Fighting spammers with spam

Alexandros A. Antonopoulos, Kyriakos G. Stefanidis, and Artemios G. Voyiatzis
Department of Electrical and Computer Engineering
University of Patras
GR-26504, Greece
Email: {antonopa,stefanid,bogart}@ece.upatras.gr

***Abstract*—A spammer is making profit and stays in business even when a tiny fraction of recipients replies to spam messages. Despite enormous effort put in spam identification and elimination, spam is still dominating our inbox.**

We seek a novel and complementary approach to force spammers stop sending unsolicited messages. Our approach is to artificially inflate the number of available recipients by contaminating spammers' databases with addresses not monitored by human beings. Thus, we aim to drastically reduce the number of messages delivered to human beings as to reduce the response rate.

***Index Terms*—Unsolicited commercial email, UCE, spam, mail**

I. Introduction

Unsolicited commercial email (UCE) or most commonly referenced as spam has become a major problem in Internet recently. Spam existed even in first days of Internet but lately it dominates email traffic and forces users to seek alternative means of communication. In the enterprise environment, spam results in lost productivity and costs billions of dollars worldwide. Currently, more than 85% of overall email traffic is considered to be spam messages [1].

In the early years of Internet spam traffic was low and not a significant annoyance. Today, spam has transformed to a profitable business process. The spam business model includes three activities [2]:

1) Find potential customers (spam recipients).
2) Offer a product or service to the potential customers.
3) Close the deal.

In the Internet, the cost to find and address possible customers is rather low. Compared to other marketing techniques, spam incurs extremely low costs on the sender side of the spam and moves the real costs towards the receiver side of the information flow. In order to effectively stop the advance of spam traffic we seek a method to revert the cost of spam messages back to their senders. If this is possible, then we seek to make spam sending more expensive than alternative marketing means. In that case, alternative means are preferable over spam and so there will be no spam industry.

The paper is organized as follows: Section II presents currently known spammer techniques in collecting email addresses for targeting messages. Section III presents current advances on fighting spam and their shortcomings. Section IV presents our contribution, a method to incur spam costs back to spammers. Finally, Section V presents our conclusions and future work.

II. Spammer Activities

In this section we review the operating model of a spam business. Fig. 1 illustrates the spam flow from one edge (the sender) to the other edge (the receiver).

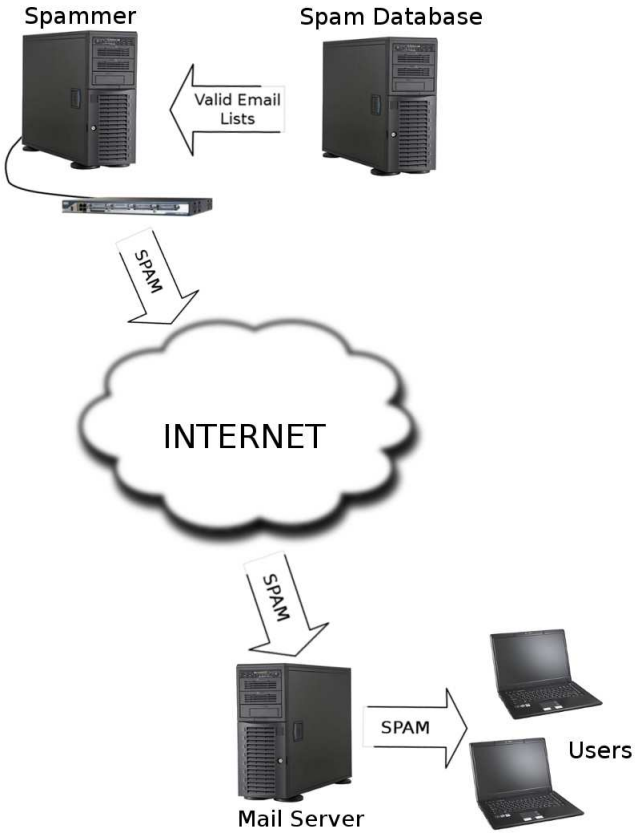

Fig. 1. Spam flow

The first step for a profitable spam business operation is the collection of customer addresses. Little information is available up to day on how potential customer addresses are collected by spammers. Clayton provides some insight [3] and so does Judge et al. [2].

A first method is to harvest valid addresses from mailing list archives, Usenet feeds, websites, and so on. Harvesting can occur directly through spammers computer or via a set of "zombie" stations i.e., computer systems of innocent people that are under the control of a spammer. This set of innocent people's computers is often called a "botnet" and can be

used for other vigilant actions too. The harvesting method can be further enhanced (from a spammer's point of view) by spreading malware such as virus, trojans, and rootkits. These malware can scan user's address book and old email messages. These sources can provide new addresses for spammer's database. Further, by forwarding malware to the newly found addresses, the spammer hopes to infect more computers and thus, to increase the size of the operated botnet.

An improved method is to combine the local part of an email address already in database with different domains: given that `john@example.com` is valid, it might be the case that `john@example2.com` is also valid. Such attempts are called "dictionary" or "Rumpelstiltskin" attacks. If the attack succeeds on a new domain, the address is recorded on the spam database for future use.

A third method is to purchase email lists of varying quality from underground sources. Such lists contain information collected by the aforementioned methods and their price is defined by their quality. For example lists with customers that have purchased in the past through spam messages are reported to be non-tradeable because of their value [2].

Once enough addresses are collected, the spammer starts sending the (spam) messages. For this, computing resources (to store message and recipients and to process the mail protocol) and bandwidth are needed. The bandwidth is the most crucial factor on sending messages as fast as possible. Spammers either rent bandwidth or utilize the botnet for sending messages through it. The former approach (bandwidth) is much more preferable, if an ISP is willing to sell bandwidth for such an activity, since the spammer has full control on message sending. The latter (botnet) may cause some troubles, since the spammer must accomodate with diffenent security policies that various ISP might enforce on their network. In cases that a policy changes, the spammer must adapt botnet member's operation accordingly. Such an example is when the ISP blocks general access to port 25/TCP (used by the SMTP protocol) and requires authentication prior to accessing it. Adapting in a constantly changing environment with different policies can introduce significant management costs for the spammer.

III. Fighting spam

Over the time, we have witnessed a rewind process in fighting spam compared to Fig. 1. The first, natural approach was to do nothing about spam and let the user, the receiver, delete unwanted messages. Since then, the spam messages have risen to 80-90% of all mail traffic. Such high volumes of spam result in lost productivity which translates to increased costs for an organization. Thus, the second approach was to automate the process and enhance user's mail software with filtering mechanisms that examine incoming messages and flag possible spam messages. Spammers quickly adapted to these filters, for example by sending images instead of text, inserting random text, and salting the random text. Software makers have improved detection algorithms and capabilities accordingly but still the amount of spam is increasing.

The next step was to move filtering in central locations i.e., the user's mail server. This allowed for less administration overhead and better training of the filters. Yet, it is still even nowadays that spammers have a clear win.

Since spam originates from specific addresses or block of addresses, a new method was sender taxonomy independently of content; the sender IP could be whitelisted ("good" or "trusted" sender), greylisted (unknown or undefined sender), or blacklisted ("bad" sender). Collaborative action between mail server operators resulted in publicly accessible (black)lists that are available 24x7. However, these lists require continuous update and spammers find ways to bypass them even for a small fraction of time. This is sufficient for them to send enormous volumes of spam messages until getting blacklisted once more.

One different approach was the proposal to enhance current Internet protocols, while maintaining backwards compatibility, or to propose totally new mail protocols. Such methods include tarpits, puzzles, and sender authentication.

Apart from the aforementioned technical means, there have been proposals for fighting spam using regulations and political means. Such actions include legislation that holds senders accountable (and punishable) for their actions, maintenance of "do not call" lists, and charging email messages. Given the autonomous and decentralized nature of Internet architecture, such means have exhibited little or no success. Legislation is not enforceable globally but rather in country or even smaller administrative units. Charging faces same challenges as legislation, but is amplified by the fact that Internet citizens are not accustomed and not willing to pay for such a service.

It is our belief that user education provides the long term solution to spam. If users are educated to ignore spam, then the response rate will be so low that there will not be any motivation to send spam mail. However, such long term and dedicated education is really hard to achieve, given the fact that Internet population increases exponentially and steadily refreshes.

A. Related Work

Current work on spam fighting can be categorized as follows:

- Content-based: Search messages for suspicious content. Many solutions search for specific keywords or patterns in order to filter spam messages. Nevertheless these techniques are easily bypassed by using word obscuring (misspelling, images etc.), token breaking (adding spaces inside words or replacing characters with the escaped equivalent) and text replacement with images.
- IP-based: Message senders are categorized in white, grey and black lists. Trusted senders are added in white lists. Unknown senders begin in grey lists. Depending on their behavior they can be moved either to white lists or if they are proven to be spammers to black lists. Currently blacklists are widely used. One drawback of this method is that the attacker can use address spoofing,

thus bypassing this filtering and furthermore legitimate users can be blocked if a spammer spoofs their address.
- Protocol-based: Ongoing modifications of existing mail protocols or even proposals for new protocols to support fight against spam [4], [5].
- Distributed spam filter databases such as the one described in [6]. Like content based filtering, those databases suffer from the fact that they rely on the same spam message being sent to multiple recipients. Spam messages can be easily customised for each recipient thus limiting the success rate of these methods.
- Reputation filtering [7]: By adopting a social networking system, trusted users can report a message as spam. This way only few users will receive a spam message. After they report it, a server can filter out this message. Although this method can reduce the number of clients a spam message reaches, it still requires action from some volunteers.
- Fingerprints, checksums and signatures. By creating a checksum of various message parts, the mail server can detect how many times a message has been sent. Distributed Checksum Crealinghouses (DCC) use fuzzy checksums to prevent spammers from inserting random characters in their messages [8]. By producing correct checksums, DCC can detect mass mailers and report the total number of targeted clients. DCC reports are utilized to block or allow a message via tools like SpamAssasin [9].
- Mail encryption. PKI offers the infrastructure to sign and verify the mail sender [10]. However, only a small percentage of mail users have a valid certificate, thus rendering PKI inefficient.
- Spam lists pollution. Tools like `Wpoison` combats spam by injecting random fake email addresses to spam web crawlers [11]. The problem with random email addresses is that they can be validated and dropped from the spammer database.
- Other solutions try to move the cost of spam sending to the spammer. Spammers depend on sending a vast amount of emails in relatively small periods. By introducing send delays or monetary fees the cost for the spammer increases significantly [10]. For a normal user this cost (time and/or money) is minimal.

By just keeping spam messages away from inboxes the problem is only hidden and not fixed. Most of the above spam fighting approaches don't try to bring the problem to the spammer. We believe that the only efficient way of fighting spam is by eliminating the main motive for spamming i.e., the profit. Techniques that minimize the profit per spam message are needed in order to minimize spamming profit margins. This profit comes from mails that lure people to follow a link or try and buy a product. In the next Section we propose a method to contaminate spammer databases with addresses that a legitimate mail server will happily accept mail for as valid; this traffic is never routed to real users and thus, the response rate of spam will decrease and the cost per message will increase.

IV. Spam Database Pollution

Spammers collect top quality addresses i.e., addresses that are known to work. Currently a spam database contains almost 100% valid addresses. It has been testified that it suffices even one such address to get fooled every one million messages in order to keep the spammer in business [12].

Our proposal is to make the collection of valid addresses as hard as possible. It is preferable to make the percentage 10%, 1% or even less. Then, order(s) of magnitude more messages will be required to get one fooled reply. Thus, much more precious network bandwidth resources are needed in order to keep up the response rate in sufficient levels.

The obvious approach is to educate users not to publish their real addresses to such places. This approach is unrealistic to our belief, at least for the next few years. Furthermore, it is a rather selfish approach; if we "remove" addresses from the spam database, then the quality of the database rises back to 100%. This means that spam victims that are not removed from the database and do not reply in spam messages will now receive even more spam messages.

It is our belief that we need a cooperative approach. This approach requires that we collectively inject false (but routable) addresses on spam databases. As we show in the following, there are incentives and motivation for participation both in a local (company) and in a global (world) scale.

In real world, the population P is "well-known" through census and statistics. So, the parameter P is under control. This does not hold in the virtual world and specially for email addresses.

Our proposal is to try to increase in a cheap way the target population P, instead of trying to minimize the response rate r of the fixed population. In either way, the ratio r/P will fall, hopefully below some limiting response rate $x\%$ and the spammer will be out of business. The idea is to publish fake email addresses that will pollute spammer's address databases. If we increase tenfold our addresses, then only $r/10$ of spam messages blindly directed to our domain will manage to reach a "real" mailbox and thus, our domain users will receive 90% less spam.

We consider as t_C the total resources of a company C. Resources can include for example bandwidth, processing time, and storage. We consider an amount e_C of these resources to be wasted for spam processing. We assume that $e_C << t_C$ i.e., the spam processing occupies only a small fraction of the total resources of the company. We propose to allocate $10e_c$ or even more resources for spam processing. This overallocation for useless processing is devoted in publishing and accepting more spam messages; these messages are not stored or further processed but rather dropped upon reception.

This approach is a win-win situation if applied in local and in global scale. In local (domain-wide) scale, our domain users receive 90% less spam with no intervention. In global scale, spammer's network resources must be increased tenfold, in

order to achieve same response rates. Here is the asymmetry of the cost; this asymmetry is the key to bring the cost of sending spam back to the spammer. Under the tenfold increase assumption, the spammer must generously increase the bandwidth used to send spam messages, since spam email is the only traffic a spammer generates. On the other hand, for each domain, the email traffic is minimal compared to web surfing, content downloading, or even P2P traffic. Thus, the cost of the mail traffic increase can be considered negligible.

Someone could argue that this approach would cause for a domain an increase to the number of received spam messages. The average spam bot sends about 10 spam emails per day to the same domain [13], [14]. This hinders the effectiveness of todays anti-spam methods and blacklists which rely on the detection of spikes or persistence in spam email traffic from individual hosts. On the other hand, it serves as one more driving conclusion for our anti-spam scheme. We see that the amount of spam send to each domain is not analogous to the number of harvested email addresses that exist in the bots spam database but is limited to a certain threshold that helps the bot to stay undetected from personal or distributed anti-spam methods. Thus, the more we manage to infect this database with false email addresses from a certain domain, the less real spam this domain gets.

A. Practical Examples

A company maintains a cleverly designed "corporate directory" web page which "leaks" company email addresses. A first-generation spam crawler will happily harvest the email addresses and signal spam database to start sending messages. It is easy to see that by dedicating some more resources for the extra traffic that these fake addresses will produce, quite less spam traffic will be directed to company mailboxes. If the company has 10 real addresses and the spammer sends every day 10 spam messages, then 1 message will reach each real address. Now if the company manages to infect back the spam database with 90 fake addresses and the spammer keeps sending 10 messages, then, statistically, only 1 out of 10 messages will eventually reach a real mailbox. Thus, the company becomes more productive and receives less "real" spam.

In order to fight back second-generation spam crawlers, the company collects fake "corporate directory" information from other companies (for example from its clients or vendors). Addresses on these directories are then happily accepted on its mail server thus increasing the "attack" surface for a spammer (more useless messages that will not be delivered to real mailboxes).

Except from web crawlers as a method of harvesting email addresses, worms also perform the email harvesting as part of their multipurpose functions (DDoS, Spam sending, Key-logging, etc). By infecting a user machine, the worm has in its disposal a collection of 100% valid email addresses that reside in the users various local address books. With the use of vulnerable honeypots we can control the harvested information. These honeypots can keep address books with valid mail addresses that don't correspond to real users. At the same time while infection of real machines is undesired it is reality. This can be used in the fight against spam. Users can get transparently fake email addresses from their ISP's. There isn't any cost in keeping these lists but if a user is infected at least there is the bonus of polluting spammer databases. Web servers can also feed these lists of valid but not real email addresses to web crawlers. The mail server can be instructed to delete upon reception all messages destined to these addresses. Fig. 2 illustrates this information flow on manipulating harvesting tools.

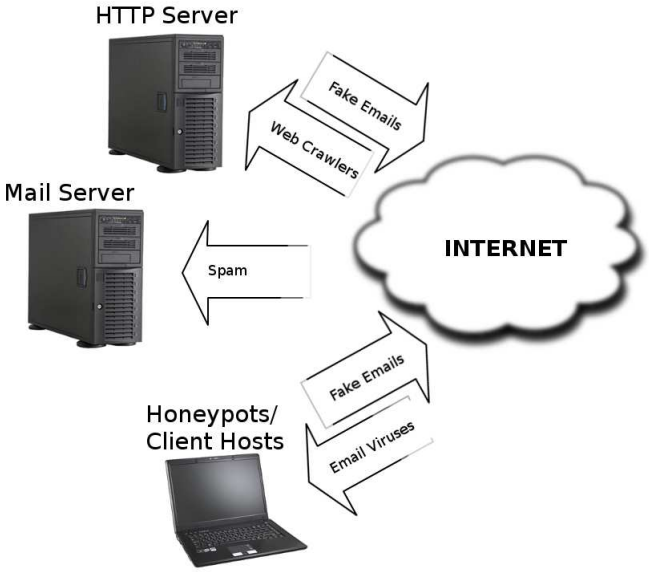

Fig. 2. Manipulating web crawlers and worms

V. Conclusion and Future Work

We proposed an novel approach to fight spam; spam itself. Our approach is to contaminate spam databases with fake addresses that are not monitored by human beings. In this case, spammers waste orders of magnitude more resources to retain the absolute minimum response rate to stay in business. The asymmetry of resources is the key observation for the successful deployment of out approach: the spammer dedicates all the resources to spam sending while on the other hand the recipients dedicate only a fraction of their resources for general email processing (including spam).

We seek to further evaluate the effectiveness of our idea; at first in a small scale controlled environment and then in the open Internet. Our approach can work complementary with other spam fighting techniques. It is an interesting to explore how our idea can further assist or enhance other approaches, both in an enterprise environment and in email service providers.

ACKNOWLEDGMENT

A. Antonopoulos' contribution is part of the 03ED375 research project, implemented within the framework of the "Reinforcement Programme of Human Research Manpower" (PENED) and co-financed by National and Community Funds (20% from the Greek Ministry of Development-General Secretariat of Research and Technology and 80% from E.U.-European Social Fund).

REFERENCES

[1] SpamLinks. Spam statistics. [Online]. Available: http://spamlinks.net/stats.htm

[2] W. Judge and D. Alperovitch, "Understanding and reversing the profit model of spam," in *Fourth Workshop on the Economics of Information Security*, Boston, MA, USA, Jun.2–3, 2005.

[3] R. Clayton, "Do zebras get more spam than aardvarks?" in *Fifth Conference on Email and Anti-Spam*, Mountain View, California, Aug.21–22, 2008. [Online]. Available: http://www.cl.cam.ac.uk/~rnc1/aardvark.pdf

[4] D. Bernstein. Internet Mail 2000. [Online]. Available: http://cr.yp.to/im2000.html

[5] B. Weinman. A replacement for SMTP. [Online]. Available: http://amtp.bw.org/

[6] E. Damiani, S. D. C. di Vimercati, S. Paraboschi, and P. Samarati, "P2P-based collaborative spam detection and filtering," in *Proc. of the Fourth International Conference on Peer-to-Peer Computing*, Zurich, Switzerland, Aug.15–17, 2004.

[7] V. V. Prakash and A. O'Donnell, "Fighting spam with reputation systems," *ACM Queue*, vol. 3, no. 9, pp. 36–41, November 2005.

[8] Rhyolite Software. Distributed Checksum Clearinghouses. [Online]. Available: http://www.rhyolite.com/dcc/

[9] Apache Software Foundation. The Apache SpamAssassin Project. [Online]. Available: http://spamassassin.apache.org/

[10] W. Gansterer, M. Ilger, P. Lechner, R. Neumayer, and J. Straub, "Anti-spam methods - state of the art," Institute of Distributed and Multimedia Systems, Faculty of Computer Science, University of Vienna, Tech. Rep., 2005.

[11] Infinite Monkeys & Co. LLC. WPOISON. [Online]. Available: http://www.monkeys.com/wpoison

[12] I.-H. Hann, K.-L. Hui, Y.-L. Lai, S. Lee, and I. Png, "Who gets spammed?" *Communications of the ACM*, vol. 49, no. 10, pp. 83–87, Oct. 2006.

[13] J. Jung and E. Sit, "An empirical study of spam traffic and the use of DNS black lists," in *4th ACM SIGCOMM Conference on Internet Measurement*, Taormina, Sicily, Italy, Oct.25–27, 2004.

[14] A. Ramachandran, D. Dagon, and N. Feamster, "Can DNS-based blacklists keep up with bots?" in *Third Conference on Email and Anti-Spam*, Mountain View, California, Jul.27–28, 2006. [Online]. Available: http://www.ceas.cc/2006/14.pdf