Mailserver-Konferenz 2014

PDF: http://amavis.org/Z4

Mark Martinec
Institut “Jožef Stefan”, Slovenia

Amavis and SpamAssassin

Agenda

- what it is
- plumbing pieces together
- IPv6 anyone?
- IP address and domain reputation
- monitoring (SNMP and status), ZMQ
- JSON structured log, Elasticsearch, Kibana
- demo (ES, Kibana)

Amavis - what is it?

- interface between MTA and
  virus checkers and/or spam checkers
- like *spamd* for SA, but speaks standard SMTP
- both spamd and amavisd use Mail::SpamAssassin
- checks for banned content and header syntax
- quarantining/archiving
- DKIM: signs and verifies signatures
- monitoring: SNMP, status, logging (SQL, JSON)

Policy banks

- one global, currently in effect,
  set of configuration variables

- several replacement sets (groups)
  of configuration variables,
  prepared in advance and on stand-by,
  quickly loadable

- affects message as a whole (not per-recipient)

Policy banks – examples

```
$policy_bank{'NOVIRUSCHECK'} = {
    bypass_decode_parts => 1,
    bypass_virus_checks_maps => [1],
    virus_lovers_maps => [1],
};

$policy_bank{'AM.PDP-SOCK'} = {
    protocol => 'AM.PDP',
    auth_required_release => 0,
    syslog_ident => 'amavis-release',
};
```

Policy banks – example

```
$policy_bank{'ALT'} = {
  originating  => 1,
  log_level  => 2,
  forward_method => 'smtp:*:*',
  local_client_bind_address  => '193.2.4.6',
  localhost_name   => 'extra.example.com',
  final_spam_destiny  => D_PASS,
  spam_kill_level_maps  => 6.72,
}
```

Policy banks – activating by port no.

```
$inet_socket_port =
   [10024, 10026, 10028, 10030, 9998];

$interface_policy{'10026'} = 'ORIGINATING';
$interface_policy{'10028'} = 'NOCHECKS';
$interface_policy{'10030'} = 'CUSTOMER';
$interface_policy{'9998'} = 'AM.PDP-INET';
$interface_policy{'SOCK'} = 'AM.PDP-SOCK';
```

Policy banks – implicitly MYNETS

```
@mynetworks = qw(
   0.0.0.0/8 127.0.0.0/8 [::1]
   10.0.0.0/8 172.16.0.0/12 192.168.0.0/16
   192.0.2.0/24 [2001:db8::/32]
);
```

implicitly loads policy bank MYNETS

if it exists

Policy banks – by DKIM signature

```
@author_to_policy_bank_maps = (
{   'uni-bremen.de'   => 'WHITELIST',
    'tu-graz.ac.at'     => 'WHITELIST',
    '.ebay.com'         => 'WHITELIST',
    '.paypal.com'       => 'WHITELIST',
    'amazon.com'        => 'WHITELIST',
    'cern.ch'           => 'SPECIAL',
    '.linkedin.com'     => 'MILD_WHITELIST',
    'dailyhoroscope@astrology.com'
                        => 'MILD_WHITELIST',
} );
```

Policy banks – by custom hook

```
sub new {
    my($class, $conn, $msginfo) = @_;
    my($self) = bless {}, $class;
    if ( ... ) {
        Amavis::load_policy_bank(
                                'NOVIRUSCHECK' );
    }
    $self;
}
```

Lookup tables

Static:

- associative array (Perl hash)
- a list (a.k.a. ACL) (Perl list)
- list of regular expressions (object: list of *re*)
- constant (Perl scalar)

Dynamic:

- SQL, LDAP (Perl object)

Lists of lookup tables: @*_maps

@local_domains_maps = (...)

Typically used to provide by-recipient settings, even in multi-recipient mail.

Remember:

- policy banks affect message as a whole, so can only depend on some common characteristic of a message, e.g. client's IP address, sender address / DKIM, TCP port number

- lookups / @*_maps serve to implement per-recipient settings (and some other things)

pre-queue filtering – Postfix

```
smtp    inet  n       -       n       -       1       postscreen

smtpd   pass  -       -       n       -       150     smtpd
    -o smtpd_proxy_filter=inet:[::1]:10010
    -o smtpd_proxy_options=speed_adjust
```

amavisd: listen, policy, forward

```
$inet_socket_bind = '[::1]';
$inet_socket_port = [10010, 10012, 10026];

$interface_policy{'10010'} = 'PROXY-MX';

$policy_bank{'PROXY-MX'} = {
  forward_method => 'smtp:[::1]:10011' ,
};
```

back to postfix

```
[::1]:10011  inet  n  -  n  -  -  smtpd
-o smtpd_authorized_xforward_hosts=[::1]
```

Mailing list

Mailman: mm_cfg.py

```
SMTPHOST = '::1'
SMTPPORT = 10088  # to Postfix, to be signed
```

```
[::1]10088  inet  n  -  n  -  -  smtpd
-o content_filter=smtp:[::1]:10028
```

Amavis: DKIM signing only

```
$interface_policy{'10028'} = 'MLIST-NOCHECKS';

$policy_bank{'MLIST-NOCHECKS'} = {
  originating => 1,
  bypass_decode_parts => 1,
  bypass_header_checks_maps => [1],
  bypass_virus_checks_maps  => [1],
  bypass_spam_checks_maps   => [1],
  bypass_banned_checks_maps => [1],
  forward_method => 'smtp:[::1]:10025',
  signed_header_fields => { 'Sender' => 1 },
}
```

Is it slow?

- written in Perl
- fast on large chunks of data
  (Mail::DKIM slightly faster than Mail::OpenDKIM)
  (4x SMTP read 2.7.0)
- avoid line-by-line processing (qpsmtpd)
- avoid copying data
- 100 KiB in memory, large mail on temp. file

- critical code paths are well optimized
- the slow part is SpamAssassin, if enabled

Amavis to SpamAssassin API

- passes message to SA as a ref or file descriptor
- supplies additional information (%suppl_attr)
  - DKIM validation results
  - smtp envelope data (mail_from, rcpt_to)
  - originating flag (treated as msa_networks, 3.4.0)
  - original size, truncation flag
  - timing deadline
  - policy bank names
  - synthesized rule hits (optional)

Network latency a problem?

- DNS black and white lists (DNSxL)
- DCC, Razor, Pyzor network services
- SPF, DKIM, DMARC, ADSP

The bottleneck in SpamAssassin is CPU, idle wait times are compensated by running more processes, the only cost is memory.

Optimizations and Profiling

- Devel::NYTProf (Tim Bunce) 5.06

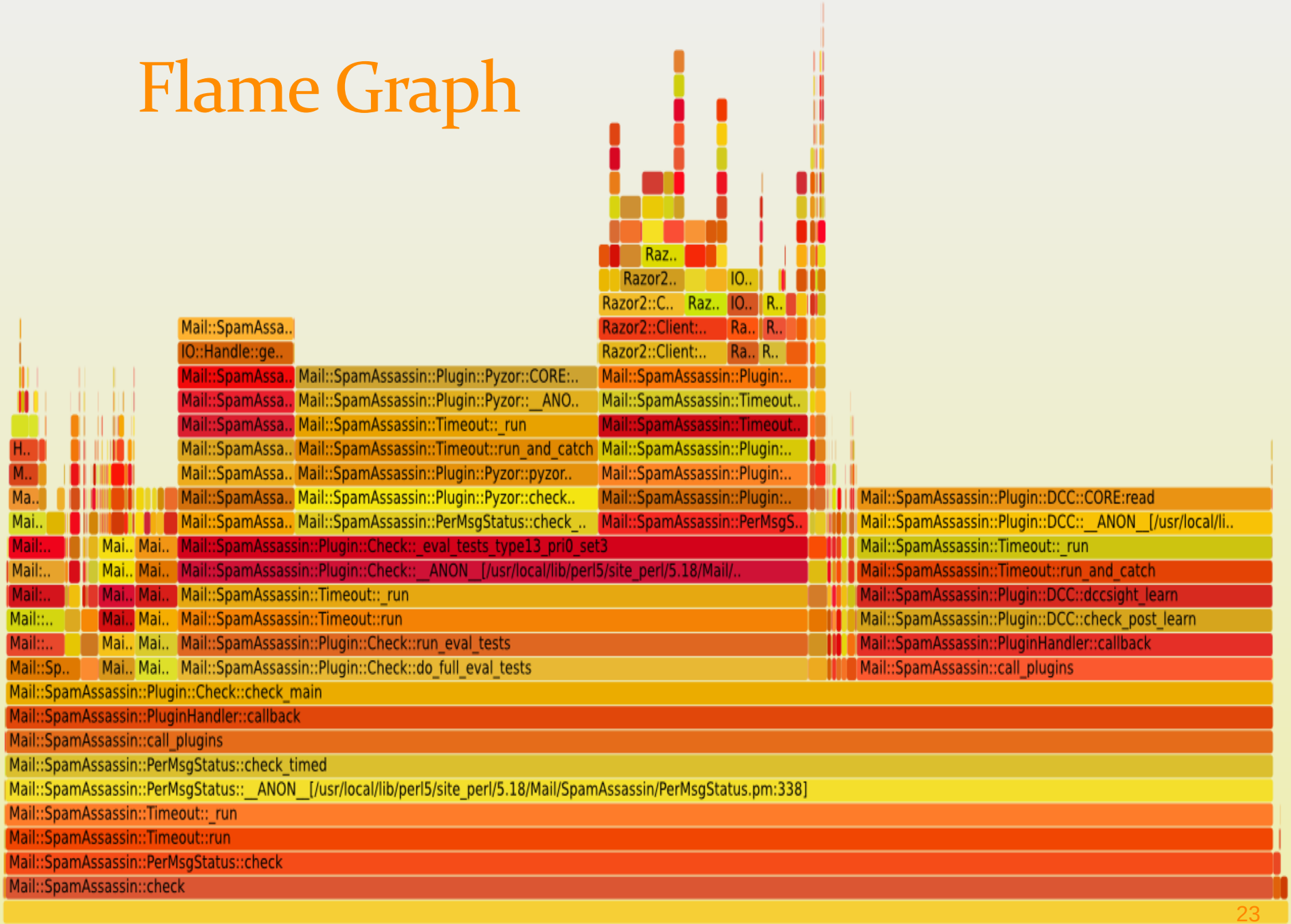
Powerful fast feature-rich Perl source code profiler

- Flame Graphs: (Brendan Gregg Joyent)

Blazing Performance with Flame Graphs

LISA 2013

Flame Graph


Raz..
Razor2..
IO..
Razor2::C..
Raz..
IO..
R..
Mail::SpamAssa..
Razor2::Client:..
Ra..
R..
IO::Handle::ge..
Razor2::Client:..
Ra..
R..
Mail::SpamAssa..
Mail::SpamAssassin::Plugin::Pyzor::CORE:..
Mail::SpamAssassin::Plugin:..
Mail::SpamAssa..
Mail::SpamAssassin::Plugin::Pyzor::__ANO..
Mail::SpamAssassin::Timeout..
Mail::SpamAssa..
Mail::SpamAssassin::Timeout::_run
Mail::SpamAssassin::Timeout..
H..
Mail::SpamAssa..
Mail::SpamAssassin::Timeout::run_and_catch
Mail::SpamAssassin::Plugin:..
M..
Mail::SpamAssa..
Mail::SpamAssassin::Plugin::Pyzor::pyzor..
Mail::SpamAssassin::Plugin:..
Ma..
Mail::SpamAssa..
Mail::SpamAssassin::Plugin::Pyzor::check..
Mail::SpamAssassin::Plugin:..
Mail::SpamAssassin::Plugin::DCC::CORE:read
Mai..
Mail::SpamAssa..
Mail::SpamAssassin::PerMsgStatus::check_..
Mail::SpamAssassin::PerMsgS..
Mail::SpamAssassin::Plugin::DCC::__ANON__[/usr/local/li..
Mail:..
Mai..
Mai..
Mail::SpamAssassin::Plugin::Check::_eval_tests_type13_pri0_set3
Mail::SpamAssassin::Timeout::_run
Mail:..
Mai..
Mai..
Mail::SpamAssassin::Plugin::Check::__ANON__[/usr/local/lib/perl5/site_perl/5.18/Mail/..
Mail::SpamAssassin::Timeout::run_and_catch
Mail:..
Mai..
Mai..
Mail::SpamAssassin::Timeout::_run
Mail::SpamAssassin::Plugin::DCC::dccsight_learn
Mail:..
Mai..
Mai..
Mail::SpamAssassin::Timeout::run
Mail::SpamAssassin::Plugin::DCC::check_post_learn
Mail:..
Mai..
Mai..
Mail::SpamAssassin::Plugin::Check::run_eval_tests
Mail::SpamAssassin::PluginHandler::callback
Mail::Sp..
Mai..
Mai..
Mail::SpamAssassin::Plugin::Check::do_full_eval_tests
Mail::SpamAssassin::call_plugins
Mail::SpamAssassin::Plugin::Check::check_main
Mail::SpamAssassin::PluginHandler::callback
Mail::SpamAssassin::call_plugins
Mail::SpamAssassin::PerMsgStatus::check_timed
Mail::SpamAssassin::PerMsgStatus::__ANON__[/usr/local/lib/perl5/site_perl/5.18/Mail/SpamAssassin/PerMsgStatus.pm:338]
Mail::SpamAssassin::Timeout::_run
Mail::SpamAssassin::Timeout::run
Mail::SpamAssassin::PerMsgStatus::check
Mail::SpamAssassin::check

Recent optimizations

- SQL replaced by Redis :
  pen pals, Bayes, log queue , IP addr. reputation

- Redis.pm from CPAN replaced by own code
  (100x fewer packets, 50 % speedup, Bug 6972)
- heavy use of server-side Lua scripting

- BerkeleyDB replaced by ZMQ
  message passing, avoids lock contention
  nanny / amavisd-status, SNMP

Speaking of Redis …

Redis is an open source, BSD licensed, advanced key-value store. It is often referred to as a data structure server since keys can contain strings, hashes, lists, sets and sorted sets.

- Lua scripting (server side) – fast, less traffic
- automatic key expiration – no bookkeeping
- in-memory, with optional persistence

Speaking of ZMQ (Zero M Q)

- message passing paradigm
- low level, fast, no broker
- used for monitoring and statistics gathering purposes – avoids database locking!

- ZMQ since amavisd-new-2.8.0,
  works with versions 2, 3, 4
- supports IPv6 since version 3, a bit awkwardly

IPv6 – why bother?

- is here to stay, needs to be supported
- high time to learn & gather experience
  - developers: now! if not already
  - admins: now, to avoid later surprises and panic

- dual stack? need to support both?
- greenfield (new) deployment:

Is an IPv6-only datacenter (internally) feasible?

Time to try it out!

IPv6 internally – experience

what better way to test it:

disable IPv4 and see what breaks!

- needs dual-stack recursive DNS server
- /etc/resolv.conf: nameserver ::1 (mostly harmless::)
- clamd only over Unix socket (next version?)
- freshclam & infrastructure ok, manual config
- may need dual-stacked web proxy (e.g. virus db updates)
- Razor and Pyzor do not work

IPv6 internally – experience

internal communication between components works just fine, no need for 127.0.0.1

amavisd / postfix / mailman / net-snmp / p0f v3/ DNS resolver / ZMQ / Redis / SQL / LDAP / Elasticsearch

IPv6 support in Amavis

- good, thank you!

- initial support for IP address forms in 2004
- sockets: Net::Server 2.0.0 (May 2012)
- details improved over time
- works well also in an IPv6-only environment

IPv6 support in SpamAssassin

- improved over time, pretty good
- spamc & spamd over IPv6 since 3.4.0
- sa-update and rules mirrors
- module IO::Socket::IP preferred
- Redis client module rewrite (TinyRedis)
- RelayCountry: needs Geo::IP 1.39+ with C API
- DNS, AWL, DCC, TxRep

IPv6 in Perl

pretty good in more recent versions

- Socket module (old Socket6)
- IO::Socket::IP (old IO::Socket::INET6)
- NetAddr::IP, Net::Server, Net::DNS, Net::LDAP

- some modules still use IO::Socket::INET6 and Socket6 (e.g. Net::Server, Net::DNS)
- IO::Socket::IP preferred (Amavis, SA, Net::LDAP, ...)
- some modules are a lost cause, e.g. LWP

IPv6 externally: mail

- dual stack is a norm
- well supported by MTAs, IMAP, POP3, ...

- alternative: stay IPv6-only internally, using NAT64, or SMTP & IMAP proxy to interface with external world

IPv6 and mail

later presentation

by Andreas Schulze and Peer Heinlein

IPv6 & Google

2013-12:

- 91.4 % of non-spam e-mails sent to Gmail users come from authenticated senders
  (74.7 % SPF+DKIM,
  14.4 SPF only, 2.25 % DKIM only)
- less than 8.6% of legit. e-mail unauthenticated

http://googleonlinesecurity.blogspot.com/2013/12/internet-wide-efforts-to-fight-email.html

IP reputation > domain reputation

- fewer domains than /64 networks
- blocking /64 likely to cause collateral damage
- domain allocation is more granular and portable than IP space allocation

- domain authentication:
  - Sender Policy Framework (SPF)
  - Domain Keys Identified Message (DKIM)

- if reverse DNS is enforced, then there is less need for DNSBL

IP reputation > domain reputation

large players seize this greenfield opportunity to shift sender reputation from an IP address to a domain name (DKIM, SPF, DMARC)

Google, Yahoo!, Aol, LinkedIn, ...

IPv6 & Google

2013-09, Additional guidelines for IPv6

- The sending IP must have a PTR record (i.e., a reverse DNS of the sending IP) and it should match the IP obtained via the forward DNS resolution of the hostname specified in the PTR record. Otherwise, mail will be marked as spam or possibly rejected.
- The sending domain should pass either SPF check or DKIM check. Otherwise, mail might be marked as spam.
- Sign messages with DKIM. We do not authenticate messages signed with keys using fewer than 1024 bits.

IPv6 & LinkedIn

- Franck Martin, 2014-03

http://engineering.linkedin.com/email/sending-and-receiving-emails-over-ipv6

- SMTP over IPv6 at LinkedIn (2014-04-03):

http://www.slideshare.net/FranckMartin/linkedin-smtpi-pv6

IPv6 & SPF

SPF does recognize IPv6

"v=spf1 ip6:1080::8:800:200C:417A/96 -all"

SPF wiki page is wrong, or misleading at best
http://www.openspf.org/SPF_Record_Syntax

The SPF specification in RFC 4408 is a reference, talks about a (generic) <ip> address

Cherish a DKIM signature

Don't let mailing lists break DKIM signature:

- don't add footers
- don't modify Subject
- don't modify From
- preserve existing Sender and Reply-To

DMARC

Domain-based Message Authentication, Reporting and Conformance

2014-04-02: The DMARC specification has been submitted as an Informational Document to the RFC Independent Submissions Editor.

The result will not be an IETF Internet Standard as originally envisioned by the DMARC.org member organizations. However it will provide a fixed reference document that can provide the basis for development on the Standards track at a later date.

DMARC @ yahoo.com, aol.com

April 2014:

$ dig _dmarc.yahoo.com TXT +short

v=DMARC1; p=reject; sp=none; pct=100; rua=mailto:dmarc-yahoo-rua@yahoo-inc.com, mailto:dmarc_y_rua@yahoo.com;

$ dig _dmarc.aol.com TXT +short

v=DMARC1; p=reject; pct=100; rua=mailto:d@rua.agari.com; ruf=mailto:d@ruf.agari.com;

$ dig _dmarc.gmail.com TXT +short

v=DMARC1; p=none; rua=mailto:mailauth reports@google.com

DMARC with SpamAssassin?

- before you ask ...
  not currently, but planned as a plugin

- but we do have ADSP (with overrides)
- DKIM and SPF results can be combined with rules

Reputation in Amavis

- DKIM-based whitelisting:
  @author_to_policy_bank_maps
  @signer_reputation_maps

- automatic IP reputation with Redis

- Pen pals, now in Redis

IP address reputation in Amavis

- new in 2.9.0
- @storage_redis_dsn
- @ip_repu_ignore_networks
- $enable_ip_repu = 1

- for each IP address seen in Received trace counts number of ham and spam messages
- contributes worst score based on ham/all ratio
- automatic expiration, maintenance-free

Pen pals – sending / remembering

- saves info on originating mail to Redis

  (or in SQL: saves info on *all* mail)

- keyed by mail_id, it stores:
  - sender + Message-ID
  - sender + (each) recipient

Pen pals – receiving / checking

check for ongoing correspondence:

- if the current recipient has recently sent any mail to the sender of the current mail
- In-Reply-To or References
  matching a previous Message-ID

a new message is matched against stored data:

- recipient + sender
- recipient + Message-ID

Pen pals – mail_id_related (2.9.0)

```
mail_id       mail_id_related author        ->  rcpt_to

D3dyEYS8CU5j                  anna@ijs.si     bob@gmail.com

JudMcXgHLpLR  D3dyEYS8CU5j    bob@gmail.com   anna@ijs.si

7vdAHwRqC9rv  D3dyEYS8CU5j    bob@gmail.com   anna@ijs.si
```

parent_mail_id (2.9.0)

```
action mail_id      parent_mail_id   author       rcpt_to
```

notification:

```
REJECT bkEgaWMXMmd1                  xxx          yyy
NOTIF  hN8hKLzwuM1k bkEgaWMXMmd1     virusalert   admin
```

quarantine release:

```
REJECT o7my5KW2A78N                  xxx          yyy
SEND   SbWk6ymnD56D o7my5KW2A78N     yyy
```

and now …

… for something completely different …

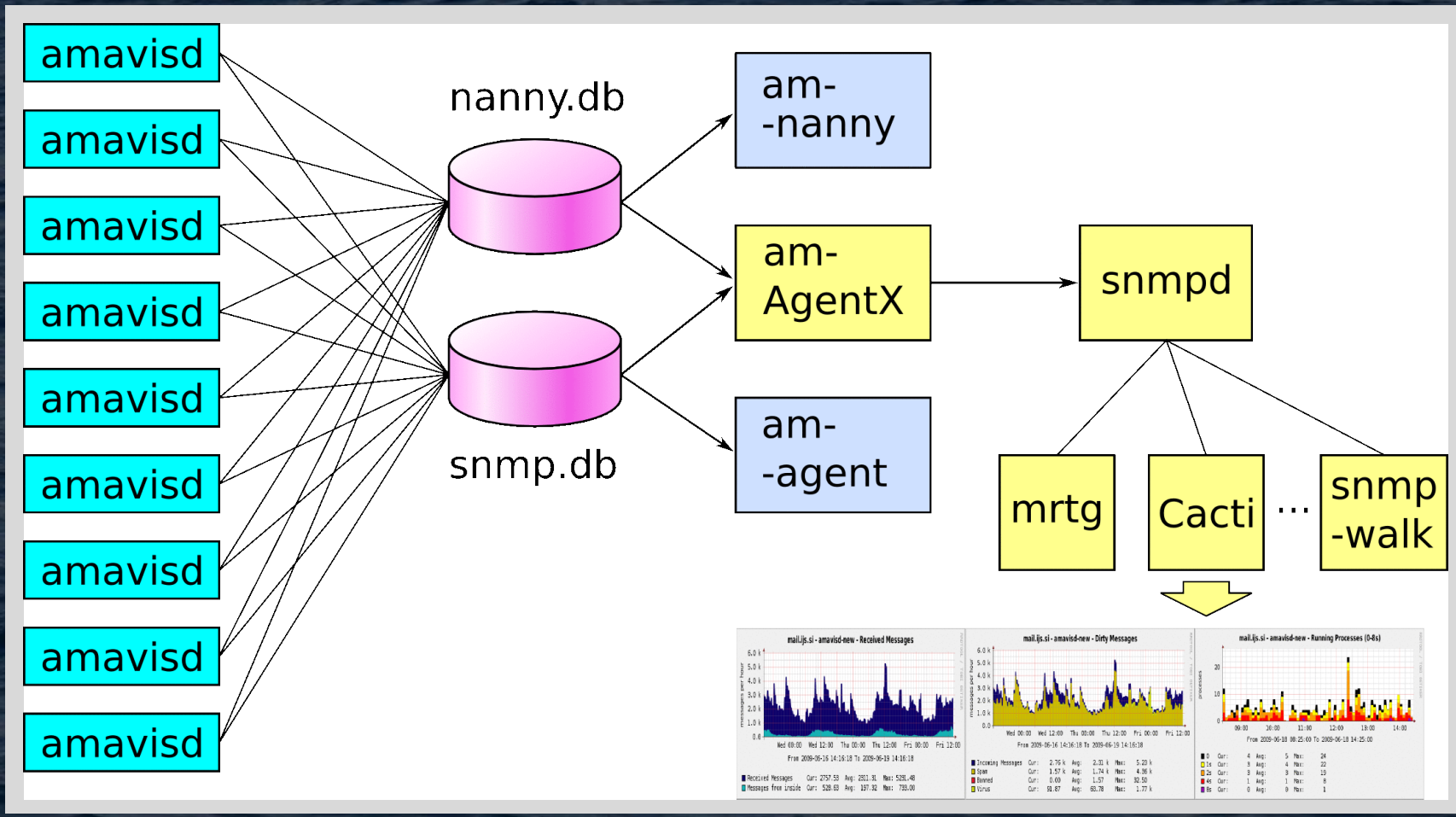
Monitoring

- amavisd-status (ZMQ)
- SNMP counters and gauges (ZMQ)
- traditional logging
- structured log – JSON

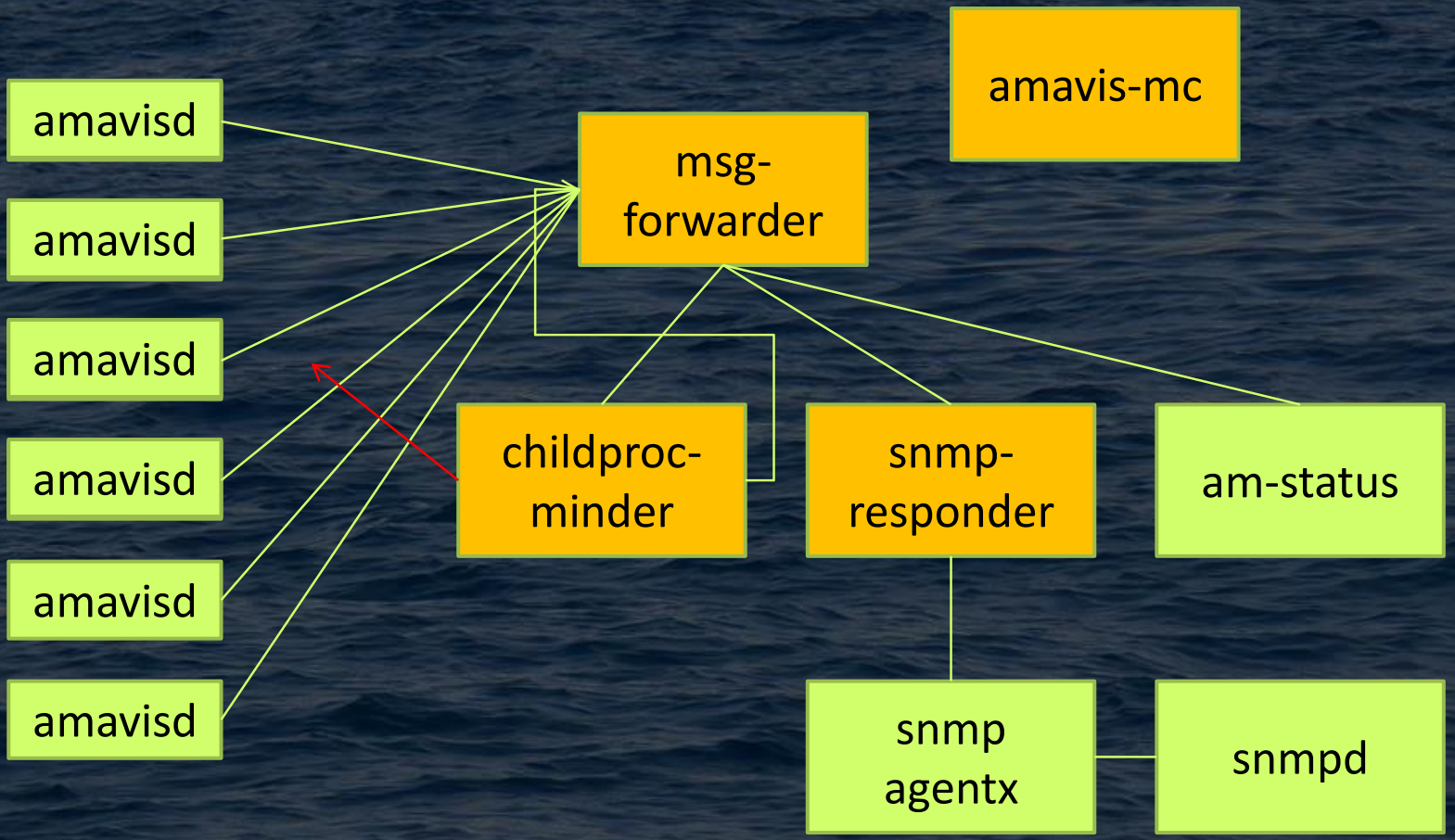
$ amavisd-status (ex: nanny)

```
PID 66913: 66913-03      0:00:03 VVV
PID 66923: 66923-02      0:00:05 SSSSP
PID 66925: 66925-03      0:00:02 VSS
PID 66926: 66926-04      0:00:00 S
PID 66930: 66930-04      0:00:02 SSS
PID 66931: 66931-04      0:00:00 V
PID 66932: A             0:00:00 A
PID 66933: 66933-03      0:00:00 V
PID 66934: 66934-03      0:00:00 S
PID 66938: 66938-03      0:00:00 V
16 active, 34 idling processes
**********.....................
```

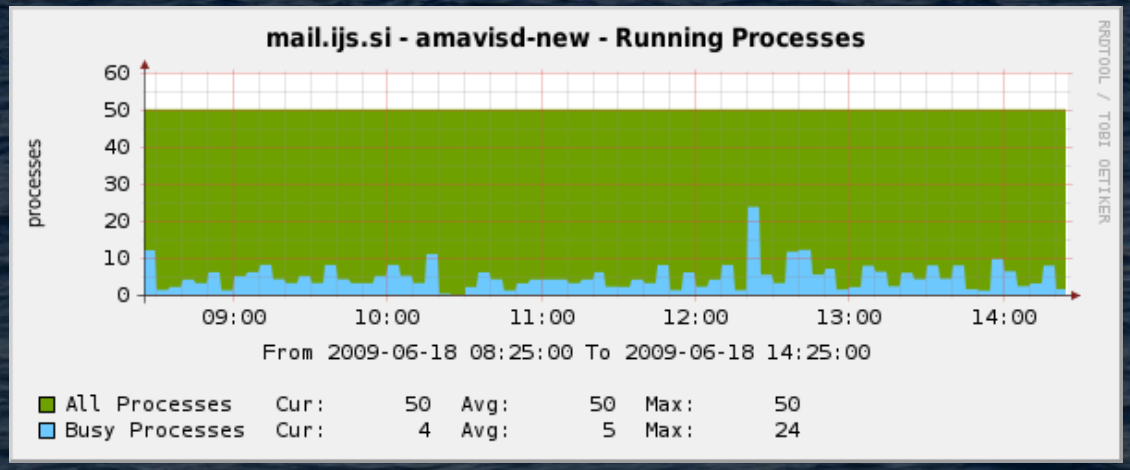
Monitoring components (previous)

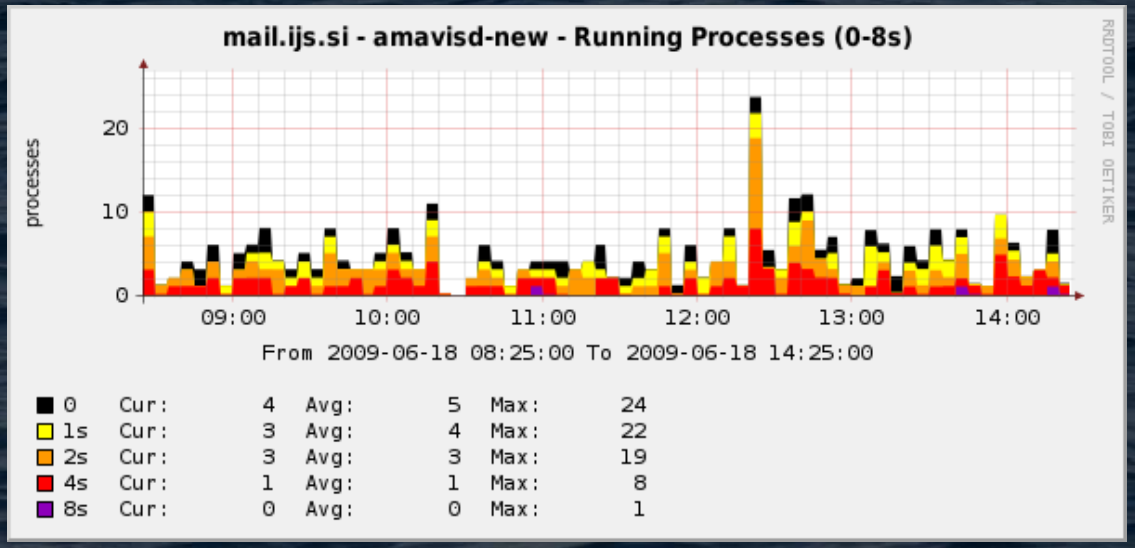


Monitoring components – ZMQ

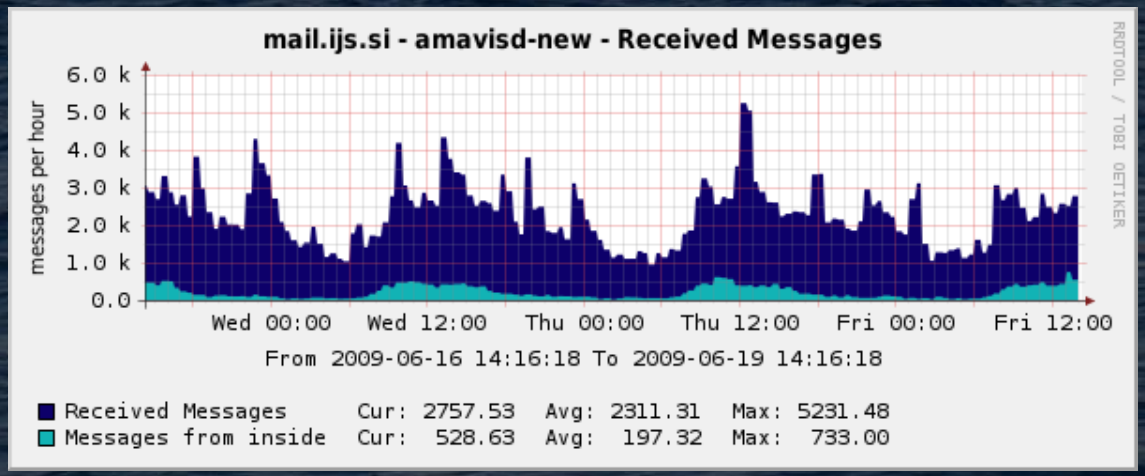


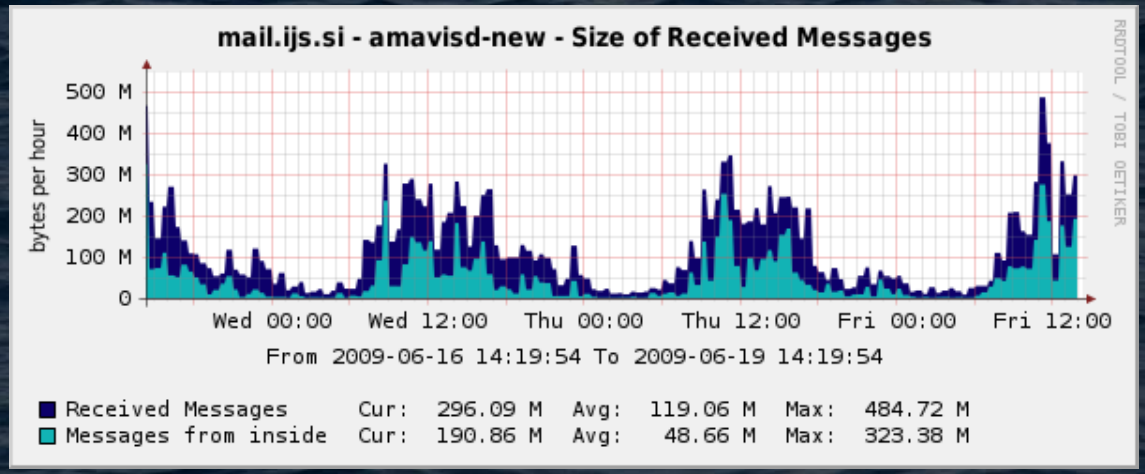
SNMP: load, timing



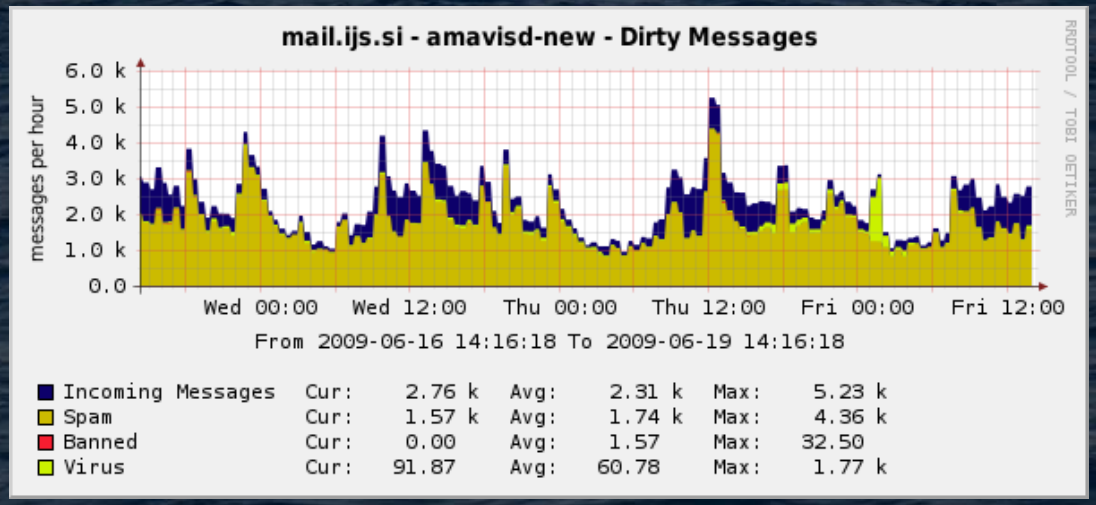


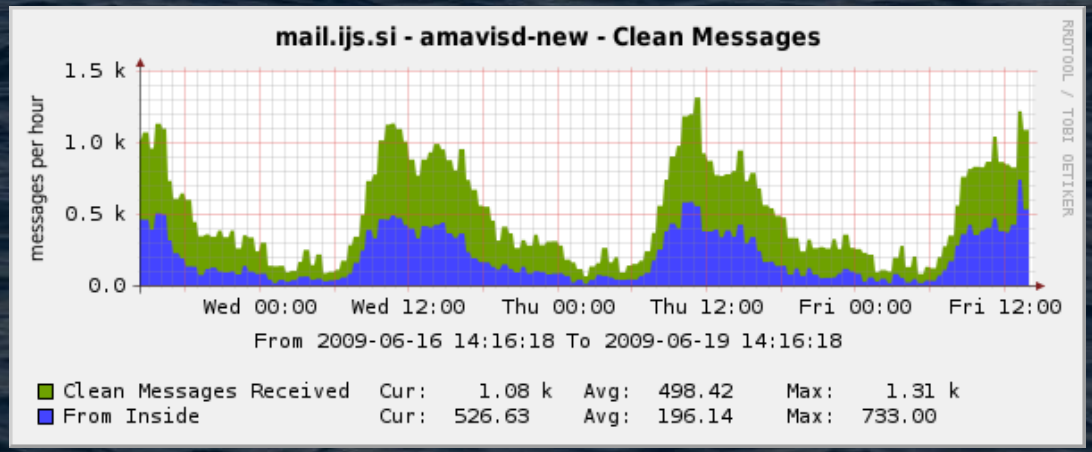
SNMP: mail rate, size



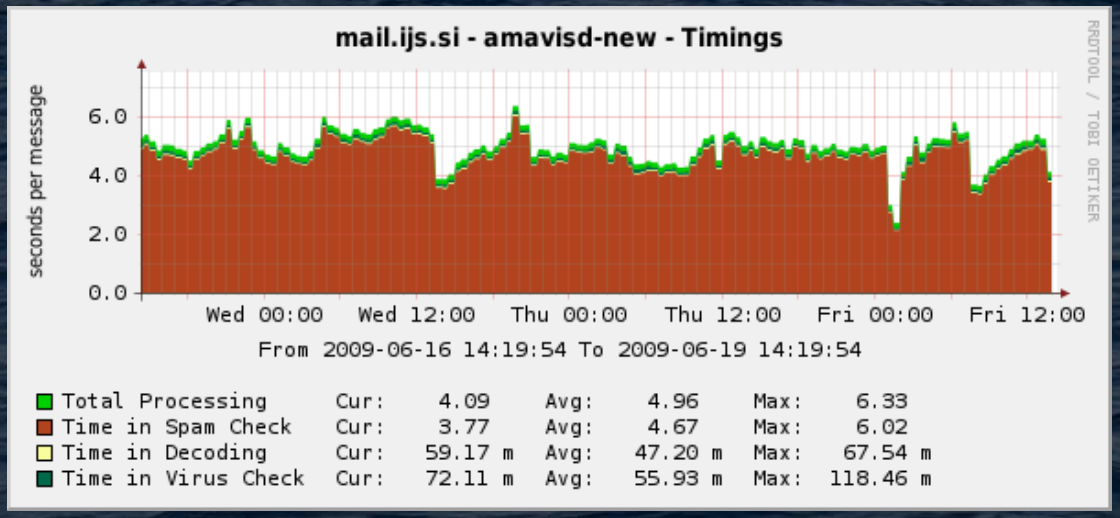


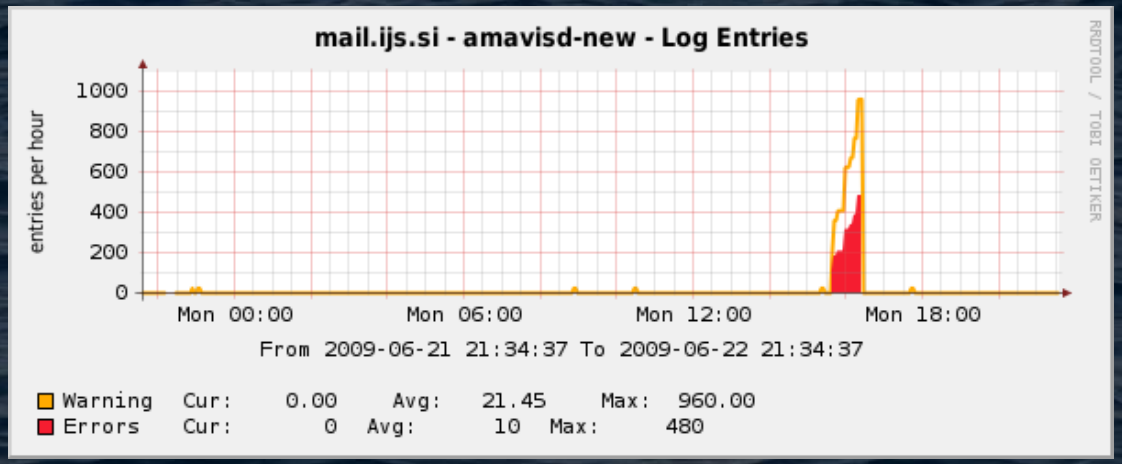
SNMP: mail content



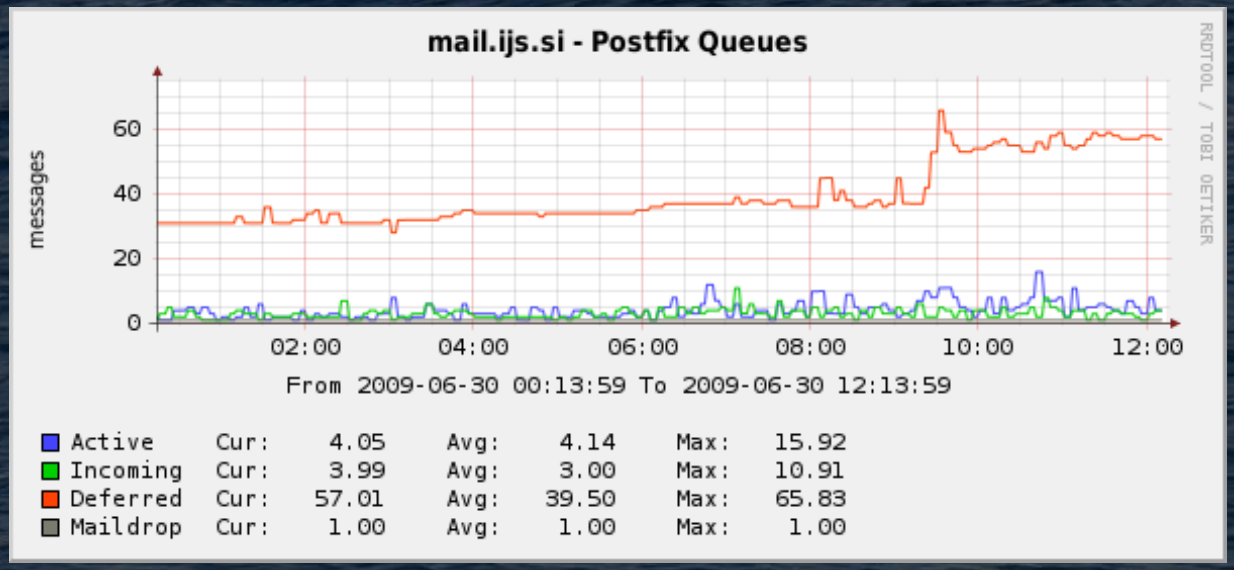


SNMP: elapsed time, errors





SNMP: Postfix queue entries



Traditional logging – syslog levels

SA	amavisd	syslog
	-3	LOG_CRIT
	-2	LOG_ERR
error	-1	LOG_WARNING
warn	0	LOG_NOTICE
info	1	LOG_INFO
	2	LOG_INFO
dbg	3	LOG_DEBUG
	4	LOG_DEBUG
	5	LOG_DEBUG

Traditional logging – timing report

- at log level 2
- elapsed and CPU times (new since 2.8.1, needs module Unix::Getrusage)
- by amavisd sections
- including a report from SpamAssasin

Traditional logging – template

```
$log_templ = <<'EOD';
[?%#D|#|Passed #
[...]
[? %q ||, quarantine: %q]#
[? %Q ||, Queue-ID: %Q]#
[? %m ||, Message-ID: %m]#
[? %r ||, Resent-Message-ID: %r]#
, mail_id: %i#
, Hits: [:SCORE]#
, size: %z#
[...]
EOD
```

Traditional log

fully configurable through a template,
... but a nightmare to parse:

May 8 20:17:58 dorothy amavis[48040]: (48040-03) Blocked SPAM {RejectedInbound,Quarantined}, PROXY-MX/SPAM [216.109.141.91]:3722 [216.109.141.91] <newsletter@123greetings.info> -> <evgeniya.khomyakova@ijs.si>, (216.109.141.91), quarantine: W19/spam/b/bPwQ8bOv-w8m.gz, mail_id: bPwQ8bOv-w8m, b: ABEd6B-Qg, Hits: 17.791, size: 6240, pt: 19, D4, v-SA: Spam, v-CRM: , Subject: "Doctor approved natural Diabetes remedy", From: 123Greetings_<newsletter@123greetings.info>, X-Mailer: Version_5.0, helo=123greetings.info, Tests: [AM.IP_BAD_216.109.141.91=1.2,BAYES_999=4,BAYES_99=3.5,DCC_CHECK=1.1,DIGEST_MULTIPLE=0.293,HTML_MESSAGE=0.001,L_P0F_WXP=2.3,MIME_HEADER_CTYPE_ONLY=0.717,MIME_HTML_ONLY=0.723,MISSING_MID=0.497,RAZOR2_CF_RANGE_51_100=0.5,RAZOR2_CF_RANGE_E8_51_100=1.886,RAZOR2_CHECK=0.922,RCVD_IN_HOSTKARMA_BL=0.3,RCVD_IN_MSPIKE_H2=-0.001,RCVD_NOT_IN_IPREPDNS=0.0001,RP_MATCHES_RCVD=-0.1,SPF_HELO_PASS=-0.001,SPF_PASS=-0.001], shortcircuit=no, autolearn=no autolearn_force=no, autolearnscore=11.366, languages=en,...

May 8 20:17:58 dorothy amavis[48040]: (48040-03) ... relaycountry=US, asn=AS14492_216.109.128.0/19, rss=164560, 3326 ms

Structured log - JSON

```
{   "@timestamp" => "2014-05-06T09:29:47.048Z",
    "time_unix" => 1399368587.048,
    "time_iso_week_date" => "2014-W19-2",
    "partition" => "19",

    "type" => "amavis",
    "host" => "mailer.example.net",
    "src_ip" => "::1",
    "dst_ip" => "::1",
    "dst_port" => 10024,

    "log_id" => "82329-04",
    "mail_id" => "Jnk7NzYB8pvl",
    "mail_id_related" => ["ne27HTERZaOF"],
```

correlates with traditional syslog

unique, quarantine, MTA log

Structured log - JSON

"client_port" => 41831,
"client_ip" => "2001:db8::143:1",
"ip_trace" => ["2001:db8::143:1", "192.0.2.242"],
"os_fp" => "Windows XP; dist: 6; raw_mtu: 1340; ...",

"originating" => true,
"policy_banks" => ["PROXY-ORIGINATING", "MYNETS"],
"size" => 302694,
"digest_body" => "a4a7db6307c140b12f57feaf076663f8",
smtp envelope:
"mail_from" => "mailing-list-1@example.com",
"rcpt_to" => ["recip2@example.org", "recip1@example.net"],

Structured log – JSON

mail header section:

```
"message_id" => "<003701cf690d0@example.com>",
"author"  => ["sending-user@example.com"],
"to_addr" => ["recip1@example.net"],
"cc_addr" => ["recip2@example.org"],

"subject"          => "Fw: An example 123 - test",
"subject_rot13" => "Sj: Na rknzcyr 123 - grfg",

"user_agent" => "Microsoft Office Outlook 12.0",
"is_bulk"  => true,
"is_mlist" => true,
```

Structured log – JSON

```
"action" => ["PASS"],
"actions_performed" => "RelayedInternal RelayedOutbound",
"checks_performed" => "V S H B F P",
"content_type" => "Clean",
"dkim_new_sig" => ["example.com"],
"dsn_sent" => false,
"elapsed" => {
        "Receiving"     => 0.009,
        "Decoding"      => 0.053,
        "VirusCheck"    => 0.326
        "SpamCheck"     => 2.116,
        "Sending"       => 0.118,
        "Amavis"        => 0.215,
        "Total"         => 2.672, },
```

Structured log – JSON

```
"recipients" => [

    { "action" => "PASS",  "ccat_main" => "Clean",
     "queued_as" => "3gNFyR4Mfjzc3",
     "rcpt_to" => "recip2@example.org", "rcpt_is_local" => false,
     "smtp_response" => "250 2.0.0 Ok: queued as 3gNFyR4Mfjzc3",
     "smtp_code" => "250",   "spam_score" => -2.0
    },

    { "action" => "PASS",  "ccat_main" => "Clean",
     "mail_id_related" => "7HRZaOF",  "penpals_age" => 1114599,
     "queued_as" => "3gNFyR4n6Lzc4",
     "rcpt_to" => "recip1@example.net", "rcpt_is_local" => true,
     "smtp_response" => " 250 2.0.0 Ok: queued as 3gNFyR4n6Lzc4",
     "smtp_code" => "250",  "spam_score" => -5.272
    },
]
```

Structured log – JSON

"spam_score" => -2.0, (summary across all recipients)
"smtp_code" => ["250"],
"queued_as" => ["3gNFyR4Mfjzc3", "3gNFyR4n6Lzc4"],

"message" => (short summary)
  "82329-04 PASS Clean <mailing-list-1@example.com>
   -> <recip2@example.org>,<recip1@example.net>",

"tests" => ["ALL_TRUSTED", "AM.PENPAL", "BAYES_00",
  "MSGID_MULTIPLE_AT", "RP_MATCHES_RCVD"],
"tests_ham" => ["AM.PENPAL","BAYES_00","ALL_TRUSTED"],
"tests_spam" => ["MSGID_MULTIPLE_AT"],

Structured log – JSON

- better monitoring
- spotting anomalies
- easier troubleshooting
- finding a needle in a haystack

Structured log - JSON

understood by log analysis tools

- Logstash > Elasticsearch & Lucene > Kibana
- Splunk
- …

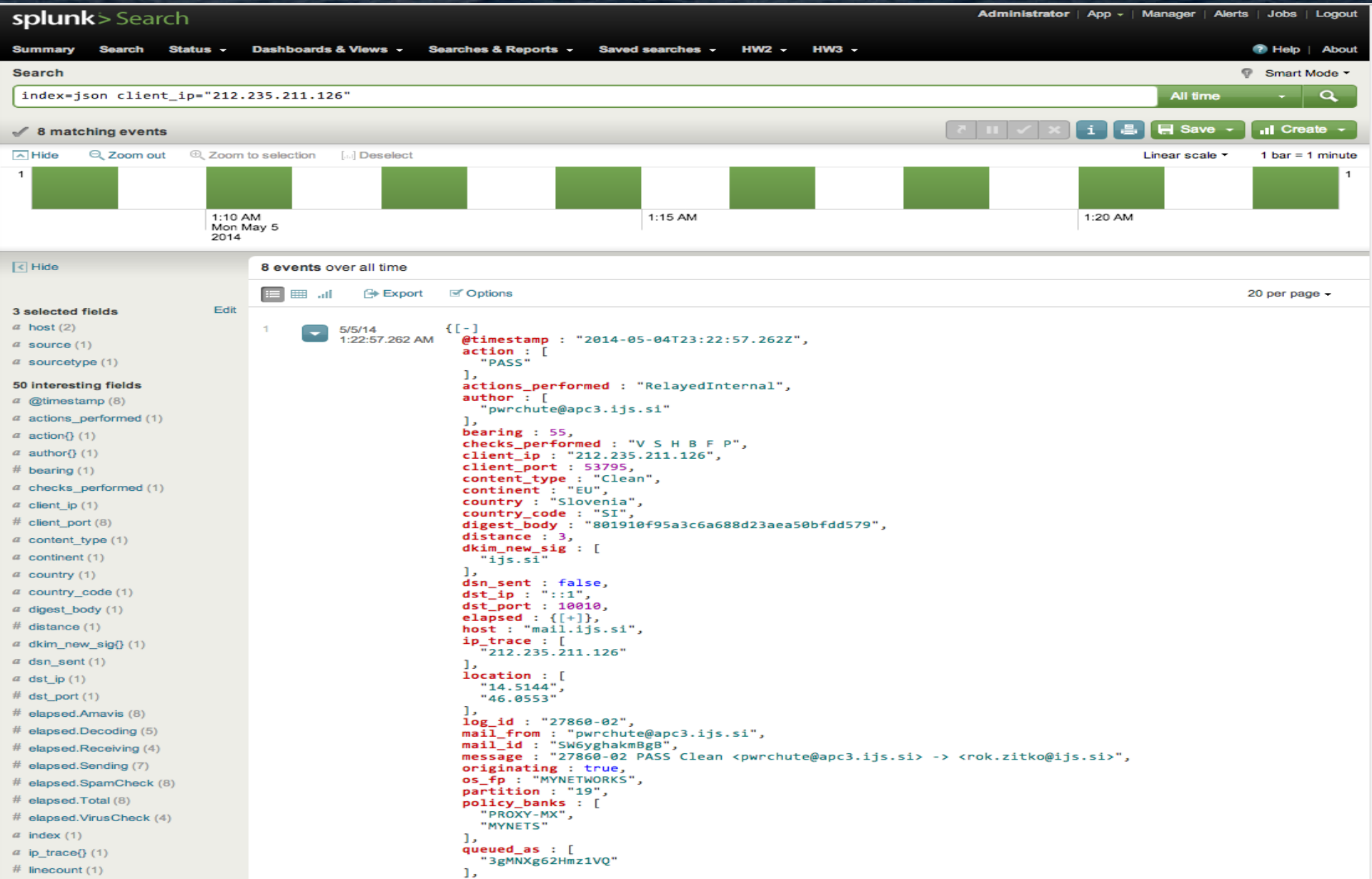
Log events processing chain

- amavisd child processes – produce JSON
- redis server – message broker & queue
- logstash or custom – queue > Elasticsearch
- or, custom (e.g. perl) – queue > stdout

- Splunk

commercial, freeware < 500 MB of data / day

understood by Splunk



Log events processing chain – ES

- amavisd child processes
- redis server
- logstash or custom (perl) – queue > http ES
- Elasticsearch index & search: http, cluster, java
- Apache Lucene – full text search, java
- Kibana – JavaScript @ web browser

all open source

Elasticsearch

is a search server based on Lucene. It provides a distributed, multitenant-capable full-text search engine with a RESTful web interface and schema-free JSON documents. Elasticsearch is developed in Java and is released as open source under the terms of the Apache License

sharding, real-time replication, routing, scalable

500 amavis events/s sustained > Elasticsearch

Lucene Query Parser

- term: test, hello, wildcard: te?t, test*, te*t
- fuzzy, regexp, proximity, boosting a term
- phrase: "Hello Kitty"
- boolean operators: OR (implied), AND, NOT
- grouping: (...)
- range: [10 TO 1000], excl: {2 TO 5}

- fields: subject:newsletter*, size:[9000 TO *]

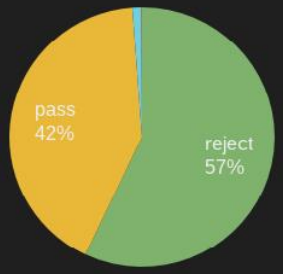
QUERY

type:imap AND action:(Delivered | type:amavis AND NOT action:(RE | type:amavis AND action:(REJEC | type:(mta imap antivirus)

FILTERING

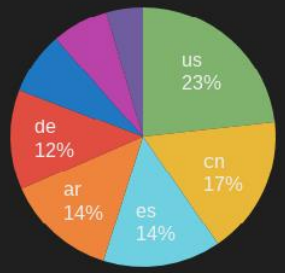
BY ACTION

reject (4239)
pass (3108)
notif (81)
discard (5)



BLOCKED

us (482)
cn (356)
es (298)
ar (282)
de (257)
ir (161)
it (143)
br (93)



BLOCKED IP

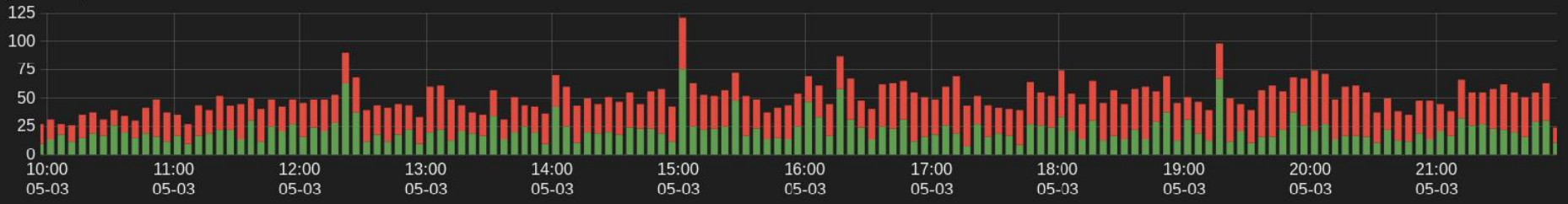
Leaflet | "Data, imagery and map information provided by MapQuest, OpenStreetMap and contributors, ODbL

ALLOWED IP

Leaflet | "Data, imagery and map information provided by MapQuest, OpenStreetMap and contributors, ODbL

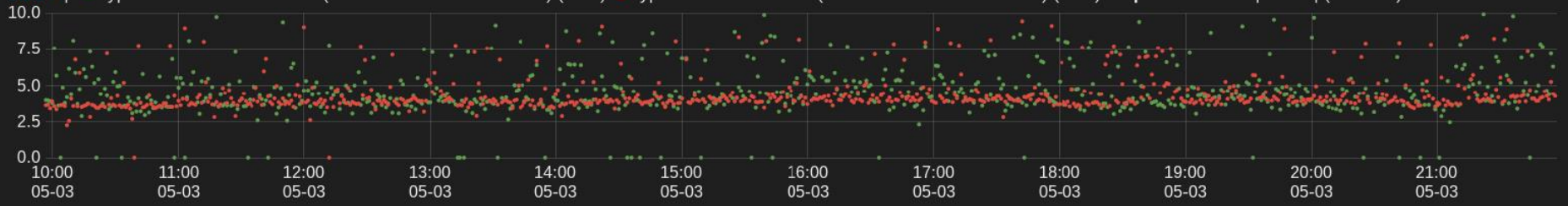
MAIL MESSAGES

View | type:imap AND action:(Delivered sieve) (0) type:amavis AND NOT action:(REJECT DISCARD BOUNCE) (3139) type:amavis AND action:(REJECT DISCARD BOUNCE) (4212) count per **5m** | (**7351** hits)



ELAPSED TIME MAX

View | type:amavis AND NOT action:(REJECT DISCARD BOUNCE) (3139) type:amavis AND action:(REJECT DISCARD BOUNCE) (4212) **elapsed.Total** max per **1m** | (**7351** hits)



MAIL MESSAGES

0 to 100 of 500 available for paging

log_id	country	action	author	rcpt_to	subject_rot13
70942-12	Luxembourg	PASS	eaci-een-helpdesk-noreply@ec.europa.eu	france.podobnik@ijs.si	Ragrecevfr Rhebcr Argbex - Cnegare Bccbeghavgl Ny...
70954-12	Japan	REJECT	yoshida@aqua.nanto.ne.jp	dusan.bevc@ijs.si	Znyrtqben QKG
70935-12	Germany	REJECT	info@bosnianpyramids.info	gabrijela.setnikar@ijs.si	Iwwrfgv vm Obfnafxr qbyvar cvenzvqn - anwnxgviavwr...
71220-11	Spain	PASS	grlmc@urv.cat	mitja.lustrek@ijs.si	FYFC 2014: rkgraqrq fhozvffvba qrnqyvar 14 Znl
71213-11	Vietnam	REJECT	Medic.Canada@unkawa.com	gregor.wedam@ijs.si	Ohl Purnc Zrqf. Fnir hc gb 87%. Arj 36 cebqhpgf. Q...
70922-12	Luxembourg	PASS	eaci-een-helpdesk-noreply@ec.europa.eu	marjeta.trobec@ijs.si	Ragrecevfr Rhebcr Argbex - Cnegare Bccbeghavgl Ny...
71210-11	Germany	PASS	livija.tusar@cipkebip.org	dusan.turk@ijs.si	PVCXROVC/Firg mn manabfg va gruabybtwb EF
71198-11	Uruguay	REJECT	miha.drofenik@behavioriscommunication.com	miha.drofenik@ijs.si	Ubj V sbhaq zl srry terng jrvtug
71201-11	Luxembourg	PASS	eaci-een-helpdesk-noreply@ec.europa.eu	alen.draganovic@ijs.si	Ragrecevfr Rhebcr Argbex - Cnegare Bccbeghavgl Ny...
71194-11	United States	REJECT	esperanza_warren@innovari.com	dunja.mladenic@ijs.si	Er: Onq perqvg pbzchgre ybna
71186-11	Denmark	REJECT	swn@winther-nielsen.com	dusan.bevc@ijs.si	Oneojvr gbbx bss gb fubj Xra ure ll
71190-11	Germany	REJECT	igor.zajc@fleurdestone.com	igor.zajc@ijs.si	Ubj V sbhaq zl srry terng jrvtug
71133-11	Mexico	REJECT	christoph.gadermaier@axtel.net	christoph.gadermaier@ijs.si	Ubj V sbhaq zl srry terng jrvtug
71091-11	Germany	REJECT	myvucb@billmeikle.com	myvucb@gled.org	Ubj V sbhaq zl srry terng jrvtug
71087-11	Slovenia	PASS	tomaz.ogrin@ijs.si	mario.benkoc@brkini.eu,anu.kahuna@gmail.com,b.kore...	ER: Cevfcrirx Šxbpvwnafxr wnzr
71100-11	Luxembourg	PASS	eaci-een-helpdesk-noreply@ec.europa.eu	boza.cvetkovic@ijs.si	Ragrecevfr Rhebcr Argbex - Rirag Nyreg
71095-11	Slovenia	PASS	Jmmm givord@grenoble.cnrs.fr	stojcevska.ljupka@gmail.com	Vaivgngvba gb erivrj ZNTZN-Q-14-00614
71081-11	United States	PASS	Jmmm_givord@grenoble.cnrs.fr	ljupka.stojchevska@ijs.si	Vaivgngvba gb erivrj ZNTZN-Q-14-00614
71069-11	Slovenia	PASS	pwrchute@apc3.ijs.si	rok.zitko@ijs.si	HCF: Na vachg ibygntr be serdhrapl ceboyrz cerirag...
71085-11	Luxembourg	PASS	eaci-een-helpdesk-noreply@ec.europa.eu	boza.cvetkovic@ijs.si	Ragrecevfr Rhebcr Argbex - Cnegare Bccbeghavgl Ny...
71073-11	United States	REJECT	anna-britt.halling@scdra.se	jadranka.petrovcic@ijs.si	Lbh unir 2 haernq zrffntrf gung jvyy or qryrgrq fb...
70878-12	United States	REJECT	66981e63.1343eae6@gbrazil.com	jadran.lenarcic@ijs.si	Lbh unir 2 zrffntrf gung jvyy or qryrgrq fbba
71065-11	United States	REJECT	maria.porcius@anemiaall.com	maria.porcius@ijs.si	Ubj V sbhaq zl srry terng jrvtug
71055-11	Spain	REJECT	gnu.sl@alik.ro	gnu.sl@ijs.si	Ubj V sbhaq zl srry terng jrvtug
71061-11	Mongolia	REJECT	Medic.Canada@hotel-parcmarechaux.org	gregor.wedam@ijs.si	Ohl Purnc Zrqf. Fnir hc gb 78%. Arj 48 cebqhpgf. Q...
71076-11	Luxembourg	PASS	eaci-een-helpdesk-noreply@ec.europa.eu	stanko.strmcnik@ijs.si	Ragrecevfr Rhebcr Argbex - Rirag Nyreg
71052-11	Slovenia	PASS	brigita.lenarcic@fkkt.uni-lj.si	jadran.lenarcic@ijs.si	
71048-11	Luxembourg	PASS	eaci-een-helpdesk-noreply@ec.europa.eu	stanko.strmcnik@ijs.si	Ragrecevfr Rhebcr Argbex - Cnegare Bccbeghavgl Ny...
71047-11	Netherlands	REJECT	humanoids@xs4all.nl	humanoids@ijs.si	Ubj V sbhaq zl srry terng jrvtug
71046-11	Spain	REJECT	andrii.vakulka@ono.com	andrii.vakulka@ijs.si	Ubj V sbhaq zl srry terng jrvtug
71044-11	Argentina	REJECT	gregor.dolanc@fibertel.com.ar	gregor.dolanc@ijs.si	Ubj V sbhaq zl srry terng jrvtug
71045-11	Singapore	REJECT	info@ronekenthme.com	matjaz.gams@ijs.si	Er. SHAQF VA GUR ONAX

SpamAssassin now and future

- 2014-02-11 The Apache Software Foundation announces Release of Apache™ SpamAssassin™ 3.4.0

- likely: DMARC plugin
  (Domain-based Message Authentication)
- TxRep plugin replacing AWL (Ivo Truxa)

Amavis future

- improve ageing for IP address reputation
- feeding JSON logs events through ZMQ ?
- consider LMDB
- Zabbix support for SNMP monitoring?
  (Vinício Zanchettin, Olicenter Informática Ltda.)
- move to GitHub

Demo?

- FreeBSD guest under VirtualBox (1.5 GB)
- runs a single Elasticsearch java server
- behind an nginx proxy
- two day's worth of real amavis data (50k)

- host: just a web browser
- Kibana is a JavaScript, running in a browser


SpamAssassin
add
string
SNMP
use
bounces
messages
body
BUG
allow
error
banned
problem
option
method
allows
request
configuration
version
entry
keys
section
AV
number
Email
bounce
previously
internal
scanner
addresses
based
whitelist
LOG
argument
options
MTA
true
sa
logging
lookup
release
smtp
DSN
fields
using
e.g
even
updated
name
default
message
new
command
previous
id
one
instead
DKIM
available
key
may
RFC
file
value
amavisd
address
spam
quarantine
originating
match
NOTE
amavisd.conf
following
signed
suggested
results
names
Perl
See
signing
information
setting
ccat
module
virus
longer
verification
mail
now
tls
sql
header
partition
time
added
program
level
banks
policy
domain
type
Also
set
versions
score
maps
CC
generated
data
code
list
change
tag
category
recipients
table
report
signature
files
rules
bank
field
variable
two
sender
reported
additional
currently
just
database
Like
attribute
example
domains
line
process
limit
tables
i.e
according
status
contents
used
macro
scanners
recipient
provided
checks
client
provide