送信者ドメイン認証を入れてみました

神保道夫
(karl ＠ ebug.jp)

送信者認証とは？

- メール受信時に、メールの差出人のサーバーに、そのメールが送信されたIPアドレスを用いたものか問い合わせを行い、正しいメールかどうかを確認する。
- 迷惑メールやフィッシングに対する対策として期待されているものが「送信者ドメイン認証」である。

送信者認証の歴史

- 2004年頃から本格的に研究が始まる
  1. 米Microsoftの「Caller ID for E- Mail」
  2. 米Yahoo!の「DomainKeys」
  3. 米PoboxのMeng Wong氏が開発し，米AOLが担いでいた「SPF（Sender Policy Framework）」

の3種類が注目される。

1. 3.の特徴(1)

- SPFでは，MAIL FROMで送られたアドレス（ドメイン）のDNSサーバーにアクセスする。そして，そのDNSサーバーのSPFレコードに，現在通信している相手のIPアドレスが記されていれば，相手はMAIL FROMのアドレスを詐称していないことになる。同様に，Caller ID for E-MailではFromヘッダーのアドレス（ドメイン）が詐称されていないかどうかをチェックする。

1. 3.の特徴(2)

- 要約すると、DNSサーバーに格納した情報をもとに、メールの送信者が正しいことをチェックする。
  →「Sender IDとして統合される」

2. の特徴(1)

- 送信者ドメインのDNSサーバーに、そのドメインの公開鍵を格納しておく。メールの送信者（送信サーバー）は、送信時にメールの中身のハッシュ（メッセージ・ダイジェスト）を計算し、そのハッシュを前述の公開鍵に対応した秘密鍵で暗号化しておく。

2. の特徴(2)

- 暗号化データはメールのヘッダー情報の一つとして、メールに含めておく。メールの受信者は、メールのFromヘッダーに書かれたアドレス(ドメイン)の DNSサーバーへアクセスして、公開鍵を取得。その鍵で復号したハッシュと、受け取ったメールから計算したハッシュが一致すれば、そのドメインから送られてきたことが検証できる。

2. の特徴(3)

- 1. 2. と比べて、改ざんを検出することも可能なので、有利？

Sender IDの現状

- ライセンスの不明瞭さから異論が出て、頓挫！
  →ワーキンググループが解散
- http://itpro.nikkeibp.co.jp/free/ITPro/OPINION/20041115/152576/?P=2&ST=security に詳しく書いてあるので、そちらを参照

ここまでのまとめ

- あくまでも送信者認証はSPAM対策ではなく、フィッシング対策であることに留意！

Sender-ID の実装をしてみる

- メール差出人のドメインに対して、“TXT”レコードを使って、使用するIPアドレスを記述する。
- 受け取り側のメールサーバーに、milterなどを使って(sendmailの場合)送信者認証に必要な処理をする。

sendmailにSender-ID・SPFを導入してみる

- sendmail 8.13以降では、sid-milterというmilterがある。これを利用すると、SPFにもSender-IDにも対応できるので、これを利用する。
- 最近のFreeBSDでは、sendmail 8.13以降でかつ、MILTERが使える状態でコンパイルされているので、特に何もする必要はない。(今回は、portsから、sendmail 8.14を入れてやってみた)

動作の概略

- 送信元ホストから送信先ホストにSMTP接続する。
- SPF の場合、SMTP ハンドシェイクの MAIL FROM を見て（Sender-ID の場合、BODY の From: を見て）該当ドメインの TXT レコードを引く。
- TXT レコードに書いてあるマシンから送信されているか確認する。
- 書いてあるマシンからであれば受信処理を継続する。そうでなければ受信拒否処理を行う。

sid-milterのインストール(1)

- まず、sid-milterは、8891/TCPを使う場合があるのでファイアウォールで拒否している場合は設定変更をする。
- /usr/ports/mail/sid-milterをインストールする。
- /etc/rc.d/sendmail stopをする
- FQDN.mcファイルに、
  INPUT_MAIL_FILTER(`sid-filter',`S=local:/var/run/sid-filter')dnl
- を追加し、/etc/mail/sendmail.cfを作り直す。
- touch /var/run/sid-filterをする

sid-milterのインストール(2)

- /etc/rc.confに、以下の行を追加する
  miltersid_enable="YES“
  miltersid_flags="-h -l -r 0“
- -r は0～4が指定できます。
  0 … 全メールを受信します（ただしヘッダの記録は残します）。
  1 … Sender-ID でも SPF でも不可の場合には拒否します。
  2 … Sender-ID か SPF どちらかが不可の場合には拒否します。
  3 … Sender-ID か SPF どちらかが合格でなかったら拒否します。
  4 … Sender-ID も SPF も両方合格でないなら拒否します。
- 今回はテストのために、0を指定します。また、メールのヘッダにsid-milterが動作していることを入れ(-h)、syslog経由で挙動を記録します(-l)

sid-milterのインストール(3)

- /etc/mail/mailer.confを書き直して、/usr/local/sbin/sendmailが動くように書き直す。
- ここまででsendmail関係の設定は終了です

DNSの設定

- 自宅の例では、jinbo.jp というメールアドレスには、sv.jinbo.jp というマシンからしかメールを送らないので、jinbo.jp zoneに対して、
  IN TXT "v=spf1 mx a:sv.jinbo.jp -all“
- sv.jinbo.jpに対しては、
  IN TXT “v=spf1 a:sv.jinbo.jp ~all“
- と書いてやる。

各種デーモンの起動

- ここまでで設定が終わったので、各種デーモンを再起動する。
  #rndc reload jinbo.jp
  #/usr/local/etc/rc.d/milter-sid start
  #/etc/rc.d/sendmail start

sid-milterに引っかかった例

- ドメインに定義されていない場合
  Authentication-Results: casper.jinbo.jp from=kingstonGaffney@18004baylor.com; sender-id=neutral; spf=neutral
- 失敗例1(おそらくYahoo!のIPアドレス以外から送信を許可している場合)
  Authentication-Results: casper.jinbo.jp from=kaori_s@yahoo.co.jp; sender-id=softfail; spf=softfail
- 失敗例2(許可しているドメイン以外から送信した場合)
  Authentication-Results: casper.jinbo.jp from=stuarta.stuart_bb@mecte.supplementgrasp.com; sender-id=fail (NotPermitted); spf=fail (NotPermitted)

sid-milterの設定を見直す

- ここまでの設定で、sid-milterが動作していることが確認できたので、/etc/rc.confのレベルを調整して、好みの動作にする。

まとめ

- SPFやSender-IDは、あくまでドメイン認証であり、その送信メールが正しい差出人から出されたものかどうかは保障できないので、運用には注意が必要である。

参考文献

- http://www.gulf.or.jp/~too/freebsd/spf.html
- http://itpro.nikkeibp.co.jp/free/ITPro/OPINION/20041115/152576/
- その他、各種SPF等の解説文書