Vol.2010-IOT-10 No.7
2010/7/16

マルチドメインによる Mailman メーリングリストのセキュリティ対策

王　躍[†]　久長　穣[†]　小河原　加久治[††]

メーリングリストは，電子メールを使って，特定の複数ユーザの間で情報交換するシステムである．１つのグループに登録しているユーザに電子メールを同時配信することにより実現している．しかし，メールを多数に同報するシステム構造に起因するセキュリティ上の問題点（例えば，ウイルスメールやスパムメールの問題）も指摘されている．そこで，セキュリティ対策として，メーリングリストを外部向けのものと内部向けのものを分けて管理することによって，メーリングリストへのウイルスメールやスパムメールを減らすことが考えられる．本稿では，現在最も広く利用されているメーリングリスト管理システムの１つである Mailman について，Mailman のマルチドメイン機能を利用して外部ドメインと内部ドメインに分けた Mailman メーリングリスト管理システムを構築する方法を提案して説明する．

Security Measures for Mailman Mailing List Management System, with Using Multiple Virtual Domains

Yue Wang[†], Yutaka Hisanaga[†] and Kakuji Ogawara[††]

An electronic mailing list allows for widespread distribution of information to many internet users. However, there are some known security concerns with using of electronic mailing lists (for example, the e-mail spam problem, the e-mail worms problem, and etc.). To reduce the amount of spam and virus emails that reach end users subscribed to a mailing list, it is effective to divide the domains of electronic mailing lists into internal and external ones, as a server-side security measure. In this paper, we will give an explanation of how to construct a Mailman mailing list manage system, which is the most popular free software mailing list manager, with using the internal and external virtual domains.

1. はじめに

Mailman は，ウエブと統合されたメーリングリスト管理システムのフリーソフトとして，電子メールやメルマガの配送に現在広く利用されている[1, 2]．メーリングリストは，情報交換の手段の一つとして古くから存在する仕組みであるが，その一方で，メーリングリストの活用に伴うセキュリティ上の問題点も存在する[3]．例えば，ウイルスメールやスパムメールがメーリングリストに送られた場合，メーリングリスト参加者全員にそのメールが届いてしまうことになるので，メーリングリストの規模が大きいほどその影響が大きくなる．また，ウイルスメールやスパムメールの配信及びそれらのメールに関連するエラーメール処理によるメールサーバ負荷の増大になるケースもよくある．そこで，本稿ではメーリングリスト管理システム側での対策として，メーリングリストを外部向けのものと内部向けのものを分けて管理することによって，ウイルスメールやスパムメールを減らすことが考えられる．

Mailman のバージョン 2.1 からマルチドメインをサポートする[4]ようになったので，本稿では，Mailman のマルチドメイン機能を利用して，外部ドメインと内部ドメインに分けた Mailman メーリングリスト管理システムを構築する方法を提案して説明する．なお，本稿では，Postfix[5]メールサーバを使って Mailman メーリングリスト管理システムの構築を行うこととする．また，全ての動作確認は FreeBSD7.2 において行われる．

2. 外部／内部向け DNS サーバの設定

本稿では，メーリングリスト用の外部 ML ドメインと内部 ML ドメインをそれぞれ，「mlex.my.domain」と「mlin.my.domain」とする．

外部 ML ドメインは，外部向け DNS サーバと内部向け DNS サーバ両方に MX レコードとして登録されるが，内部 ML ドメインは，外部からの投稿を一切受け付けしないため，内部向け DNS サーバだけに MX レコードとして登録する．

なお，MX レコードの登録状況は，次のコマンドで確認することができる．

```
# nslookup
> server [DNS サーバの IP アドレス]
> set q=mx
> [FQDN]
> exit
```

[†]山口大学　大学情報機構　メディア基盤センター
Media and Information Technology Center, Organization for Academic Information, Yamaguchi University
[††]山口大学　理工学研究科
Graduate School of Science and Engineering, Yamaguchi University

ただし，[FQDN]は，ML 外部（内部）ドメインの名前である．

3. Postfix と Mailman の統合

本稿では，Postfix と Mailman のインターフェースとして，Dax Kel son 氏が作成した Python スクリプト postfix-to-mailman.py[6]を用いて，Postfix と Mailman の統合を行う．この方法は，Postfix の transport 機能を使って，バーチャルドメインを通して Mailman にメールを渡して処理する．その利点は，メーリングリストを新規に登録する度に Postfix のアリエスを用意せずに済むことと，スクリプトによるニーズに応じた動的な対応ができることにある．

Mailman は，バージョン 2.1 からマルチドメインを対応しているが，異なるドメインが同じリスト名をもつことができないという制限がある[4]．そのため，外部向けと内部向けのメーリングリストは，Mailman では区別されない．そこで，本稿では，「mlex.my.domain」と「mlin.my.domain」両ドメインのリストへの対応は，それぞれ，Python スクリプト postfix-to-mlex.py と postfix-to-mlin.py で行う．ただし，スクリプト postfix-to-mlex.py と postfix-to-mlin.py は，それぞれ，ドメイン「mlex.my.domain」と「mlin.my.domain」のみのものを受理するように修正した postfix-to-mailman.py のコピーである．例えば，postfix-to-mlex.py は，postfix-to-mailman.py を次のように修正した．

```
if os.path.exists(local):
        from Mailman.MailList import MailList
        m = MailList(mlist, lock=0)
        host_name = m.host_name
        if host_name == 'mlex.my.domain':   # ML ドメインのチェック
            os.execv(
                        MailmanHome + "/mail/mailman",
                        (MailmanHome + "/mail/mailman", type, local)
            )
 els        e:
            bounce()
```

次に，Postfix と Mailman 統合の実現方法については，以下の手順に沿って具体的に説明する．

(1) postfix-to-mlex.py と postfix-to-mlin.py を/usr/local/mailman/bin/にインストールする．

(2) Mailman の設定ファイル mm_cfg.py に次の行を（編集して）追加する．

```
MTA = None   #   No   MTA   processing   required   for   postfix-to-mailman.py
```

(3)「mlex.my.domain」と「mlin.my.domain」を Postfix の transport マップファイル /usr/local/etc/postfix/transport に次のように登録する．

```
mlex.my.domain m       lex:
mlin.my.domain m       lin:
```

なお，上述の設定を反映させるために，コマンドpostmap transport を実行する．

(4) Postfix の設定ファイル main.cf に次の行を追加，もしくは，編集する．

```
transport_maps = hash:/usr/local/etc/postfix/transport
relay_domains = ..., mlex.my.domain, mlin.my.domain
mlex_destination_recipient_limit = 1
mlin_destination_recipient_limit = 1
```

(5) Postfix の設定ファイル master.cf に次の行を追加する．

```
mlex unix - n n - - pipe
 flags        =FR user=mailman:mailman
 ar         gv=/usr/local/mailman/bin/postfix-to-mlex.py ${nexthop} ${user}
mlin unix - n n - - pipe
 flags        =FR user=mailman:mailman
        argv=/usr/local/mailman/bin/postfix-to-mlin.py ${nexthop} ${user}
```

4. Postfix の SMTP フィルタの利用

Postfix のバージョン 2.1 から，Postfix SMTP サーバは入ってくるメール全てを Postfix メールキューに入れられる「前に」全てのメールを検査するコンテンツフィルタリングプロキシサーバに転送できるようになった[7]．この機能を利用すれば，Postfix が実際に Mailman にメールを配送する前に，送信者（MAIL FROM）または受信者（RCPT TO）アドレスが配送可能かどうかを検証することができる．配送不可能と判断されたメールについては，Postfix に適切な SMTP ステータスコードを送り返すことでメールを拒否する．そして，Postfix はリモートの SMTP クライアントにそのステータスを返す．このようなアドレスの検証をすれば，Postfix（Mailman）はバウンスメッセージを送る必要はない．多くの場合は，リモートの SMTP クライアントがバウンスメッセージより SMTP ステータスコードを望ましい．なぜならば，「迷惑メール」に関するバウンスメッセージは，多くの迷惑メールの送受信者アドレスがでたらめであるため，

ほとんど新たな「迷惑メール」になってしまうからである．なお，その場合のバウンスメッセージの責任は，リモートの SMTP クライアントに委ねることになる．

本稿では，Bennett Todd 氏が開発された，軽量・単純な SMTP プロキシ smtpprox [8] を用いて，外部 ML ドメインと内部 ML ドメインのリストへの投稿メールに対するアドレス検証を実現する方法について説明する[9].

(1) Postfix SMTP プロキシ機能の設定

メール処理の流れが次のようになる．

【リモート SMTP クライアント】⇒
【Postfix SMTP サーバ on port 25】⇒
【smtpprox フィルタ on localhost port 10025】⇒
【Postfix SMTP サーバ on localhost port 10026】⇒
【Postfix cleanup サーバ】⇒・・・

フィルタ前の Postfix S MTP サーバは，リモート SMTP クライアントからのメールを，localhost ポート 10025 でメールを待つ smtpprox フィルタに渡す．そして，フィルタ後の Postfix SMTP サーバは, localhost ポート 10026 を通して smtpprox フィルタからメールを受け取る．それ以降 Postfix は通常通りメールを処理する．

Postfix の設定ファイル master.cf での smtp に関する設定を次のように修正する．

```
smtp inet n - n - 20 smtpd
 -o        smtpd_proxy_filter=127.0.0.1:10025
 -o        smtpd_client_connection_count_limit=10
        #Postfix 2.7 and later performance feature.
 #-o        smtpd_proxy_options=speed_adjust

 127.0.0.1:10026 inet n - n - - smtpd
 -o        smtpd_authorized_xforward_hosts=127.0.0.0/8
 -o        smtpd_client_restrictions=
 -o        smtpd_helo_restrictions=
 -o        smtpd_sender_restrictions=
 -o        smtpd_recipient_restrictions=permit_mynetworks,reject
 -o        smtpd_data_restrictions=
 -o        mynetworks=127.0.0.0/8
 -o        receive_override_options=no_unknown_recipient_checks
```

ただし，ここで，master.cf の新しいエントリ「127.0.0.1:10026」でフィルタ後の Postfix SMTP サーバを定義している．また，この SMTP サーバをネットワークに晒さないように、localhost アドレスのみで待機させるように設定している．

(2) smtpprox フィルタのカスタマイズ

smtpprox は，言語 Perl で作成されたソフトウェアで，Postfix のコンテンツフィルタリングフック機能を利用した，SMTP で送受信するような高度なコンテンツフィルタリングプロキシを構築するためのフレームワークとして利用できる．本稿では，エンベロープ情報（MAIL F ROM，RCPT T O）に基づくフィルタリングを行う方法について説明する．同様にメールのメッセージ情報に基づくフィルタリングもできる．

smtpprox フィルタは，フィルタ前の Postfix SMTP サーバからの SMTP コマンドを受け取り，Mailman に配送可能と判断されたメールを，そのまま（つまり，受け取った SMTP コマンドのコピーを）フィルタ後の Postfix SMTP サーバに送る．そして，Mailman に配送不可能と判断されたメールに対しては，SMTP 応答コード「550」をフィルタ前の Postfix SMTP サーバに返し，フィルタ後の Postfix S MTP サーバへの SMTP 接続を強制的に切断する．

そのために，smtpprox に次のソースコードを編集する．

```
if ($what =~ /^mail[ ¥t]*from:[ ¥t]*<?[ ¥t]*([^ ¥t]+)¥@([^ ¥t>]+)[ ¥t]*>?/i) {
        my $from_domain = $2;
}

if ($what =~ /^rcpt[ ¥t]*to:[ ¥t]*<?[ ¥t]*([^ ¥t]+)¥@([^ ¥t>]+)[ ¥t]*>?/i) {
        my $listname = $1;
        my $listdomain = $2;

        my $result =
`/usr/local/mailman/bin/check_this_post $listname $listdomain $from_domain`;

        if ($result eq 'no') {
            # Send our reject of this line back and head to the next line
            $server->fail(‘550 insufficient authorization: ’ . $listname);
            # We keep processing in case we're one of multiple recipients
            next;
 }
}
```

ただし，「/usr/local/mailman/bin/check_this_post」は，（言語 Pyhton で適切に書かれた）リスト名と ML ドメイン名と投稿者のメールアドレスに基づいて配送の判断を行うスクリプトとする．

また，smtpprox デーモンは，次のコマンドで起動する．

```
smtpprox 127.0.0.1:10025 127.0.0.1:10026
```

更に，smtpprox フィルタの動作については，次のようなコマンドで確認することができる．

```
# telnet localhost 10025
HELO my.domain
MAIL FROM: <me@my.domain>
RCPT TO: <list@mlex.my.domain>
DATA
From: <me@my.domain>
To: <list@mlex.my.domain>
Subject: test

test email
.
QUIT
^]
telnet> close
```

5. むすび

本稿では，Mailman メーリングリスト管理システムに対するセキュリティ対策として，Mailman のマルチドメイン機能を利用して外部ドメインと内部ドメインに分けた Mailman メーリングリスト管理システムを構築する方法を提案して説明した．利用者は自分のニーズに応じて，外部向けのメーリングリストと内部向けのメーリングリストを分けて利用することによって，リスクを必要最小限にすることができる．また，Postfix の SMTP フィルタの利用によって，ウイルスメールやスパムメール配信及びそれに関連するエラーメール処理によるメールサーバ負荷の増大を防ぐことを期待する．

謝辞 本研究にご協力いただいた山口大学・学情報機構・メディア基盤センターの皆様に，謹んで感謝の意を表する．

参考文献

1) Ma ilman「GNU メーリングリスト管理システム」http://mm.tkikuchi.net/
2) Ma ilman「the GNU Mailing List Manager」http://www.list.org/
3) ウィキペディア「メーリングリスト」http://ja.wikipedia.org/wiki/メーリングリスト
4) Ba rry A. Warsaw，Zope Corporation：「GNU Mailman, Internationalized」，Proceedings of the USENIX 2003 Annual Technical Conference, FREENIX Track，pp.39-50(2003)
5) Po stfix「Documentation」http://www.postfix.org/documentation.html
6) Dax Kelson「Interface mailman to a postfix with a mailman transport」http://www.gurulabs.com/downloads/postfix-to-ma man-2.1.py
7) SMTP D「Postfix SMTP サーバ」http://www.postfix-jp.info/trans-2.1/jhtml/smtpd.8.html
8) s mtpprox「simple efficient SMTP proxy in perl」http://bent.latency.net/smtpprox/
9) SMTP D_PROXY「Postfix Before-Queue Content Filter」http://www.postfix.org/SMTPD_PROXY_README.html