Editor: **Michiel van Genuchten**
VitalHealth Software
genuchten@ieee.org

Editor: **Les Hatton**
Oakwood Computing Associates
lesh@oakcomp.co.uk

Delivering Genuine Emails in an Ocean of Spam

Leo Hatton and Alan John

This is our first column from the tiger economy of Singapore and our first example of software as a service for email. So, readers can enjoy yet another dimension of the ubiquitous impact of software. —*Michiel van Genuchten*

EMAIL IS ONE of IT's longer-standing success stories. Way back in 1971, Ray Tomlinson at Bolt, Beranek and Newman sent the first email across Arpanet and back to himself. In the 46 years since, it has become a true Internet standard. Its range is unlimited, and it's instant, free, and ubiquitous. Perhaps most important, it's open. No other global communications system can or could match email's many qualities.

Sadly, no other system can match email's potential for abuse, either. Its protocols were laid down many years ago when the world was a simpler, more trusting place. Today, with spam, scams, and phishing attacks of many inventive kinds constituting the vast majority of the hundreds of billions of emails sent every day, how then can we deliver the emails that matter?

The decades-old arms race between spammers and filtering systems shows no signs of slowing. In fact, increasingly sophisticated, and therefore more successful, phishing attacks are making headlines with greater frequency, suggesting that the battle might even be swinging in the attackers' favor. By 2016, mail accounts were sending about 400 billion spam emails a day, accounting for 86 percent of the world's email traffic.[1] This number might now be a little low. Figure 1 shows the junk mail load on SendForensics' email servers over the past year; this includes categorizing legitimate bulk mail as good.

The industry is being forced to respond by fundamentally changing its approach to the problem. It's no longer enough to rely solely on spam filter competence and user vigilance to separate the wheat from the chaff. The focus must shift from defense to prevention. In conventional spam filtering, accepting some spam (false negatives) as a price for not losing any genuine email (false positives) was the accepted policy. With more and more to lose from increasingly nefarious phishing attacks, this balance must be reassessed.

A valid question is what senders can do to assist the receivers' spam filters. Here,

0740-7459/17/$33.00 © 2017 IEEE

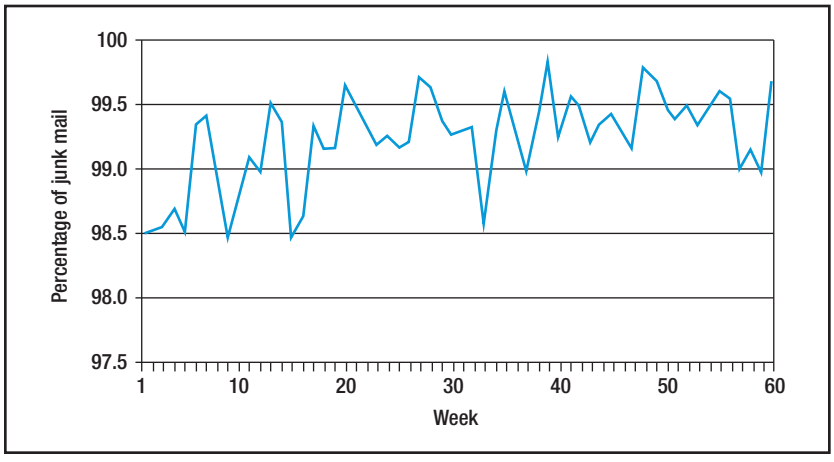

FIGURE 1. The percentage of junk mail received by SendForensics over 60 weeks. The spam battle might be swinging in the attackers' favor.

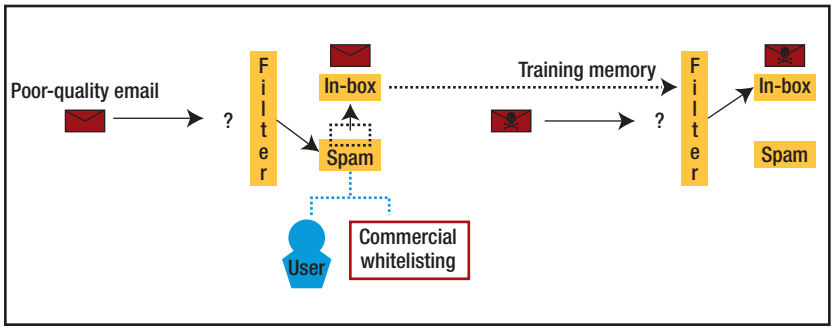

FIGURE 2. How whitelisting affects email filtering. If poor-quality emails are artificially whitelisted (by the user or through commercial whitelisting services), filtering systems will then have been trained to repress their warnings, allowing real attacks to sail through.

we present a system that lets senders analyze and optimize all outgoing email before sending it. This system aims to widen the gap between legitimate and illegitimate email in terms of the respective forensic footprints, ultimately making it far easier for existing and future filtering technologies to tell the difference.

A Question of Responsibility

In the fight against abuse, the email industry has already implemented a host of authentication protocols—for example, secure sending via Transport Layer Security (TLS), DomainKeys Identified Mail (DKIM), Sender Policy Framework (SPF), and, most recently, Domain-Based Message Authentication, Reporting and Conformance (DMARC). However, their adoption has so far been largely voluntary, with little penalty for noncompliance. In addition, with the increasing sophistication with which botnets are populated and deployed, the use of hijacked computers in corporate networks to send emails of apparently trustworthy origin can render some of these protocols largely irrelevant. In these cases, there's usually nothing but the content to give the receiving filtering system a fighting chance to identify the spoofed email.

So, a flawless sending infrastructure is the absolute minimum, with squeaky-clean content the ultimate goal. However, even with an automated analysis system, this puts a lot of responsibility on the sender. Luckily, consistently sending high-quality email provides huge benefits that aren't just related to security:

- protecting customers from phishing attacks,
- protecting staff from phishing attacks, and
- boosting engagement across email marketing channels.

Regarding the first two benefits, when users are used to receiving only the highest-quality legitimate email, the filtering systems protecting them will be far better calibrated to sniff out the subtle forensic differences during sophisticated phishing attacks. In contrast, if poor-quality emails are artificially whitelisted (by the user or through commercial whitelisting services), the systems will then have been trained to repress their warnings, allowing real attacks to sail through (see Figure 2).

Unfortunately in this day and age, unless a business has already suffered a major breach, the third benefit (boosting engagement) ends up being the primary motivator for taking greater responsibility for the email the business sends.

Deliverability

Deliverability is the industry term for an email's ability to reach a given in-box. If an organization sends high-quality emails that maintain a sizeable forensic distance between themselves and the hordes of spam,

more of them will pass the filtering inspections and end up in the customer's in-box. If more emails end up in more customers' in-boxes, then more are opened and clicked on (*engaged with*, in marketing speak). But this isn't just a desirable outcome for marketing-oriented emails. If you need to deliver an alert or a confirmation email to users, it's imperative that it lands in their in-box.

For example, suppose you're trying to send information on medications that are vital to your customers' health. Huge amounts of spam continually try to sell various dubious medicines to the public, and automated spam filters have become sensitive to them. So how do you convince these filters that your products are genuine and that you're sending your emails to genuine customers? Ideally, these emails should be constructed such that the forensic distance between them and their junk equivalents is easily identifiable.

SendForensics has turned deliverability into a quantifiable metric for individual emails. We compute this metric from a host of factors in an email. We then use these factors to inform senders, in conventionally understood language, on how to increase the forensic distance given the current state of the Internet and the attackers and spammers using it. This provides the opportunity for measurable optimization before the email leaves.

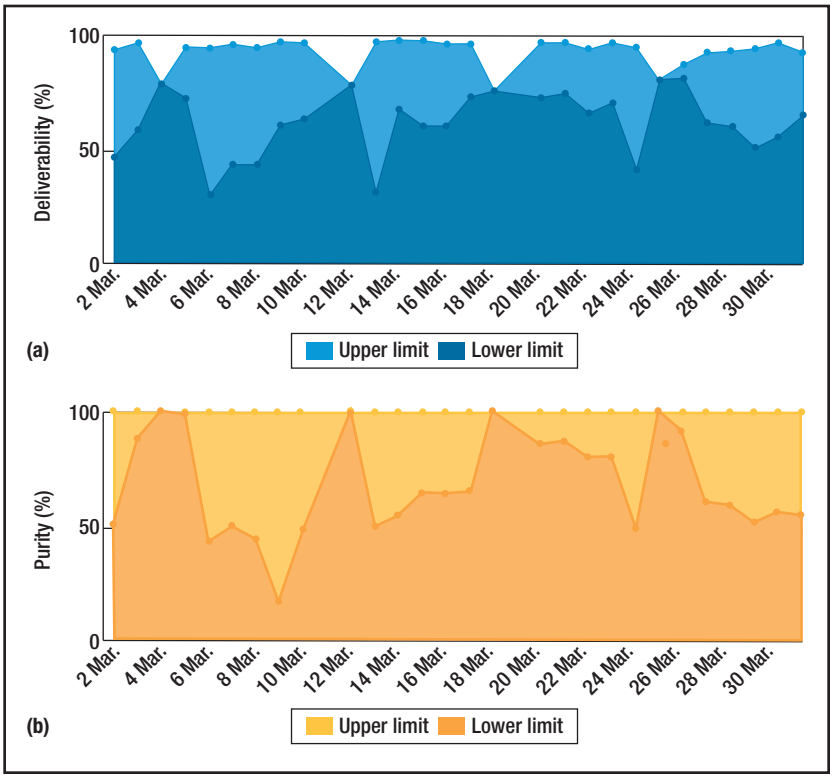

FIGURE 3. A comparison of (a) deliverability and (b) purity for a mail delivery network over the same period. Deliverability is email's ability to reach a given in-box. Purity measures an email's legitimacy. Both are given as a percentage, with the upper and lower limits showing the best- and worst-performing emails on a given day.

Purity

The deliverability metric alone is all well and good for increasing in-box placement, but it's not enough to determine the overall quality of an email's forensic signature. The latest generation of phishing attacks are engineered to be highly deliverable while still carrying the inevitable nefarious payload (often a single link). They can be sent from a hijacked machine (probably through a botnet) so as to benefit from a reputable sending infrastructure (no IP blacklisting, and the authentications check out), with a plausibly spoofed sending address. The content can be duplicated from legitimate mailings, with no obvious telltale spam markers (such as misspellings).

So, a second level of analysis is needed. This level employs *purity*, a predictive metric measuring an email's legitimacy using techniques similar to those for predicting deliverability. This metric indicates how legitimate the email actually is—not how legitimate it looks. As we've all seen, the best phishing emails are frighteningly legitimate-looking to even the most technically savvy user.

For example, Figure 3 shows the deliverability and purity metrics for a mail delivery network over the same period. Both correlations and anticorrelations are evident. In other words, reasonably deliverable mail can be toxic and therefore have low purity (as can happen through botnets—for example, around 10 March in Figure 3). Genuine high-purity mail can have depressed deliverability because it's poorly crafted (around 5 and 13 March in Figure 3).

Both analyses require a lot of computational resources. However, the cost is worth it because this approach

provides greater protection from email abuse for an organization's customers and staff and has the pleasant side-effect of higher engagement for its marketing programs.

The Software

The software to do this combines forensic algorithms and multivariate, multichannel, multilingual Bayesian statistical models built by continual analysis of large amounts of email over many years. Astonishingly, the Internet has a time constant of only a few minutes in assessing a particular email's deliverability. In other words, if the same email is sent just a few minutes later, a small percentage of the messages will produce different deliverability statistics. This is because of updates and other variations in spam-filtering systems, the ever-changing landscape of real-time block lists for abusing IP addresses, and so on.

The software was written in PHP for the UI dashboard and Perl for the forensic-analysis algorithms. A typical email contains a series of headers separated by a blank line from a potentially multipart email body that uses the MIME format. Perl proved ideal for the forensic analysis because libraries are readily available that let emails be correctly parsed to provide access to the necessary information. This feature allowed SendForensics to use its expertise quickly, without getting bogged down in the details of parsing emails. (We also chose Perl for its unusual combination of performance, portability, functionality, stability, and suitability for pattern recognition.) The forensic tests constitute an assembly of proprietary signal-processing and pattern recognition algorithms developed from the company's early experience in safety-critical systems and scientific data processing.

After four years of development, there are now around 80,000 lines of PHP, CSS (Cascading Style Sheets), XML, HTML, and other web languages in the dashboards. There are also around 133,000 lines of Perl, and this has been growing over the last three years with a compound annual growth rate of 1.19, almost exactly in the middle of the range that Les Hatton and his colleagues reported.[2]

The Platform

From the beginning, it was clear that we could expect serious volumes of data. Although the system samples emails in a statistical sense, the sheer scale of email volume meant that processing loads would be high, particularly given the variety and number of forensic tests we applied. Four years ago, cloud computing was still stabilizing, and initial versions of our system were deployed on a farm of CentOS Linux "heavy iron" servers at Hetzner in Germany with a handcrafted switched IP failover system. This setup was astonishingly reliable, with no recorded downtime (hardware or OS), but the in-house-developed failover system was ultimately unsatisfactory ("clunky" according to its author). So, 18 months ago, SendForensics moved the system from the physical servers to an Amazon Web Services (AWS) cloud solution after carefully vetting the available features.

Because SendForensics' software is built in open source highly portable languages, moving the software to AWS was gratifyingly quick; it took just a few days. Much is made of the difficulties of moving IT in companies, but it's almost entirely painless if you take care to maintain portability during design and development. The real learning curve manifested itself when we tried to understand the increasingly bewildering array of options in cloud systems. The concepts of virtualization aren't difficult, and the idea of assembling systems using a toolkit of virtualized components was very attractive, but it took a while to understand the jargon. Because email volumes vary dramatically with the time of day, we ended up with a distributed, load-balanced, highly resilient system of grouped mail engines, webservers, and master–replica databases.

This setup has met all expectations of this groundbreaking but somewhat opaque technology regarding elasticity, resilience, and efficiency. The mail engines (Mail Transport Agents) employ SendForensics proprietary forensic technology layered on top of Postfix, a formidably reliable and easy-to-use open source mail program created by Wietse Venema.[3]

Other columns in the Impact department have described single copies of software,[4,5] but in our case, the software's functionality is delivered as a service with cloud instances scaling up as necessary—for example, in the world of search engines.[6] These new SAAS (software as a service) and IAAS (infrastructure as a service) delivery models are increasingly popular and are natural for an Internet standard service such as email.

Despite the growth of social media and other instant-messaging media, email is still the preferred medium for corporate transfer of many kinds of information. It seems likely to remain so, provided the industry can continue to protect itself from nefarious

practices. The arms race will continue, and the benefits to the winner are huge.

Acknowledgments

Department editor Les Hatton is a technical advisor to SendForensics and therefore didn't contribute to any editorial decisions concerning this article.

References

1. J. Robertson, "E-mail Spam Goes Artisanal," *Bloomberg Technology*, 19 Jan. 2016; www.bloomberg.com/news/articles/2016-01-19/e-mail-spam-goes-artisanal.
2. L. Hatton, D. Spinellis, and M. van Genuchten, "The Long-Term Growth Rate of Evolving Software: Empirical Results and Implications," *J. Software: Evolution and Process*, 16 Feb. 2017; doi:10.1002/smr.18472017.
3. K.D. Dent, *Postfix: The Definitive Guide*, O'Reilly, 2004.
4. D. Rousseau, "The Software behind the Higgs Boson Discovery," *IEEE Software*, vol. 29, no. 5, 2012, pp. 11–15.
5. G.J. Holzmann, "Landing a Spacecraft on Mars," *IEEE Software*, vol. 30, no. 2, 2013, pp. 83–86.
6. M. Andrews, "Searching the Internet," *IEEE Software*, vol. 29, no. 2, 2012, pp. 13–16.

LEO HATTON is a cofounder of and the chief executive officer at SendForensics. Contact him at leo@sendforensics.com.

ALAN JOHN is a cofounder of and the chief technology officer at SendForensics. Contact him at alan@sendforensics.com.