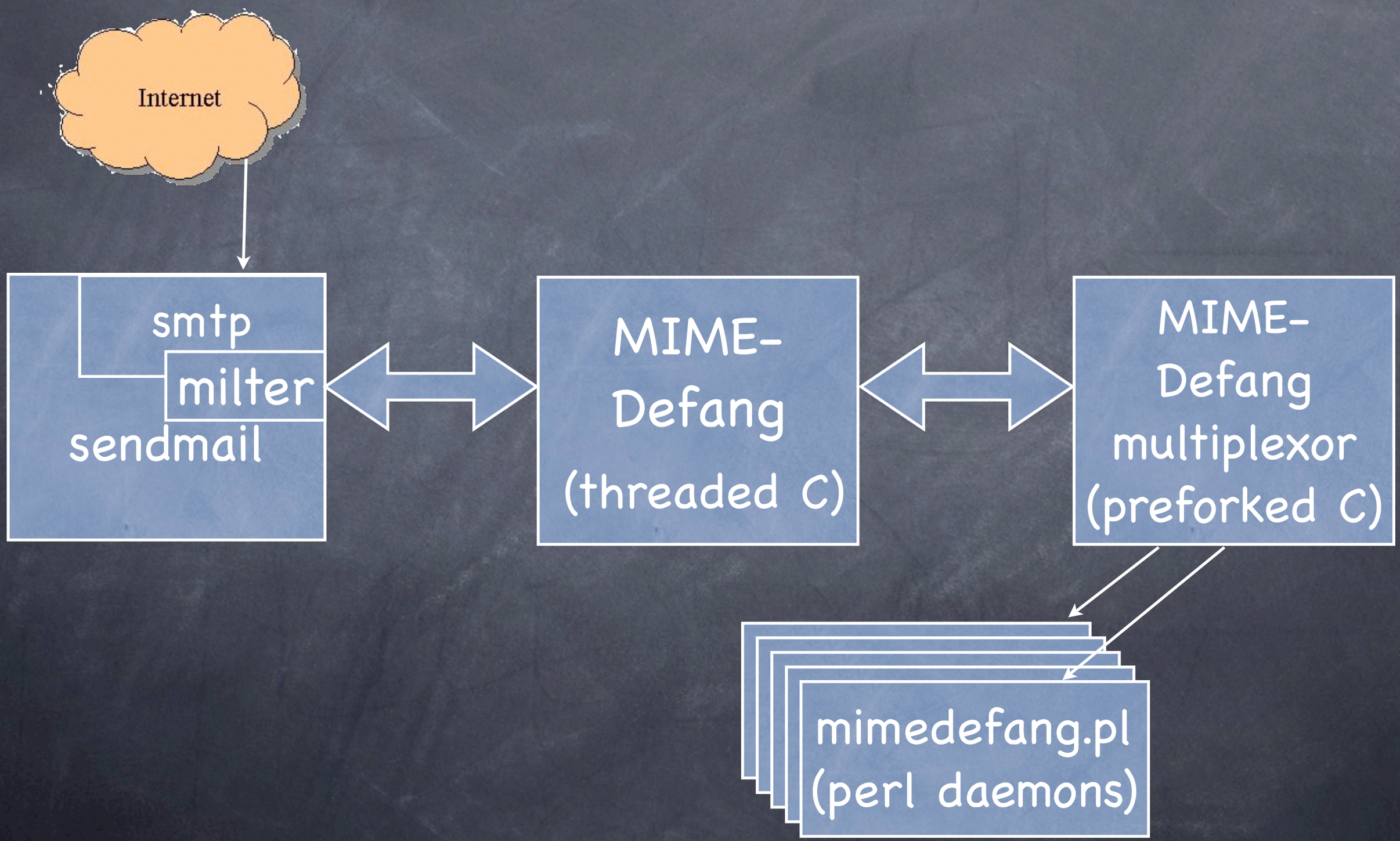
Spam filtering with perl using MIMEDefang

Jan-Pieter Cornet
http://www.xs4all.nl/~johnpc/
johnpc@xs4all.nl

MIMEDefang architecture



MIMEDefang basics

- Filter written in perl
- Runs continuously as daemons: fast
- well-maintained, good community support
- Allows interaction with SMTP protocol at any point

smtp example

220 my.domain.name ESMTP

smtp example

421 Bugger off, adsl13-105-42-7.tampa.fl.rr.com,
or get yourself a real reverse DNS.

smtp example

220 my.domain.name ESMTP
HELO aol.com
250 OK
MAIL From:<fake@fake.id>
571 Domain fake.id does not exist

smtp example

220 my.domain.name ESMTP
HELO aol.com
250 OK
MAIL From:<fake@fake.id>
250 Sender OK
RCPT To:<user@our.domain>
451 Quota exceeded

smtp example

220 my.domain.name ESMTP
HELO aol.com
250 OK
MAIL From:<fake@fake.id>
250 Sender OK
RCPT To:<user@our.domain>
551 Quota exceeded

smtp example

220 my.domain.name ESMTP
HELO aol.com
250 OK
MAIL From:<fake@fake.id>
250 Sender OK
RCPT To:<user@our.domain>
250 Recipient OK
RCPT To:<user2@our.domain>

smtp example

220 my.domain.name ESMTP
HELO aol.com
250 OK
MAIL From:<fake@fake.id>
250 Sender OK
RCPT To:<user@our.domain>
250 Recipient OK
DATA
354 Send DATA, end with .
....

smtp example

MAIL From:<fake@fake.id>
250 Sender OK
RCPT To:<user@our.domain>
250 Recipient OK
DATA
354 Send DATA, end with .
....
.
571 Virus W32/klez.B detected

Virus scanning

- 3 x virus scanning: ClamAV and 2 commercial.
- All scanners detect about the same number of viruses
- upto 25% of virus-detected mails not picked up by all 3 (broken bounces, bad disinfects)

Virus scanning

- known non-From-forging virus (eg: Word Macro): REJECT (550 virus $name detected)
- Most viruses are header-forging (>99%)

Virus scanning

- header forging virus detected by 1 scanner: TEMPFAIL (451 possibly virus $name)
- header forging virus detected by 2 or more: DISCARD (250 you made /dev/null happy)

greenlists

- If From: address on greenlist, no spam-scanning is done.

BogusMX

- some.domain. MX 0 dev.null.
- some.domain. MX 10 localhost.
- some.domain. MX 10 valid.looking.name.
  valid.looking.name. A 127.1.2.3

SpamAssassin rant

- scoring: vague, random, or wrong
- small message chunks: lines, paragraphs
- bayes unusable on large scale
- internals horrifyingly inconsistent
- worth 100% of cost

fsck*ng spammers

Via<div style="float:

right">X gra

SpamAssassin to the rescue!

```
body DRUG_ED_CAPS      /\bCIALIS|LEVITRA|VIAGRA/

...
    if ( $body_of_rule_X ) {
        $self->get_pattern_hit(`rule_X');
    }
...
```

SpamAssassin to the rescue!

```
body DRUG_ED_CAPS      /\bCIALIS|LEVITRA|VIAGRA/

...
    if ( /\bCIALIS|LEVITRA|VIAGRA/ ) {
        $self->got_pattern_hit('DRUG_ED_CAPS')
    }
...
```

SpamAssassin to the rescue!

```
body MY_HACK_RULE       do { print "0wn3d" }

...
    if ( do { print "0wn3d" } ) {
        $self->got_pattern_hit('MY_HACK_RULE')
    }
...
```

SpamAssassin Hack

```
rawbody XS_LEVANTINE                    do { my $x=(($self->{_SCRATCH_}
||={})->{XS_LEVANTINE}||={}); $x->{MSG} = substr(($x->{MSG}||"").'$_ ",
-10240); my @c = $x->{MSG} =~ m{<(?:div|span)[^>]+\bstyle\s*=\s*["']?(?:
[^"']*;)?\s*(?:float\s*:\s*right|display\s*:\s*none)\b[^"']*["'][^>]*>}ig; @c >
15 }
describe XS_LEVANTINE           Uses excessive <div style="float: right">
score XS_LEVANTINE               5.0
```

SpamAsssassin Hack

- No verification?
- if (eval "m{$pattern}") { OK }
- hole plugged in 3.1
- Solution: plugins.
- perldoc -i mail::spamassassin::plugin