;login:

THE MAGAZINE OF USENIX & SAGE

April 2003 • volume 28 • number 2

inside:

SYSADMIN

Chalup: Not on My Watch!

USENIX & SAGE

The Advanced Computing Systems Association &
The System Administrators Guild

not on my watch!

Filtering Options Revisited

Over the past several years, we have all seen "Unsolicited Commercial Email," a.k.a. "spam," grow from an annoyance primarily propagated through Netnews to something that routinely lands in everyone's mailbox. The evolutionary path followed by anti-spam measures somewhat resembles that of network security. Remember when firewalls were (allegedly) optional?

by Strata R. Chalup

President, VirtualNet. Starting as a Unisys 68K admin in 1983, Strata Chalup is now an IT project manager but allegedly has retained human qualities. Her mixed home network (Linux, Solaris, Windows) provides endless opportunties to stay current with hands-on tech.

strata@virtual.net

Attempts to stem the rising tide of spam have had humorous consequences, at least to observers, if not participants. Some articles from our friends across the water point out the ruckus caused by unintended consequences of anti-spam software, ranging from stifling discussions on certain bills in the UK Parliament to rejecting internationalized messages as "inappropriate content." I realize that Welsh isn't for everyone, but, really, that's a bit extreme.

The first-ever Spam Conference recently concluded at MIT and brought together both cutting-edge research and authors of popular freeware and commercial packages. Much good work continues to come out of the conference, and I look forward to seeing the next set of results. Of course, the "Spam Conference" was really about anti-spam methods and the problem of spam, but that's just the way conferences are named.[1]

> "Yes, the danger must be growing,
> For the rowers keep on rowing,
> And they're certainly not showing
> Any signs that they are slowing!"
> — Willy Wonka

> "Analysts believe inbound spam email for the corporation is
> at least 30% now and will grow to 50% in the next two years."
> — Gartner, 2002

"Not on *my* watch!" — everyone to whom I've quoted the above

Tutorials like those provided by ServerWatch can compare and contrast commercial systems, but we've chosen to focus largely on the freeware systems here. Obviously, the best protection is not to get yourself on the lists in the first place, but, as Arlo Guthrie said, "This is not a song about Alice." It is worth mentioning that the spam/anti-spam arms race shows no signs of slowing down. Techniques such as obfuscation with hedge characters or HTML symbol encoding now offer spam-harvesting webbots without the slightest hiccup. Embedding mailto links or email addresses in a protective bezoar of JavaScript is good protection now but probably the next bit of digestive evolution for the e-bots. Like server-side generation of text images, this is also an accessibility issue, foiling conventional text-to-speech systems as well as address-harvesting 'bots.

As always, things will get worse before they get better. A recent MessageLabs report shows that as filtering options improve, spammers (and virus writers!) are increasingly targeting loopholes in our mail clients and mail-handling procedures. For example, what if you get an attachment called *our-new-house.jpg.exe.jpg*?

To quote from the report, "The malware relies on especially crafted email headers, creating an attachment with three file-extensions. . . . The first extension . . . is visible to

1. I still haven't seen a more unfortunate name than that posted on a call for papers at the MIT Psychology Department, where I once managed systems: "(n)th Annual International Invitational Traumatic Head Injury Conference." Ouch!

The issues of anti-spam and anti-virus are increasingly converging.

the email user, and is intended to persuade them that the attachment is "safe." The final extension . . . is used by Outlook Express to set the icon to represent the application for opening the attachment. . . . However, the unusual middle extension (.EXE) is used by Outlook Express to determine how to launch the attachment; therefore an .EXE file will be executed if a user double clicks on an infected attachment." The next generation of spam harvesting tools will probably include viruses which gather spam directly from people's address books, so the issues of anti-spam and anti-virus are increasingly converging.

The more things change, the more they stay the same. Email anti-spam technology is recapitulating the ontogeny of Usenet anti-spam technology. Aren't we overdue for the Breidbart Index Filtering on ISP mail gateways and email security products? (*http://www.stopspam.org/usenet/mmf/breidbart.html*). Blacklists have been around forever, and whitelists are gaining in popularity. Two years ago there were only one or two freely available quasi-automated whitelisting systems, while now a double handful can be found, and even a company or two staking its future on a special type of whitelisting. Sophisticated pattern matching is being augmented by even more sophisticated heuristic-based Bayesian modeling. Service providers are even attempting to require authentication and/or control of accessible servers to try to stop spam at its source. Let's take a look!

Follow the White(list) Rabbit

One article mentioned that a favorite trick of randomizing spammers is to twiddle with the comments in HTML-formatted spam. The message looks identical to the unlucky recipient, but generates a different checksum. The author's response was that "No one I care to talk to sends mail as HTML" and that his practice is to "direct HTML mail to my spambox."

We should all be so lucky! The reality is that shunting HTML-formatted mail to a spam box only works tolerably if accompanied by aggressive whitelisting of friends, family, and coworkers. The primary disadvantage of whitelisting, of course, is the onus on you, the recipient, to keep the whitelists updated as people change their addresses, send mail from other accounts while traveling, and the like. Fortunately, there are a plethora of options from which to choose, many of which are listed in this article's listing of links.

Taking the concept of whitelisting to perhaps its most extreme level is the Habeas system. This unusual system rests on modern patent and trademark law and will be truly useful only when large numbers of persons start using it. As you might thus expect, it is currently free to individuals and service providers. Commercial entities must pay a licensing fee but, more importantly, jump through some well-defined hoops. Habeas has copyrighted a specific haiku, and it has a patent pending for their use of "protected" text, called a Warrant Mark, in message headers to provide authentication. It is unclear from their Web site if the patent includes their specific blacklist of noncompliant entities.

To be a Habeas-compliant entity, one must only send messages containing the special text headers to recipients who have truly opted in to receiving the message. Spammers who use the Warrant Mark in their mails are liable for prosecution under good old-fashioned copyright and patent law. Habeas claims to have created a structure in which a traditional legal framework is sufficient for prosecution, with no reliance on newfangled and often confusing cyberlaws. If widely adopted, the system would provide a

combination of guaranteed marking of non-spam mail and a way to go after spammers who abuse the Warrant Mark.

Why is this supposedly better than generic whitelisting? The company's FAQ reminds people that whitelists cannot detect spammers forging popular "From" addresses, such as notification addresses from retailers. To be scrupulously fair, a forged set of headers containing the Habeas Warrant Mark would also not be detected, unless sent by a repeat offender already blacklisted. However, you may feel better about viewing it, given that your complaint (to Habeas) will actually cause something to happen, namely blacklisting and an aggressive legal pursuit of the spammers for infringement.

Down a Different Rabbit Hole: RFC 2476

Service providers, and an increasing number of corporations, are requiring authentication to internal mail servers and blocking access to port 25 of external servers. Together these steps can certainly reduce the amount of spam generated at a typical huge ISP, but they can also really cramp your style if you are traveling and would like to preserve your email independence. "Hmm," you say, "sounds like it's time to find another port." Exactly so, but as an Upstanding Net Citizen you worry about sending mail to *Adam.West@WayneManor.org* through a port other than the well-known service port for SMTP. Holy protocol, Batman! Enter RFC 2476 to the rescue!

The issue is not really one of which port to use, although the RFC 2476 does define port 587 as the WKS port for message submission. The primary focus of the RFC is to distinguish between *message transport*, in which an MTA must not meddle with certain aspects of the message, and *message submission*, where it may be useful or needful to alter or add to a message. The first two reasons given in the RFC are extremely germane to this discussion, namely:

- Implement security policies and guard against unauthorized mail relaying or injection of unsolicited bulk mail
- Implement authenticated submission, including off-site submission by authorized users such as travelers

In his excellent series of articles, "RFCs for the Rest of Us," Paul Boutin discusses RFC 2476 in detail, along with RFC 2554 (SMTP Authentication) and RFC 2505 (Anti-Spam Recommendations).

Communities and Checksums

Vipul's Razor (v2) is a checksum-based method of tagging messages as potential spam. Netnews administrators may recognize this methodology from various NNTP filtering systems. Razor has the familiar advantages of digest or checksum-based approaches over pattern-matching rule-based systems, most notably lower computational overhead and small data sets. Of course, there is a glaring disadvantage – that randomizing a small part of the message body will change the checksum and let spam sneak in the door.

A complex mesh network of hosts is aggregated under a DNS zone used by Razor Agents to find Razor Discovery Servers. The Razor Agents query Razor Discovery Servers to find the Razor Catalogue Servers (for razor-check(1)) and Razor Nomination Servers. The default is *razor.cloudmark.com*, but the appearance of a GPL'd version of Razor called "Pyzor" and a separate initiative called the Distributed Checksum Clearinghouse (DCC) now gives ET somewhere else to phone home. In practice,

SpamNet probably represents the largest user community, and it is reporting solely to Cloudmark. Spam Assassin would clearly be next in line, and while it offers reporting checksums to all three services, it's not clear how widely this has been adopted. For the curious, there is a description of the Razor reporting protocol at *http://www.stearns.org/razor-caching-proxy/razor2-protocol*.

Cloudmark's SpamNet is one of those "good news, bad news, good news" deals. The good news is that it's free. The bad news, for many of us, is that the only currently supported client is Microsoft Outlook 2000/XP/2002. But the good news beyond that is that SpamNet is essentially the pseudo-commercial arm of Vipul's Razor, as Vipul is one of Cloudmark's founders. Cloudmark is the primary aggregator of Razor/SpamNet data, and it's worth mentioning that a for-pay service, Cloudmark's Authority, leverages the data gathered by the free SpamNet community service.

Cloudmark has taken Razor's data-gathering one step further – using Bayesian classification, they turn the data into the somewhat loftily named "spamGenes" and "spamDNA." Their claim is that there are only 150 spamGenes and that their method consumes vastly fewer resources at the gateway. Let's see, there's "free," "v*g*r*a," "mrs mobuto sese seko," and, um, 147 more. One clue emerges from a *Wall Street Journal* article on Authority – namely, that the software concentrates on "the marketing message . . . it's how they make money and it doesn't change a lot." Neither does Authority; updates are made available every 30 to 60 days "in the form of spamDNA cartridges." Do those count as biohazards? Only if you're a spammer, I guess.

As does its predecessor, Razor v2, the SpamNet client preserves individual user privacy by generating a "fingerprint" or digest of a spam message and sharing only the fingerprints among SpamNet users. However, Cloudmark's Web site mentions the existence of a "Truth Evaluation System (TES)," which apparently rates each SpamNet user according to various factors, including volume, relevance, and accuracy. To quote the site, "Simply, long-time, trusted-user reports carry more weight in spam identification than new, untested reports. When a SpamNet member makes a good report, their trust rating is increased." If a user realizes that a spam that he or she marked as "Block" is actually a legitimate email, the user may "Unblock" it and get back their good SpamNet karma. This is the same mechanism employed by Razor.

"Heterodyne Portable Claw: Use Only for Good."

I have to put on my "virtual Peter Neumann hat" here and talk about some of the risks of systems like SpamNet. A virtuous privacy policy is no guarantee that one's data will not be used for marketing. It is only a guarantee that the current corporate structure will not use that data. I hope that Cloudmark has some stringent policies in place about whether their TES is a "corporate asset" or not. Currently, Cloudmark appears to be privately held, but I doubt that is their long-term strategy. If a less enlightened corporate entity were to obtain control of Cloudmark's assets through an acquisition, it would be very easy for them to build a truly impressive marketing database.

They could use existing ways of mining personal data from Web sites in the context of offering a SpamNet update or, perhaps, in the course of collecting the normal data from a SpamNet client. They might be able to cross-correlate with an existing marketing database such as DoubleClick or MSN. Since a reputation system is employed in the TES, each SpamNet client must have a unique identifier. User-reported spam fingerprints could be correlated with full-text spam, which in turn could be demographically sorted as targets via conventional marketing analysis. One could certainly create

an interesting reverse-engineered demographic database with Cloudmark's TES and a sufficiently large sample space of spam, such as the Ciphertrust spam archive.

Sound far-fetched? Companies such as DoubleClick make their living doing very similar analysis, based on Web and email cookies. I am not saying that one should not use SpamNet. I am saying that, in the spirit of RISKS-Digest, one should understand what the technology enables. When this scenario was described to a highly-placed source within Cloudmark, the danger was discounted as implausible. Then again, who believed five years ago that someday all your old radical college Usenet postings would be searchable, or that websites which you've never surfed before would greet you by name based on shared marketing profiles?

Bayesing at the Moon

You can't pick up an IT press article about anti-spam systems these days without encountering the buzzwords "Bayesian," "heuristics engine," or something similar. Hey, these were all around years ago on Usenet. So just how did we "forget" Bayesian filtering for so long? Paul Graham provides an excellent summary in his report to the MIT Spam Conference. Lack of acceptance of a "miss rate" of 92% with 1.16% false positives seems to have been the key factor. What Paul and others found is that the direct application of the Usenet technique ignored the message headers, which can arguably be said to be less meaningful in the NNTP context than in that of SMTP. When headers were factored in, the miss rate dropped to 99.5% with less than 0.03% false positives, applying the identical techniques previously used by Pantel and Lin.

Two secondary factors were the adaptive, or learning, capability of the filters, and the use of weighted tokens. The accuracy improves noticeably when the sample size is increased. Getting the accuracy rate that high involved putting a great deal more spam through the system, yielding impressive results. Additionally, by choosing the top 15 or so tokens to weight most heavily, the system can better deal with spams that, as Graham puts it, "tell you their life story" in the course of getting to the punch line.

A radically different approach to Bayesian filtering involving regular expression matching rather than tokenized input, the CRM114 system by Bill Yerazunis shows that we haven't even begun to run out of fire power to throw at the problem. The Controllable Regex Mutilator, to quote its home page, "offers sparse binary polynomial matching with a Bayesian Chain Rule. . . . Accuracy of the SBPH/BCR classifier has been seen in excess of 99 per cent, for 1/4 megabyte of learning text. In other words, CRM114 learns, and it learns fast." Yow!

The Swiss Army Knife Approach

The interestingly disjoint lists of server-side anti-spam tools at various Web sites suggest either an uninformed or highly opinionated general admin populace, or a highly insoluble problem that fits everybody like a bad pair of shoes. Are anti-spam software developers treading out their own Shoe Event Horizon?

One thing that many of these tools have in common is that they deploy mail through procmail, and then let loose with a whole arsenal of techniques. Filtering FAQs abound, but we won't reinvent the wheel, we'll just cite it in the references. Lately, even procmail substitutes are cropping up – if you're tired of procmail, try a substitute handler such as Salmon, which wraps some basic setup tasks along with the procmail functionality and an anti-spam engine.

LINKS

MPs discussions censored by protective filters:
http://www.theregister.co.uk/content/6/29175.html

Filter woes continue for MPs:
http://www.theregister.co.uk/content/6/29199.html

Triple (extension) threat:
http://www.messagelabs.com/viruseye/report.asp?id=130

ServerWatch Tutorial and Product Comparison:
http://www.serverwatch.com/tutorials/article.php/10825_1567361_2

O'Reilly article on spam harvest prevention:
http://www.macdevcenter.com/pub/a/mac/2002/11/01/spam.html

RFC 2476: Message Submission:
http://www.faqs.org/rfc/rfc2476.txt

Paul Boutin's "RFCs for the Rest of Us":
http://www.sendmail.net/rfcintro.shtml

maildrop:
http://www.flounder.net/~mrsam/maildrop/

Habeas "Sender Warranted Email":
http://www.habeas.com/faq/index.htm

Active Spam Killer (whitelist):
http://paganini.net/ask/

Tagged Message Delivery Agent (whitelist):
http://tmda.net/

Mail DeSpammer (reactive whitelist):
http://www.laas.fr/~felix/despam.html

The Infamous Big Brother Database
http://bbdb.sourceforge.net/
http://www.jwz.org/bbdb/

Filtering FAQ Fun for All and Sundry:
http://mip.ups-tlse.fr/~grundman/procmail/faq.html

Salmon (procmail++ ? YMMV):
http://is.rice.edu/~wymanm/smn/index.html

Spambouncer:
http://www.spambouncer.org/

2003 Spam Conference at MIT:
http://spamconference.org/

Bill Yerazunis' CRM114 system:
http://crm114.sourceforge.net/

Seriously technical goodies on filtering here:
http://www.paulgraham.com/bayeslinks.html

How Bayesian filtering evolved past Usenet:
http://www.paulgraham.com/better.html

Scads of clients for indiv & server ops:
http://email.about.com/cs/bayesianspamsw/

The Shoe Event Horizon
http://www.csua.berkeley.edu/~dxu/econ/shoe.html

But wait, there's more . . . :
http://dmoz.org/Computers/Software/Internet/Servers/Mail/AntiSpam/

junkfilter:
http://junkfilter.zer0.org/

Usenet anti-spam resources – all your old buddies like CleanFeed and SpamHippo and the like:
http://www.exit109.com/~jeremy/news/antispam.html

Cloudmark's SpamNet client (free):
http://www.cloudmark.com/products/spamnet/learnmore/spamnet.php

Cloudmark's Authority server software ($):
http://www.cloudmark.com/products/authority/

Web log article on Cloudmark, with screen shots:
http://www.emergic.org/archives/2003/01/10/

Girl Genius – Go Agatha!:
http://www.studiofoglio.com/girlgenius.html

Vipul's Razor:
http://razor.sourceforge.net/

Spam Assassin:
http://spamassassin.org/
http://spamassassin.taint.org/

The Outer Limits:
http://www.innermind.com/outerlimits/info/olopen.htm

Pyzor (Python, GPL version of Razor):
http://pyzor.sourceforge.net/

Distributed Checksum Clearinghouse:
http://www.rhyolite.com/anti-spam/dcc/

Email Sanitizer and Esd-l:
http://www.impsec.org/email-tools/procmail-security.html
http://www.spconnect.com/mailman/listinfo/esd-l

Brian Hatch's "Filtering Email with Postfix and Procmail" series (includes code examples). Parts 1 & 2 are Postfix-specific; 3 & 4 cover procmail and integration with various packages like Razor and Spam Assassin:
http://online.securityfocus.com/infocus/1593
http://online.securityfocus.com/infocus/1598
http://online.securityfocus.com/infocus/1606
http://online.securityfocus.com/infocus/1611

Email on SOHO Networks:
http://www.unixreview.com/documents/s=7460/uni1032893910897/ur0209o.htm

Ricochet Spam Handler:
http://vipul.net/ricochet/

A good example of this kind of technology is John Hardin's Email Sanitizer, as featured on the Email Security Discussion list (Esd-l). Introduced in 1999, it's a quiet example of a mature, refined, and ultra-configurable procmail rule suite that lets you pick the best of the best and apply it. Esd-l is also a good place to pick up breaking news about new attacks, such as the triple-threat extension trick mentioned at the beginning of this article.

One of the most popular tools, and one increasingly shipping quietly under the hood of many commercial anti-spam software suites and appliances, is Spam Assassin. It is truly the adaptive kitchen sink or Swiss Army knife of the cumulative filtering tools. Spam Assassin filters spam using a combination of traditional methods, including header and body checks, blacklists, and whitelists. On top of these metrics, it uses Vipul's Razor to score messages. Individual tests are weighted, as is the threshold at which the system decides "OK, this is spam."

Unlike the old *Outer Limits* TV show, you control the horizontal, you control the vertical, since weighting and threshold are user-adjustable. For instance, Spam Assassin now comes with a weighting for the Habeas Warranted Email service mentioned earlier; defaults are set to award a Habeas-compliant message a more beneficial status. Meanwhile, the Bayes system provided by the "sa-learn" facility keeps trying to predict what you consider spam vs. messages you want to see.

Spam Assassin's fans claim over 99% accuracy, but many first-time users report very different results. The key seems to be aggressive whitelisting, especially of mailing lists to which you have voluntarily subscribed. An unfortunate gap in the coverage results, since one source of spam for many of us is non-technical hobby or interest lists which may be spammed to reach subscribers. It might be worth experimenting with recursive calling of differently configured Spam Assassin instantiations, or combining Spam Assassin with some other program that will then sift your less well-behaved lists for secondhand spam.

But Wait, There's More!

A dizzying array of spam-prevention technologies exists to combat spam at the server level, for your home, office, or whole organization. Rather than attempt an exhaustive survey, I've included links to some of the more interesting ones. To save your cut-and-paste macros some work, we'll post the links on VirtualNet, so you can just bookmark-and-go. *http://www.virtual.net/Ref/resources.html* contains a bibliography of all my articles, and will be updated with this one.

One final note: To round out his contribution to spam fighting, the talented Vipul also wrote a spam tracer and handler called Ricochet to deal with spam that successfully runs the formidable gauntlet we've set up here. Enjoy!