milter managerで簡単迷惑メール対策

須藤功平
株式会社クリアコード
2010/09/18

話すこと

milterの動作

- ✓ 複数milter利用時の動作

milter manager

- ✓ 少しだけ

自己紹介

- ✓ クリアコード代表取締役
- ✓ プログラマ

クリアコード

> オープンソースソフトウェア開発者の技術力をビジネス分野において積極的に活用していくことを目的に…設立いたしました。

関連ソフトウェア

- milter manager
  - 迷惑メール対策
- Cutter, UxU
  - テスティングフレームワーク
- groonga
  - 全文検索エンジン

groonga

(画像: CC BY-NA Brazil)

- ✓ Sennaの後継
  - ✓ MySQL対応
  - ✓ より高速・柔軟に
- ✓ さらに発展
  - ✓ 位置情報対応
  - ✓ ネットワーク対応

milter manager

milter manager

> milterを組み合わせて効果的な迷惑メール対策を実現する迷惑メール対策ソフトウェア。

milterを組み合わせる？

- milter
  - メールフィルター
- 迷惑メールの多様化
  - 対策も多様化
  - milterの組み合わせ

milter組み合わせ時の動作

- 挙動が少し複雑
  - 1回聞いただけだとピンとこない
  - 確認しながら聞けばわかる
- milter managerとの関係
  - オススメ設定で運用: 知らずにOK
  - カスタマイズする: 必須の知識

milterとは

- 発祥: Sendmail
- メールフィルターの…
  - 仕組み
  - プロトコル
  - API
  - 実装

milterの歴史

- Sendmail 8.12.0で正式リリース
  - 2001年9月
- Postfix 2.3.0からサポート開始
  - 2006年6月
  - 当時は部分的サポート
  - 2.7.1ではほぼフル機能サポート

最近は
安心

milterの挙動

https://www.milter.org/developers/overview#ControlFlow

```
For each of N connections {
  For each filter
    process connection (xxfi_connect)
  For each filter
    process helo (xxfi_helo)
  MESSAGE:For each message in this connection (sequentially) {
    For each filter
      process sender (xxfi_envfrom)
    For each recipient {
      For each filter
        process recipient (xxfi_envrcpt)
    }
    For each filter {
      process DATA (xxfi_data)
      For each header
        process header (xxfi_header)
      process end of headers (xxfi_eoh)
      For each body block
        process this body block (xxfi_body)
      process end of message (xxfi_eom)
    }
  }
  For each filter
    process end of connection (xxfi_close)
}
```

milterの挙動: 3行で

複数のmilterを利用している場合

- ✓ 本文は直列
  （前のmilterの影響を受ける）
- ✓ 本文以外は並列
  （前のmilterの影響を受けない）
- ✓ 詳細な結果は最後に返す

Q&A

SPFとGreylist: Q

> Q: ENMAでSPFを検証して、milter-greylistでその結果を見てGreylistするかどうかを判断できる？

SPFとGreylist: A

A: △ できるけど、たぶん、想像している動作と違う。

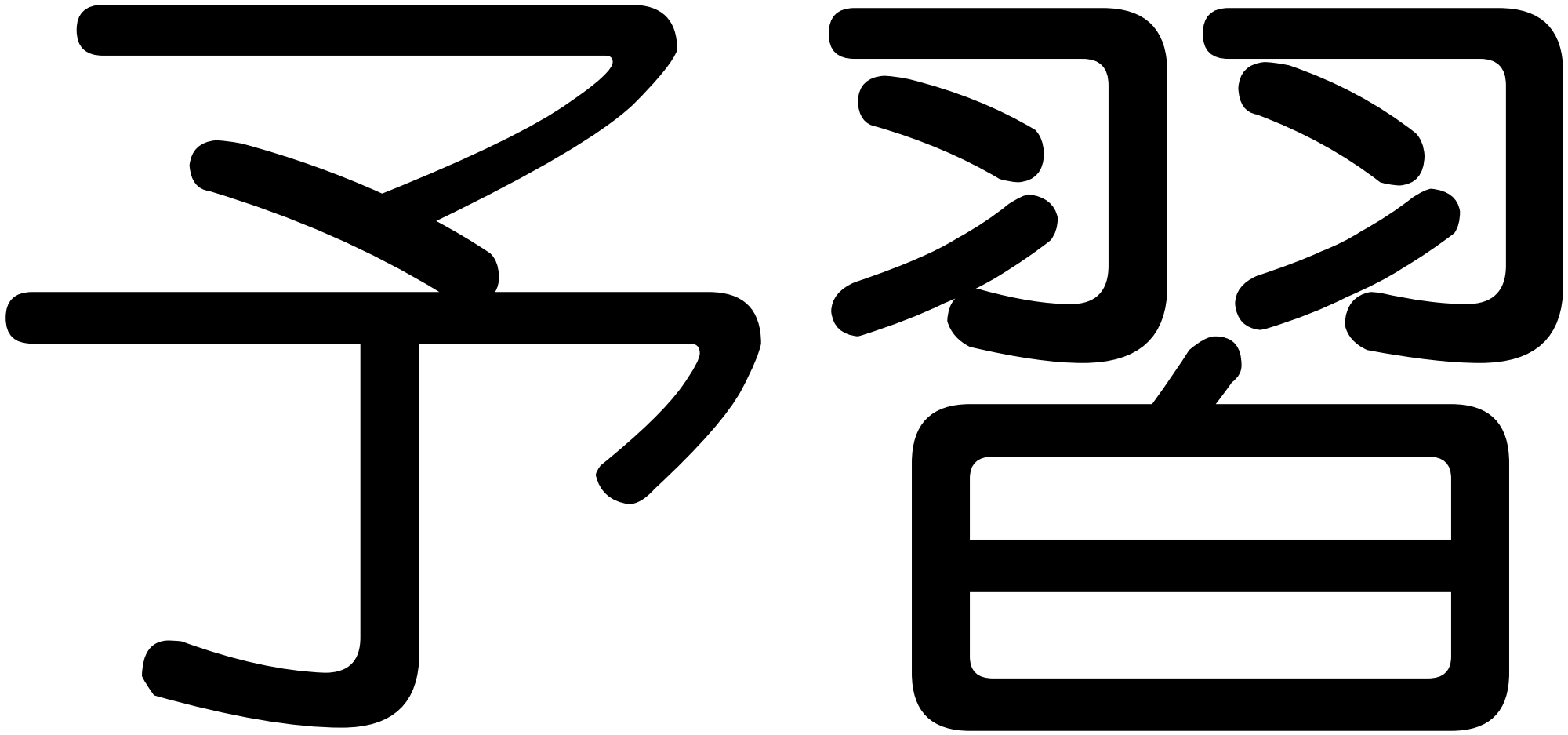
ENMAはメッセージ全体を読み込んだ後に結果を報告するので、メッセージ全体を読み込んだ後でないとGreylistできない。

ケーススタディ

- ✓ ケースを紹介
- ✓ 結果は？
- ✓ 答え合わせ
- ✓ 解説

結果は？

1. 処理続行
2. 受信
3. 拒否
4. その他



処理の流れ

```
For each of N connections {
  ...
  For each filter
    # connectステージ
    process connection (xxfi_connect)
  For each filter
    # heloステージ
    process helo (xxfi_helo)
  ...
}
```

milterプロトコル

- ステージ
  - SMTPのコマンド + α
- アクション
  - MTAへのレスポンス
- メッセージ変更
  - フィルタ機能（処理の一番最後）

ステージ

SMTP + α

connect	header
helo	end of header
mail from	body
rcpt to	end of message
data	

アクション

continue 続行	tempfail 一時拒否
accept 受信	discard 廃棄
reject 拒否	quarantine 隔離 （実はアクションじゃない）

メッセージ変更

From変更	ヘッダー追加
To追加	ヘッダー削除
To削除	ヘッダー変更
本文変更	

テンプレート

milter

どのmilterを使う？

ステージ

どのタイミングで？

アクション

どんな結果？

結果は？

1. 処理続行
2. 受信
3. 拒否
4. その他

ケース1: accept

milter

spamass-milter

ステージ

mail from

アクション

accept

SMTP Authしているとき

結果は?

1. 処理続行
2. 受信
3. 拒否
4. その他

ケース1: 答え

結果

2. 受信

解説

- ✓ milterの処理は終了
- ✓ rcpt to以降のデータ
  → 渡ってこない

結果は？

1. 処理続行
2. 受信 
3. 拒否
4. その他

ケース2: temp fail

milter

milter-greylist

ステージ

rcpt to

アクション

temp fail

グレイリスト

結果は?

1. 処理続行
2. 受信
3. 拒否
4. その他

ケース2: 答え

結果

4. その他

解説

- ✓ その宛先のみtemp fail
- ✓ 全宛先をtemp fail
  → メッセージ全体を
  temp fail

結果は？

1. 処理続行
2. 受信
3. 拒否
4. その他

ケース3: quarantine

milter

clamav-milter

ステージ

end of message

アクション

quarantine
ウィルス検出時

結果は?

1. 処理続行
2. 受信
3. 拒否
4. その他

ケース3: 答え

結果

4. その他

解説

- ✓ 受信する
- ✓ 配信はしない
  キューに入っている

結果は？

1. 処理続行
2. 受信
3. 拒否
4. その他

ケース4: ヘッダー

milter

ENMA

ステージ

end of message

アクション

ヘッダー追加
continue

結果は?

1. 処理続行
2. 受信
3. 拒否
4. その他

ケース4: 答え

結果

2. 受信

解説

- ✓ 次のステージがない
- ✓ 受信する

結果は？

1. 処理続行
2. 受信 
3. 拒否
4. その他

ケース5: ヘッダー

milter

spamass-milter + ENMA

ステージ

end of mesasge

アクション

spamass: ヘッダー変更
continue
Subjectに[SPAM]を追加

結果は?
1. 処理続行
2. 受信
3. 拒否
4. その他

ケース5: 答え

結果

1. 処理続行

解説

- ✓ 処理はENMAへ
- ✓ DKIM検証は失敗
  Subjectが変わっている

結果は？

1. 処理続行
2. 受信
3. 拒否
4. その他

milter利用時のポイント

- 連携は意外と難しい
  - 詳細な結果: end of message待ち
  - reject/temp fail: すぐに終了
- DKIMは順番が大事
  - 署名時も検証時も

milter managerの特徴

強力なmilter管理機能

- milter検出
- milterテスト機能
- 動的にmilterをoff
- milter評価モード
  - reject/temp failなどを無視
- 共通ホワイト/ブラックリスト
- 統計情報収集

管理例: ユーザ毎の設定

- MySQLにユーザ設定を格納
- ユーザ毎に利用する対策を変更
  - 必要ないmilterをoff
- ユーザ毎のブラックリスト
  - NGワードとか受信ドメインとか

ユーザ毎の設定: メリット

- MySQLとの接続を一本化
  - 各milterが接続するより効率的
- 管理画面が作りやすい
  - MySQL内のデータを変更するだけ
  - よくあるWebアプリケーション
- 環境によっては必須機能
  - ISPとか

まとめ

- 迷惑メールの多様化
  - 複数のmilterで対策
- 複数milter利用時の挙動を確認
  - 本文は直列
  - 本文以外は並列
- 複数milterの管理には milter manager

お知らせ

- 開発者募集中
  - 応募条件: プログラミングが好きな事
- OSSでお困りのことがあれば
  ご相談を