。

§ Amavisd-new　実際の作業を順番に記す。

1．**/etc/amavis/conf.d/** 寡

15-content_filter_mode の編集 Amavisd-new からアンチウィルスとアンチスパムを利用するよう指定するため、設定ファイル中でコメントとなって除外されている部分について、すべて行頭の「#」を削除して有効とする。

2．**/etc/amavis/conf.d/50-user** の編集

解説文書では **$final_spam_destiny= D_DISCARD;** とだけ記されている。しかし、これだけでは、スパムと判定されたメールは配信されず、「スパムにはすべてヘッダに印を付けて MUA へ配信する」という方針に合わない。このため、試行錯誤の結果として設定ファイル **50-user** を作成した。この内容を本稿末尾に、内容の説明を本節末「**50-user** の説明」にそれぞれ示す。

3．Amavisd-new の再起動[*9]

4．Amavisd-new と Postfix の連携

解説文書のとおりに、次に示す2つのファイルを編集した。詳細については説明を省略する。

（1）**/etc/postfix/main.cf**

（2）**/etc/postfix/master.cf**

第Ⅱ章3節「パッケージのインストール」で説明したとおり、今回は、DKIM を利用しないので、このための設定もしない。

5．Postfix の再起動

■**50-user** の説明

本節「設定の方針」で示した内容が実現できることを目標に設定ファイル **50-user** を作成した。変数名やファイル中のコメントなどはファイル **/usr/shar/doc/amavisd-new/examples/amavisd.conf-sample.gz** を参考にした。各変数の意味は **/usr/share/doc/amavisd-new/amavisd-new-docs.html** を参考にした。

§ 1～8, 38,39. 行

もともとのファイルの内容である。

§ 9～14. 行

変数 **$final_*_destiny** は、電子メールが判定された後の配信方法を示す。今回は、次に示す値を変数に割り当てた。

D_PASS 内容に依らず受信者へ配信される。
D_DISCARD 受信者へ配信されない。送信者へは通知されない。

この他に、**D_BOUNCE, D_REJECT** を使える。しかし、どちらも送信者に通知が送られるので、今回は使わない。

§15～20. 行

変数***_quarantine_method** は、隔離(quarantine) の方法を指示する。今回は、方針に従い、

＊7 Mail User Agent
＊8 **$ sudo /etc/init.d/spamassassin restart**
＊9 **$ sudo /etc/init.d/amavisd restart**

どんなときでも隔離しないので、すべての変数に **undef** を割り当てた。

§21〜26. 行

変数 **$warn*sender** は、送信者への通知方法を指定する。方針に従い、どんなときでも送信者へは通知しないので、すべての変数に **undef** を割り当てた。

§27. 行

ウィルスに感染した電子メールを処理(= 今回は、削除) したことを受信者に通知するかどうか。値 **1** は「Yes」の意味である。今回は、期待したとおりの効果はなく、通知は送られない。

§28, 29. 行

Spamassassin が電子メールのヘッダ(Subject) に附加する目印を指定する。

§30〜37. 行

Spamassassin が電子メールを判定するときの基準値を指定する。 4 節 「ヘッダ **X-Spam-Status**」で説明する。

4．構築後の試験

解説文書にあるように、コマンド **$ telnet localhost 10024** を実行すると、応答

```
220 [127.0.0.1] ESMTP 寡 *10
amavisd-new service ready
```

が返ってきたので、Amavisd-new は活きていることが確認できた。

そこで次に、**$ telnet localhost smtp** により、Postfix を経由したときに、電子メールヘッダに必要な情報が附加されるか確認した。試行錯誤してスパム得点 S の値を見ながら Postfix に入力する内容を検討した。

■ヘッダ **X-Virus-Scanned**

常に附加される。

ウィルス感染した電子メールの扱いについては、eicar*11 から擬似ウィルスを取り寄せてシステムに入力した。その結果、Postfix のログ(**/var/log/mail.log**) には「Blocked INFECTED」と記録され、受信者へは通知も含めて配信されなかった。

■ヘッダ **X-Spam-Status**

スパム得点を S とすると、この値によって次のように変化することがわかった。

§スパムではないとき

(S < **$sa_tag_level_deflt**)

X-Spam-Status ヘッダは附加されない。

§スパムかも知れないとき

(**$sa_tag_level_deflt** ≦ S < **$sa_tag2_level_deflt**)

X-Spam-Status: No ヘッダは附加され、**Subject** ヘッダには、変数 **$sa_spam_subject_tag1** で指定した目印も附加される。

§スパムと判定されるとき

(**$sa_tag2_level_deflt** ≦ S)

X-Spam-Status: Yes ヘッダは附加され、**Subject** ヘッダには、変数 **$sa_spam_subject_tag** で指定した目印も附加される。

§実際のスパム判定

ある期間に本著者へ配信されたスパム1,816(通) をこのシステムに入力した結果を示す(単位: 通)。

スパムではない:	622
スパムかも知れない:	900
スパムである:	294

本来であれば、すべてスパムと判定されなくてはならないが、未だ SpamAssassin の調整や学習を進めていないので、判定漏れの割合は 622 ÷ 1816 = 34% となった。この件は本稿の範囲外である。

*10 本稿中で「寡」は、改行せず次の行まで続けて入力することを示す。

*11 European Institute for Computer Anti-Virus Research, **http://www.eicar.org/**

Ⅲ. 考　察

配信された電子メールに附加されるヘッダ情報から、Amavisd-new は ClamAV や Spamassassin を利用できており、Postfix とも連携できていることがわかる。

しかし、スパム得点が低くスパムでないと判定されるときに、**X-Spam-Status** ヘッダは附加されないことがわかった。配信された電子メールを MUA で振り分けるときに問題となる。また、ウィルス感染したメールは単純に削除され、受信者へは通知されない。変数 **$warnvirusrecip = 1** では期待した効果を得られていない。このように、**50-user** については、更に研究が必要である。

Ⅳ. まとめ

オープンソースソフトウェアを利用して、電子メールの安全性対策として、サーバコンピュータ上にアンチウィルスとアンチスパムを構築した。基盤環境として Ubuntu 8.10 Server を利用し、構築にあたっては、解説文書にほぼそのまま従うことで何も問題はなかった。構築試験の結果、アンチウィルスとアンチスパムはうまく動いており、また、MTA との連携もできていることがわかった。

今後は、今回構築した安全対策を現在のネットワークにはさみこみ、電子メールを MUA に配信するネットワークを構築することを考えている。また MUA におけるスパムの自動振り分けやデータベースの学習[2)]も検討課題となる。

謝　辞

オープンソースソフトウェである Linux (Ubuntu), Postfix, Amavisd-new, ClamAV, Spam Assassin の開発者に感謝します。これらはたいへん有益で効果の高いソフトウェアです。

参考文献

1）川口雄一：グレイリストを利用した迷惑メール対策, 平成20 年度情報処理教育研究集会論文集, G1-3, 九州工業大学, 2008年.

2）Alistair McDonald, SpamAssassin: A Practical Guide to Configuration, Customization, and Integration, Packt Publishing, 2004.

設定ファイル 50-user*12

```
1. use strict;

2. #
3. # Place your configuration directives here. They will override those in
4. # earlier files.
5. #
6. # See /usr/share/doc/amavisd-new/ for documentation and examples of
7. # the directives you can use in this file
8. #
9. ## MAIL FORWARDING AND DKIM SIGNING
10. #
11. $final_virus_destiny = D_DISCARD;
12. $final_banned_destiny = D_PASS;
13. $final_spam_destiny = D_DISCARD;
14. $final_bad_header_destiny = D_PASS;
15. ## QUARANTINE
16. #
17. $virus_quarantine_method = undef;
18. $spam_quarantine_method = undef;
19. $banned_files_quarantine_method = undef;
20. $bad_header_quarantine_method = undef;
21. ## NOTIFICATIONS (DSN, admin, recip)
22. #
23. $warnvirussender = undef;
24. $warnbannedsender = undef;
25. $warnspamsender = undef;
26. $warnbadhsender = undef;
27. $warnvirusrecip = 1;
28. $sa_spam_subject_tag1 = '[AMaViS:##] ';
29. $sa_spam_subject_tag = '[AMaViS:#####] ';
30. ## ANTI-SPAM CONTROLS
31. #
32. $sa_tag_level_deflt = 3.0;
33. $sa_tag2_level_deflt = 6.31;
34. $sa_kill_level_deflt = 9999;
35. $sa_dsn_cutoff_level = 9999;
36. # $sa_crediblefrom_dsn_cutoff_level = undef;
37. # $sa_quarantine_cutoff_level = undef;
38. #------------ Do not modify anything below this line -------------
39. 1; # ensure a defined return
```

*12 各行頭にある番号(1.〜39.) は説明のために付けたものである。