7. Spam対策

配信制限

- 意図しないメールの受信
  - メール爆撃 (Mail Bombing)
  - 宣伝メール (Spam)
- 意図しないメールの中継
  - 組織外から組織外へ(踏み台)
  - CPU、ネットワーク、ディスクの圧迫
- 通信先制限
  - 組織内のみ
    - 学生実験、アルバイト、出向社員

判断材料

- 接続元/先のホストの
  - IPアドレス
  - ドメイン名
- 発信者のメールアドレス
  - エンベロープ
- 受信者のメールアドレス
  - エンベロープ
- 上記の組合わせ

どこで制限するか

- SMTP コネクションの接続時
- SMTP セッション中
  - HELO/EHLO ホスト名
  - MAIL FROM:<発信者アドレス>
  - RCPT TO:<受信者アドレス>
- 受信後
  - 転送先判定処理
- 負荷を抑えるためにもSMTPの時点でなんとか制限したい

従来のsendmailの場合

- TCP Wrapper (tcpd)
  - SMTP コネクションの接続時
- checkcompat( )
  - 受信後
  - バイナリへの組み込み

sendmail 8.8 では

- check_relay
  - 接続時のIPアドレス/ホスト名チェック
- check_mail
  - MAIL FROM: のアドレスチェック
- check_rcpt
  - RCPT TO: のアドレスチェック
- check_compat
  - 受信後、配信前のアドレスチェック
  - 従来の checkcompat( ) のルールセット版

*check_* ルールセット

- 配信用のルールセットとは独立
- 従来の sendmail.cf に単純に追加するだけ
- error mailer に渡すことで配信を拒否

check_relay

- 入力:

  ホスト名 $| *IPアドレス*

```
# データベース･マップの定義
Kspammers hash /etc/spammers (NEWDB が必要)
# ルールセットの定義
Scheck_relay
R $+ $| $+ $: $(spammers $1 $: OK $)
R OK                    $@ OK
R $+                    $#error $: 521 $1
```

トークン

- メールアドレスを分割文字で区切った最小単位

  "motonori" "@" "wide" "." "ad" "." "jp"

- アドレスのパターンマッチングの単位

- 分割文字

  .:%@!^/[]+ (OperatorChars)

パターン・マッチと書き換え

- R行は左辺でパターンマッチングをおこない、右辺に従って書き換える

  sendmail / 対応するregexp

$-	.	1つのトークンにマッチ
$*	.*	0以上のトークンにマッチ
$+	.+	1以上のトークンにマッチ

  - 右辺において $1, $2,... で参照できる

$@	^$	何もなしにマッチ

その他のオペレータ

$|　　区切り

$@　　ルールセットの処理の終了

- 右辺の先頭

$:　　ルールセットの繰返し処理の抑制

- 基本は while loop

$($)　　データベース･マップの参照

$#error　拒否($: に続けてメッセージを伴う)

$>　　ルールセットの呼びだし

マップ・ファイルの作り方

makemap hash /etc/spammers.db < /etc/spammers

spamhost1.domain　　any comment
spamhost2.domain　　any comment

- NEWDB が必要 (hash, btree)
- NDBM を利用する場合は makedbm などで
  - OS 添付のコマンドを利用

check_relay の問題点

- ブラックリストのメンテナンスが面倒
- 定義したホストと一切通信できなくなる
- よその踏み台が利用されると効果がない

check_mail

- エンベロープの発信者アドレスのチェック

```
Kspammers hash /etc/spammers
Scheck_mail
# check for domain name
R <>                    $@ OK          null address
R $*                    $: $>3 $1
R $*<@$+>$*             $: $(spammers $2 $: OK $)
R OK                    $@ OK
R $+                    $#error $: 551 $1
```

アドレスのフォーカス

- ドメイン部にマークをつける
  - ルールセット3 を呼び出す

motonori <@wide.ad.jp>
<@route>: motonori@wide.ad.jp

  - 転送ホスト指定などがある場合
  - 最初に送るべきホストにマーク

check_mail の問題点

- ブラックリストのメンテナンスが面倒
- 定義したドメインと一切通信できなくなる
- アドレスが偽装されると効果がない

check_rcpt (cont.)

- エンベロープの受信者アドレスについて適用される。

FR-o /etc/sendmail.cR (ファイル･クラス/オプション)

Scheck_rcpt

anything terminating locally is ok

R <$+ @ $=w > $@ OK

R <$+ @ $=R > $@ OK

check_rcpt (cont'd)

```
# anything originating locally is ok
R $*                    $: $(dequote "" $&{client_name} $)
R $=w                   $@ OK
R $=R                   $@ OK
R $@                    $@ OK
# anything else is bogus
R $*                    $#error $: "550 Relaying Denied"
```

ファイル・クラスとクラス・マッチ

- クラス定義 - 文字列の集合
  - FR ファイルによるクラスR定義
  - CR sendmail.cf直書きにのクラスR定義

    CR good.domain
- $=R
  - クラスRの要素とマッチ(左辺で利用)
- ファイルクラスの変更
  - sendmail の再起動が必要

$マクロと&$マクロ (cont.)

- 固定的な値を持つマクロ
  - $ のみで参照
    - $j ホスト名
    - $m ドメイン名

$マクロと&$マクロ (cont'd)

- 実行ごとに値が変わるマクロ
  - $& で参照
    - $&{client_addr} - 接続ホストのIPアドレス
    - $&{client_name} - 接続ホストのホスト名
    - $&f - エンベロープの発信者(解析前)
    - $&g - エンベロープの発信者(解析後)

{long_name} は sendmail 8.7 から

$&マクロとdequoteマップ

- $&マクロの展開ではトークン分割されない
- マッチングにはトークン分割が必要
  - トークン分割のためにdequoteを利用
- dequote map (本来の用途)
  - "" で囲まれた部分は1トークン
  - "" を外し、トークンに分割
- $(dequote "" $&{client_name} $) で解決

check_compat

- 全てのエンベロープの発信者と受信者のアドレスごとに呼び出される
- 入力:

  <*sender*> $| <*recipient*>
- 受信を完了し配信前に適用
  - エラーの返信処理に負荷がかかる
  - SMTPの際に拒否すれば送信側でエラー処理

check_rcpt での同等の判定

- $f (エンベロープの発信者) を追加

```
R $*        $: $1 $| $>3 $( deqote "" $&f $)
```

check_* を -bt でテスト

- $| 変換用ルールセットを作成

```
Sconv
R $* $$ | $*      $1 $| $2
```

```
% sendmail -bt
> conv,check_compat <sender> $| <recipient>
> .Df<sender>
> conv,check_rcpt <recipient>
```

転送制限とメーリングリスト

- 組織外から組織外への転送に見える
  - 組織内のホストからの接続は許可
- 組織内の2つのホストを通過させると…
  - 経路指定のついたメールアドレスの拒否

CFを使う

MAIL_RELAY_RESTRICTION=yes

LOCAL_HOST_IPADDR='130.54 133.3 192.50.8'

LOCAL_HOST_DOMAIN=kyoto-u.ac.jp

ALLOW_RECIPIENT_DOMAIN=kyoto-u.ac.jp

ALLOW_RELAY_JUST_FROM=kyoto-u.ac.jp

ALLOW_RELAY_JUST_TO=kyoto-u.ac.jp

参考

- http://www.sendmail.org/antispam.html
- http://www.informatik.uni-kiel.de/~ca/email/check.html