Haraka

A Full Featured SMTP Server written in Node.js

October 2012

What is Haraka?

- An Anti-Spam inbound SMTP server
- High Performance
- High Scalability
- A flexible outbound SMTP server

What is SMTP?

- SMTP is the protocol with which:
  - EMail is delivered from a Mail Client to a Server
  - Servers deliver Email between each other
- SMTP is not:
  - How you get your email from a server - that's generally IMAP (or POP3)

Haraka Alternatives

- Postfix
- Exim
- Qmail
- Exchange
  - Exchange does a LOT more though

What does Haraka NOT do?

- Haraka does not implement user delivery
  - Delivering mail to a user on a server
    - Often via procmail, or sending to an IMAP server
- We recommend keeping Postfix, Exim etc around for that
- It's a HARD problem, rife with security problems - something I don't want to get wrong
- Patches welcome!

Haraka Users

- SendtoDropbox
  - Extracts attachments from emails and uploads to dropbox
- FSL.com
  - Replacing their entire sendmail based anti-spam systems with Haraka
- Craigslist - 10 million emails/day
  - Reduced mail server hardware from 30+ servers to 12
  - All incoming mail to craigslist users goes via Haraka
  - Working on outbound
  - Also working on fully anonymous remailing.

Haraka Performance

```
matt@Valour ~/Perl/node$ time /usr/libexec/postfix/smtp-source -l 5000 -m 50000 -s 100 -d -f matt@local -t
matt@haraka.local -c localhost:2525
50000

real 1m28.451s
user 0m1.789s
sys  0m6.357s

matt@Valour ~/Perl/node$ time /usr/libexec/postfix/smtp-source -l 5000 -m 50000 -s 100 -d -f matt@local -t
matt@haraka.local -c localhost:2525
50000

real 0m12.020s
user 0m1.730s
sys  0m6.351s
```

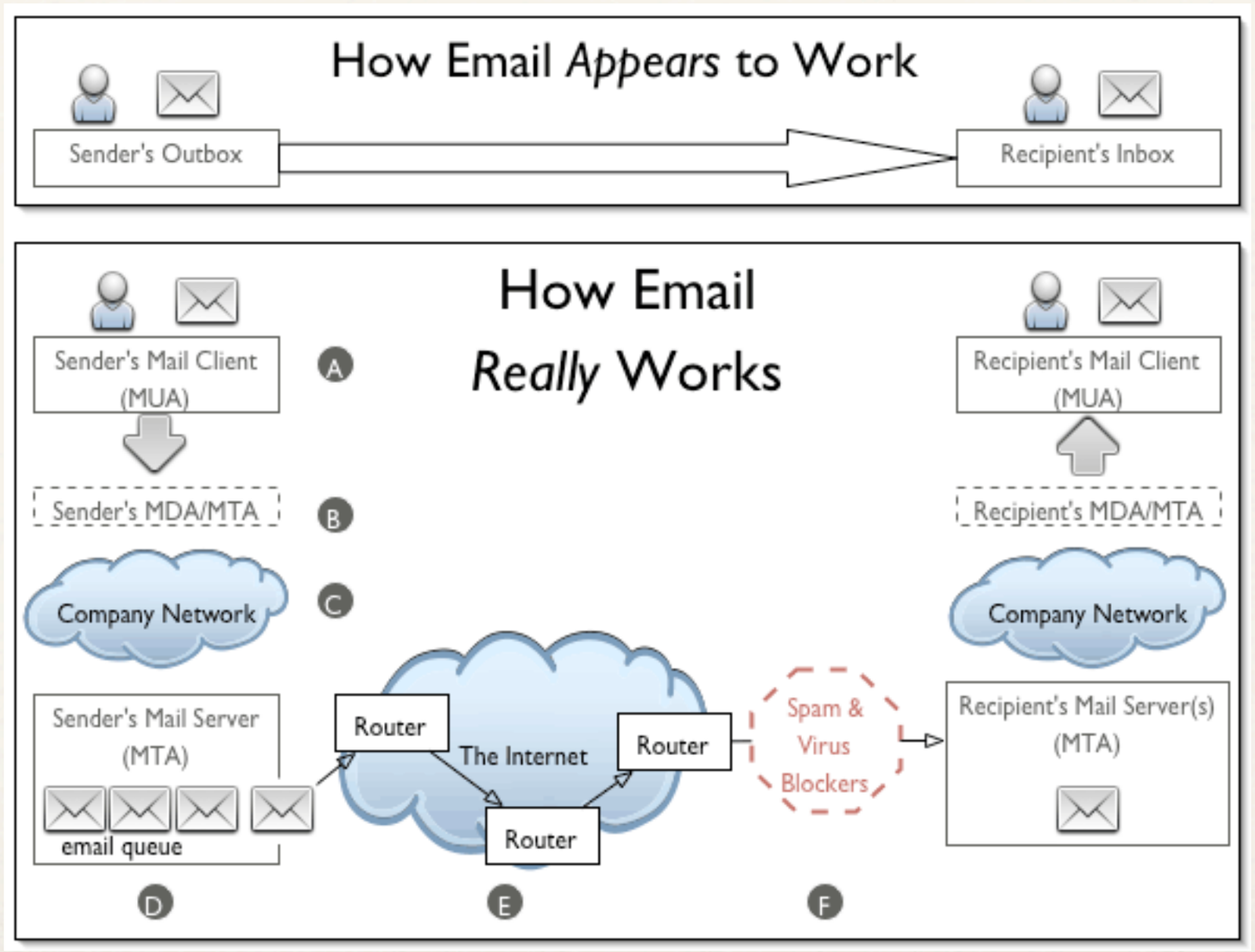
Why is it so fast?

- Node.js is just very fast
- Watch “Vyacheslav Egorov - One day of life in V8” on http://2012.jsconf.us/
- Async I/O makes a big difference
- MessageLabs spamtraps example

What is your CPU doing?

- Core: ~1 instruction per cycle
- L1 cache: 3 cycles
- L2 cache: 14 cycles
- RAM: ~250 cycles
- Disk seek: 40 million cycles
- Network: 240 million cycles
- Source: http://duartes.org/gustavo/blog/post/what-your-computer-does-while-you-wait

How Does Email Work?



Email Basics

- So you send email to “someone@haraka-is-awesome.com”
- How does that work?

Email Basics

- Email client connects to its outbound mail server
- It authenticates with a username/password (usually)
- It gives it the mail
- The server responds with “OK, thank you”
- But the mail isn’t really sent. Yet.

Email Basics

- The server queues the mail securely - if it fails to queue in ANY way whatsoever it MUST send a failure message.
- Most people think email is unreliable...
  - It is probably the most reliable protocol on the planet
  - Poorly implemented Anti-Spam measures make it unreliable

Email Basics

- Server looks up MX records in DNS for haraka-is-awesome.com
- If no MX records, looks up A records for haraka-is-awesome.com
- Attempt to deliver...
- 4xx = try again later
- 5xx = fail. Deliver bounce message back to user

Email Basics

- Setting up a mail server (outbound)
- Static IP address
- PTR record stating your real hostname
- EC2 is OK but you need an Elastic IP and PTR request (3 days), but beware delivery may be flaky at best

Haraka Basics

- First we install Haraka:
  - sudo npm install -g Haraka
- Now create a simple instance:
  - haraka -i /tmp/haraka_demo
- Check out what it installed via “ls -l /tmp/haraka_demo”

Haraka Plugins

- Haraka is entirely Plugin based
- All functionality, beyond core SMTP functionality, comes from plugins
- Core plugins provide many features - see `haraka -l` for a list
- Custom plugins go in “/tmp/haraka_demo/plugins”

Keys to the Castle

- Edit “config/me” to your hostname (should be set on install)

- Edit “config/loglevel” to “LOGPROTOCOL” to log everything down to the protocol level

- Edit “config/smtp.ini” to set the port to listen on and other “core” settings

- Run: haraka -c /tmp/haraka_demo

Plugins are key...

- Now stop the server (ctrl-C) and edit “config/plugins”
- Default plugin list works OK, but most people customize
- Again, get a list of plugins with `haraka -l`

How config/plugins works

- Plugins run in an order that only makes sense if you understand SMTP
- Plugins “hook” into SMTP processing chain
- e.g. hook_rcpt hooks into the SMTP “RCPT TO:” command
- Order of plugins in config/plugins affects the order each hook is called in

Default Plugins

```
# default list of plugins

# Log to syslog (disabled by default, see docs)
# log.syslog

# block mails from known bad hosts (see config/dnsbl.zones for the DNS zones queried)
dnsbl

# Check mail headers are valid
data.rfc5322_header_checks

# block mail from some known bad HELOs - see config/helo.checks.ini for configuration
helo.checks

# control which "MAIL FROM" addresses you accept. See docs.
mail_from.access

# Only accept mail where the MAIL FROM domain is resolvable to an MX record
mail_from.is_resolvable

# Disconnect client if they spew bad SMTP commands at us
max_unrecognized_commands

# control which "RCPT TO" addresses you reject. See docs.
rcpt_to.access

# Only accept mail for your personal list of hosts. Edit config/host_list
# NOTE: THIS IS REQUIRED for inbound email.
rcpt_to.in_host_list

# Queue mail via smtp - see config/smtp_forward.ini for where your mail goes
queue/smtp_forward
```

Reading about a plugin

```
$ haraka -h data.rfc5322_header_checks
data.rfc5322_header_checks
==========================

This plugin enforces RFC 5322 Section 3.6 which states that:

All messages MUST have a 'Date' and 'From' header and a message may not contain
more than one 'Date', 'From', 'Sender', 'Reply-To', 'To', 'Cc', 'Bcc',
'Message-Id', 'In-Reply-To', 'References' or 'Subject' header.

Any message that does not meet these requirements will be rejected.
```

Stopping Spam

- data.uribl - URL based blocklisting
- dnsbl - Connecting IP address based blocklisting
- early_talker - Stops poorly written spam bots
- spamassassin - Send emails to SpamAssassin for evaluation

Writing a Plugin

- Let's add support for disposable email addresses
- We sign up for a web site, but we don't want to hear from them after a certain date
- Give them an email address of myname-20121015@domain.com
- Haraka will automatically reject emails which arrive after that date
- Emails arriving within the expiry date will get re-written and delivered to myname@domain.com

Creating the plugin

- We already have our Haraka installation in /tmp/haraka_demo
- Create our plugin:
  - `$ haraka -c /tmp/haraka_demo -p rcpt_to.disposable`
- This creates a `plugins/rcpt_to.disposable.js` implementation file and a `docs/plugins/rcpt_to.disposable.md` documentation file (in Markdown)
- Now edit `config/plugins` to add the new plugin to your Haraka instance

```
exports.hook_rcpt = function (next, connection, params) {
    var rcpt = params[0];
    this.loginfo("Got recipient: " + rcpt);

    // Check user matches regex 'user-YYYYMMDD':
    var match = /^(.*)-(\d{4})(\d{2})(\d{2})$/.exec(rcpt.user);
    if (!match) {
        return next();
    }

    // get date - note Date constructor takes month-1 (i.e. Dec == 11).
    var expiry_date = new Date(match[2], match[3]-1, match[4]);

    this.loginfo("Email expires on: " + expiry_date);

    var today = new Date();

    if (expiry_date < today) {
        // If we get here, the email address has expired
        return next(DENY, "Expired email address");
    }

    // now get rid of the extension:
    rcpt.user = match[1];
    this.loginfo("Email address now: " + rcpt);

    next();
}
```

So what's going on there?

- exports.hook_rcpt - hook into the RCPT TO command
- next, connection, params - all hooks receive these
- We call next() when we are finished
- The email address (RCPT) is in params[0] and is an Address object
- If we want to reject the email we call next(DENY, msg)
- We can also tell the remote end to try again another time with next(DENYSOFT, msg)

Haraka For Outbound

- Most people don't need an outbound mail server
- Use gmail for personal, or Sendgrid/Mailchimp etc for business

- But Haraka can provide useful hooks for outbound mail in some situations
  - Bannering
  - Custom MX lookup routines
  - Bounce management
- If you must do it, read: http://mailchimp.com/resources/guides/email-delivery-for-it-professionals/

Sending Email via HTTP

- Mail sending libraries in some languages are fragile at best

- https://gist.github.com/1924944 - a plugin to run “express” inside Haraka and send emails based on templates

- Good for internal alerting, or just a free version of MailChimp

  - beware of deliverability issues

  - MailChimp/Sendgrid handle all that for you

Outbound Mail Performance

- On a quad core (8 hyperthread) Mac I was able to ramp up to around 800 emails/second
- That's 69 million emails/day
- Admittedly that was only localhost delivery - network issues change this behavior drastically

Scaling and System Management

- Native multi-process support using node's built-in "cluster" module
- Daemonize via the daemonize plugin
- process_title plugin shows what's going on:

```
28841 ?        Ssl    0:21 Haraka (master) cn=77 cc=1 cps=0/0.09/2
28843 ?        Sl     0:48  \_ Haraka (worker) cn=35 cc=0 cps=0/0.04/4
28845 ?        Sl     0:48  \_ Haraka (worker) cn=42 cc=1 cps=0/0.05/6
```

Cluster Management

- Some plugins support communication across a cluster via Redis
- The rate_limit plugin will limit connection rates from different hosts and balance it cleanly across an entire cluster of machines running Haraka
- Similar technique being worked on for outbound, soon

Thank You

- Read More:
  - http://haraka.github.com/ - main site
  - https://github.com/baudehlo/haraka - git repository