DOI:10.1145/1562164.1562190

Spamalytics: An Empirical Analysis of Spam Marketing Conversion

By Chris Kanich, Christian Kreibich, Kirill Levchenko, Brandon Enright, Geoffrey M. Voelker, Vern Paxson, and Stefan Savage

Abstract

The "conversion rate" of spam—the probability that an unsolicited email will ultimately elicit a "sale"—underlies the entire spam value proposition. However, our understanding of this critical behavior is quite limited, and the literature lacks any quantitative study concerning its true value. In this paper we present a methodology for measuring the conversion rate of spam. Using a parasitic infiltration of an existing botnet's infrastructure, we analyze two spam campaigns: one designed to propagate a malware Trojan, the other marketing online pharmaceuticals. For nearly a half billion spam emails we identify the number that are successfully delivered, the number that pass through popular antispam filters, the number that elicit user visits to the advertised sites, and the number of "sales" and "infections" produced.

1. INTRODUCTION

Spam-based marketing is a curious beast. We all receive the advertisements—"Excellent hardness is easy!"—but few of us have encountered a person who admits to following through on this offer and making a purchase. And yet, the relentlessness by which such spam continually clogs Internet inboxes, despite years of energetic deployment of antispam technology, provides undeniable testament that spammers find their campaigns profitable. Someone is clearly buying. But how many, how often, and how much?

Unraveling such questions is *essential* for understanding the economic support for spam and hence where any structural weaknesses may lie. Unfortunately, spammers do not file quarterly financial reports, and the underground nature of their activities makes third-party data gathering a challenge at best. Absent an empirical foundation, defenders are often left to speculate as to how successful spam campaigns are and to what degree they are profitable. For example, IBM's Joshua Corman was widely quoted as claiming that spam sent by the Storm worm alone was generating "millions and millions of dollars every day."[1] While this claim could in fact be true, we are unaware of any public data or methodology capable of confirming or refuting it.

The key problem is our limited visibility into the three basic parameters of the spam value proposition: the cost to send spam, offset by the "conversion rate" (probability that an email sent will ultimately yield a "sale"), and the marginal profit per sale. The first and last of these are self-contained and can at least be estimated based on the costs charged by third-party spam senders and through the pricing and gross margins offered by various Interne marketing "affiliate programs."[a] However, the conversion rate depends fundamentally on group actions—on what hundreds of millions of Internet users do when confronted with a new piece of spam—and is much harder to obtain. While a range of anecdotal numbers exist, we are unaware of any well-documented measurement of the spam conversion rate.[b]

In part, this problem is methodological. There are no apparent methods for indirectly measuring spam conversion. Thus, the only obvious way to extract this data is to build an e-commerce site, market it via spam, and then record the number of sales. Moreover, to capture the spammer's experience with full fidelity, such a study must also mimic their use of illicit botnets for distributing email and proxying user responses. In effect, the best way to measure spam is to be a spammer.

In this paper, we have effectively conducted this study, though *sidestepping* the obvious legal and ethical problems associated with sending spam.[c] Critically, our study makes use of an *existing* spamming botnet. By infiltrating the botnet parasitically, we convinced it to modify a subset of the spam it *already* sends, thereby directing any interested recipients to Web sites under our control, rather than those belonging to the spammer. In turn, our Web sites presented "defanged" versions of the spammer's own sites, with functionality removed that would compromise the victim's system or receive sensitive personal information such as name, address or credit card information.

Using this methodology, we have documented three spam campaigns comprising over 469 million emails. We identified how much of this spam is successfully delivered,

[a] Our cursory investigations suggest that commissions on pharmaceutical affiliate programs tend to hover around 40%–50%, while the *retail* cost for spam delivery has been estimated at under $80 per million.[14]

[b] The best known among these anecdotal figures comes from the *Wall Street Journal*'s 2003 investigation of Howard Carmack (a.k.a. the "Buffalo Spammer"), revealing that he obtained a 0.00036 conversion rate on 10 million messages marketing an herbal stimulant.[3]

[c] We conducted our study under the ethical criteria of ensuring *neutral actions* so that users should never be worse off due to our activities, while strictly *reducing harm* for those situations in which user property was at risk.

A previous version of this paper appeared in *Proceedings of the 15th ACM Conference on Computer and Communications Security*, Oct. 2008.

how much is filtered by popular antispam solutions, and, most importantly, how many users "click-through" to the site being advertised (*response rate*) and how many of those progress to a "sale" or "infection" (*conversion rate*).

The remainder of this paper is structured as follows. Section 2 describes the economic basis for spam and reviews prior research in this area. Section 4 describes our experimental methodology for botnet infiltration. Section 5 describes our spam filtering and conversion results, Section 6 analyzes the effects of blacklisting on spam delivery, and Section 7 analyzes the possible influences on spam responses. We synthesize our findings in Section 8 and conclude.

2. BACKGROUND

Direct marketing has a rich history, dating back to the nineteenth century distribution of the first mail-order catalogs. What makes direct marketing so appealing is that one can directly measure its return on investment. For example, the Direct Mail Association reports that direct mail sales campaigns produce a response rate of 2.15% on average.[4] Meanwhile, rough estimates of direct mail *cost per mille*—the cost to address, produce and deliver materials to a thousand targets—range between $250 and $1000. Thus, following these estimates it might cost $250,000 to send out a million solicitations, which might then produce 21,500 responses. The cost of developing these prospects (roughly $12 each) can be directly computed and, assuming each prospect completes a sale of an average value, one can balance this revenue directly against the marketing costs to determine the profitability of the campaign. As long as the product of the conversion rate and the marginal profit per sale exceeds the marginal delivery cost, the campaign is profitable.

Given this underlying value proposition, it is not at all surprising that bulk direct email marketing emerged very quickly after email itself. The marginal cost to send an email is tiny and, thus, an email-based campaign can be profitable even when the conversion rate is negligible. Unfortunately, a perverse byproduct of this dynamic is that sending as much spam as possible is likely to maximize profit.[8]

While spam has long been understood to be an economic problem, it is only recently that there has been significant effort in modeling spam economics and understanding the value proposition from the spammer's point of view. Rarely do spammers talk about financial aspects of their activities themselves, though such accounts do exist.[10, 13] Judge et al. speculate that response rates as low as 0.000001 are sufficient to maintain profitability.[12]

However, the work that is most closely related to our own are the several papers concerning "Stock Spam."[5, 7, 9] Stock spam refers to the practice of sending positive "touts" for a low-volume security in order to manipulate its price and thereby profit on an existing position in the stock. What distinguishes stock spam is that it is monetized through price manipulation and not via a sale. Consequently, it is not necessary to measure the conversion rate to understand profitability. Instead, profitability can be inferred by correlating stock spam message volume with changes in the trading volume and price for the associated stocks.

3. THE STORM BOTNET

The measurements in this paper are carried out using the Storm botnet and its spamming agents. Storm is a peer-to-peer botnet that propagates via spam (usually by directing recipients to download an executable from a Web site).

Storm Hierarchy: There are three primary classes of machines that the Storm botnet uses when sending spam. Worker bots make requests for work and, upon receiving orders, send spam as requested. Proxy bots act as conduits between workers and master servers. Finally, the master servers provide commands to the workers and receive their status reports. In our experience there are a very small number of master servers (typically hosted at so-called "bullet-proof" hosting centers) and these are likely managed by the botmaster directly.

However, the distinction between worker and proxy is one that is determined automatically. When Storm first infects a host it tests if it can be reached externally. If so, then it is eligible to become a proxy. If not, then it becomes a worker. All of the bots we ran as part of our experiment existed as proxy bots, being used by the botmaster to ferry commands between master servers and the worker bots responsible for the actual transmission of spam messages.

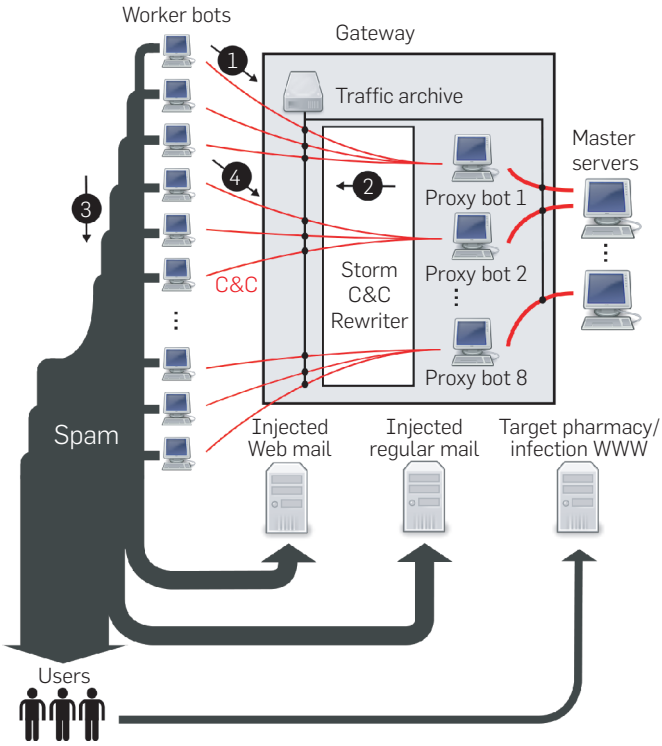
4. METHODOLOGY

Our measurement approach is based on *botnet infiltration*—that is, insinuating ourselves into a botnet's "command and control" (C&C) network, passively observing the spam-related commands and data it distributes and, where appropriate, actively changing individual elements of these messages in transit. Storm's architecture lends itself particularly well to infiltration since the proxy bots, *by design*, interpose on the communications between individual worker bots and the master servers who direct them. Moreover, since Storm compromises hosts indiscriminately (normally using malware distributed via social engineering Web sites) it is straightforward to create a proxy bot on demand by infecting a globally reachable host under our control with the Storm malware.

Figure 1 also illustrates our basic measurement infrastructure. At the core, we instantiate eight unmodified Storm proxy bots within a controlled virtual machine environment. The network traffic for these bots is then routed through a centralized gateway, providing a means for blocking unanticipated behaviors (e.g., participation in DDoS attacks) and an interposition point for parsing C&C messages and "rewriting" them as they pass from proxies to workers. Most critically, by carefully rewriting the spam template and dictionary entries sent by master servers, we arrange for worker bots to replace the intended site links in their spam with URLs of our choosing. From this basic capability we synthesize experiments to measure the click-through and conversion rates for several large spam campaigns.

C&C Protocol Rewriting: Our runtime C&C protocol rewriter consists of two components. A custom router redirects potential C&C traffic to a fixed IP address and port, where a user-space proxy server accepts incoming connections and impersonates the proxy bots. This server in turn forwards connections back into the router, which redirects the traffic

Figure 1. The Storm spam campaign dataflow and our measurement and rewriting infrastructure (Section 4). (1) Workers request spam tasks through proxies, (2) proxies forward spam workload responses from master servers, (3) workers send the spam, and (4) return delivery reports. Our infrastructure infiltrates the C&C channels between workers and proxies.



to the intended proxy bot. Rules for rewriting can be installed independently for templates, dictionaries, and email address target lists. The rewriter logs all C&C traffic between worker and our proxy bots, between the proxy bots and the master servers, and all rewriting actions on the traffic.

Measuring Spam Delivery: To evaluate the effect of spam filtering along the email delivery path to user inboxes, we established a collection of test email accounts and arranged to have Storm worker bots send spam to those accounts. These accounts were created at several different vantage points from which we could evaluate the effectiveness of different email filtering methods. When a worker bot reports success or failure back to the master servers, we remove any success reports for our email addresses to hide our modifications from the botmaster.

We periodically poll each email account (both inbox and "junk/spam" folders) for the messages that it received, and we log them with their timestamps, filtering out any messages not part of this experiment.

Measuring Click-Through and Conversion: To evaluate how often users who receive spam actually visit the sites advertised requires monitoring the advertised sites themselves. Since it is generally impractical to monitor sites not under our control, we have used our *botnet infiltration* method to arrange to have a fraction of Storm's spam advertise sites of our creation instead.

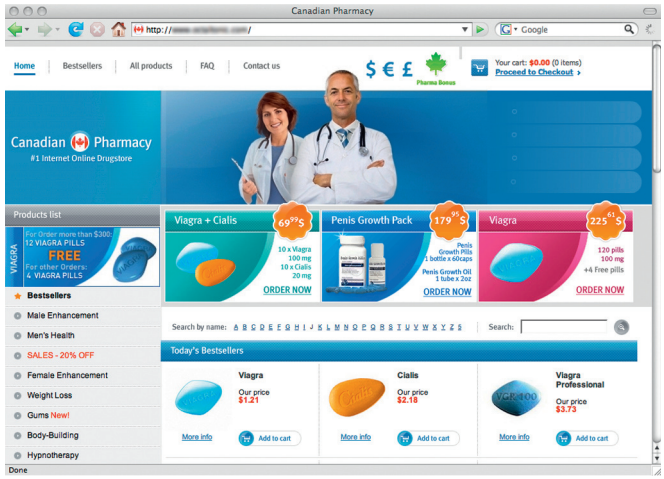
In particular, we have focused on two types of Storm spam campaigns, a self-propagation campaign designed to spread the Storm malware (typically under the guise of advertising an electronic postcard site) and the other advertising a pharmacy site. These are the two most popular Storm spam campaigns and represent over 40% of recent Storm activity.[11] We replaced Storm's links to its own sites with links to sites under our control, screenshots of which are shown in Figure 2.

These sites have been "defanged" in two important ways: the pharmaceutical site does not accept any personal or payment information, and the self-propagation site advertises a completely benign executable which only phones home to record an execution and exits.

4.1. Measurement ethics

We have been careful to design experiments that we believe are both consistent with current U.S. legal doctrine and are fundamentally ethical as well. While it is beyond the scope of this paper to fully describe the complex legal landscape in which active security measurements operate, we believe the ethical basis for our work is far easier to explain: *we strictly reduce harm*. First, our instrumented proxy bots do not create any new harm. That is, absent our involvement, the same set of users would receive the same set of spam emails sent by the same worker bots. Storm is a large self-organizing system and when a proxy fails its worker bots

Figure 2. Screenshots of the Web sites operated to measure user click-through and conversion.



(a) Pharmaceutical site

(b) Postcard-themedself-propagation site

automatically switch to other idle proxies (indeed, when our proxies fail we see workers quickly switch away). Second, our proxies are passive actors and do not engage themselves in any behavior that is intrinsically objectionable; they do not send spam email, they do not compromise hosts, nor do they even contact worker bots asynchronously. Indeed, their only function is to provide a conduit between worker bots making requests and master servers providing responses. Finally, where we do modify C&C messages in transit, these actions themselves strictly reduce harm. Users who click on spam altered by these changes will be directed to one of our innocuous doppelganger Web sites. Unlike the sites *normally* advertised by Storm, our sites do not infect users with malware and do not collect user credit card information. Thus, no user should receive more spam due to our involvement, but some users will receive spam that is less dangerous that it would otherwise be.

Needless to say, we encourage no one to recreate our experiments without the utmost preparation and care. Interacting with thousands of compromised machines that are sending millions of spam messages is a very delicate procedure, and while we encourage other researchers to build upon our work, we ask that these experiments only be attempted by qualified professionals with no less forethought, legal consultation, or safeguards than those outlined here.

5. EXPERIMENTAL RESULTS

We now present the overall results of our rewriting experiment. We first describe the spam workload observed by our C&C rewriting proxy. We then characterize the effects of filtering on the spam workload along the delivery path from worker bots to user inboxes, as well as the number of users who browse the advertised Web sites and act on the content there.

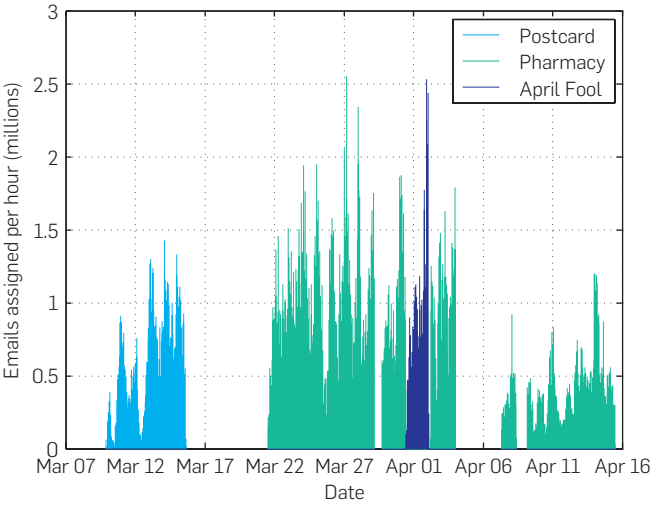
Campaign Datasets: Our study covers three spam campaigns summarized in Table 1. The "Pharmacy" campaign is a 26-day sample (19 active days) of an ongoing Storm campaign advertising an online pharmacy. The "Postcard" and "April Fool" campaigns are two distinct, serial instances of self-propagation campaigns, which attempt to install an executable on the user's machine under the guise of being postcard software. For each campaign, Figure 3 shows the number of messages per hour assigned to bots for mailing.

Storm's authors have shown great cunning in exploiting the cultural and social expectations of users—hence the April Fool campaign was rolled out for a limited run around April 1. Our Web site was designed to mimic the earlier

Table 1. Campaigns used in the experiment.

Campaign	Dates	Workers	Emails
Pharmacy	March 21–April 15	31,348	347,590,389
Postcard	March 9–March 15	17,639	83,665,479
April Fool	March 31–April 2	3,678	38,651,124
		Total	469,906,992

Figure 3. Number of email messages assigned per hour for each campaign.

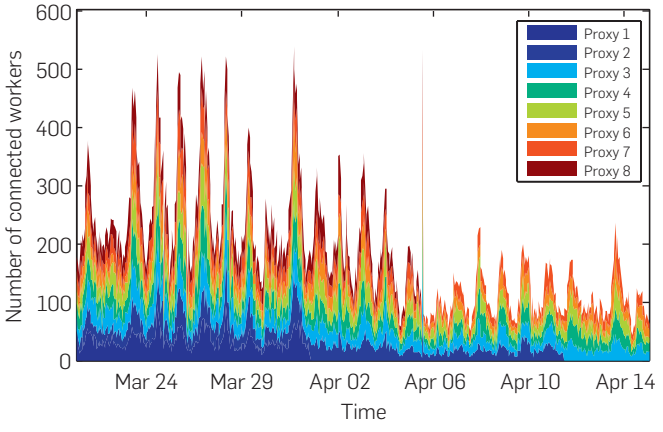


Postcard campaign and thus our data probably does not perfectly reflect user behavior for this campaign, but the two are similar enough in nature that we surmise that any impact is small.

We began the experiment with eight proxy bots, of which seven survived until the end. Figure 4 shows a timeline of the proxy bot workload. The number of workers connected to each proxy is roughly uniform across all proxies (23 worker bots on average), but shows strong spikes corresponding to new self-propagation campaigns. At peak, 539 worker bots were connected to our proxies at the same time.

Most workers only connected to our proxies once: 78% of the workers only connected to our proxies a single time, 92% at most twice, and 99% at most five times. The most prolific worker IP address, a host in an academic network in North Carolina, USA, contacted our proxies 269 times; further inspection identified this as a NAT egress point for 19 individual infections. Conversely, most workers do not connect to more than one proxy: 81% of the workers only connected to a single proxy, 12% to two, 3% to four, 4% connected to five

Figure 4. Timeline of proxy bot workload.



or more, and 90 worker bots connected to all of our proxies. On average, worker bots remained connected for 40 min, although over 40% workers connected for less than a minute. The longest connection lasted almost 81 h.

The workers were instructed to send postcard spam to 83,665,479 addresses, of which 74,901,820 (89.53%) are unique. The April Fool campaign targeted 38,651,124 addresses, of which 36,909,792 (95.49%) are unique. Pharmacy spam targeted 347,590,389 addresses, of which 213,761,147 (61.50%) are unique.

Spam Conversion Pipeline: Conceptually, we break down spam conversion into a pipeline with five "filtering" stages Figure 5 illustrates this pipeline and shows the type of filtering at each stage. The pipeline starts with delivery lists of target email addresses sent to worker bots (Stage A). For a wide range of reasons, workers will successfully deliver only a subset of their messages to an MTA (Stage B). At this point, spam filters at the site correctly identify many messages as spam, and drop them or place them aside in a spam folder. The remaining messages have survived the gauntlet and appear in a user's inbox as valid messages (Stage C). Users may delete or otherwise ignore them, but some users will act on the spam, click on the URL in the message, and visit the advertised site (Stage D). These users may browse the site, but only a fraction "convert" on the spam (Stage E) by attempting to purchase products (pharmacy) or by downloading and running an executable (self-propagation).

We show the spam flow in two parts, "crawler" and "converter," to differentiate between real and masquerading users. For example, the delivery lists given to workers contain honeypot email addresses. Workers deliver spam to these honeypots, which then use crawlers to access the sites referenced by the URL in the messages. Since we want to measure the spam conversion rate for actual users, we separate out the effects of automated processes like crawlers, including only clicks we believe to be user-generated in our results.

Table 2 shows the effects of filtering at each stage of the conversion pipeline for both the self-propagation and pharmaceutical campaigns. The number of targeted addresses (A) is simply the total number of addresses on the delivery lists received by the worker bots during the measurement period, excluding the test addresses we injected.

We obtain an estimate of the number of messages delivered to a mail server (B) by relying on delivery reports generated by the workers. The number of messages delivered to a user's inbox (C) is a much harder value to estimate. We do not know what spam filtering, if any, is used by each mail provider, and then by each user individually, and therefore cannot reasonably estimate this number in total. It is possible, however, to determine this number for individual mail providers or spam filters. The three mail providers and the spam filtering appliance we used in this experiment had a method for separating delivered mails into "junk" and inbox categories. Table 3 gives the number of messages delivered a user's inbox for the free email providers, which together accounted for about 16.5% of addresses targeted by Storm (Table 3), as well as our department's commercial spam filtering appliance. It is important to note that these are results from one spam campaign over a short period of time and should not be used as measures of the relative effectiveness for each service. That said, we observe that the popular Web mail providers all do a very a good job at filtering the campaigns we observed, although it is clear they use different methods (e.g., Hotmail rejects most Storm spam at the mail server level, while Gmail accepts a significant fraction only to filter it later as junk).

The number of visits (D) is the number of accesses to our emulated pharmacy and postcard sites, excluding any crawlers. We note that crawler requests came from a small fraction of hosts but accounted for the majority of all requests to our sites. For the pharmacy site, for instance, of the 11,720 unique IP addresses seen accessing the site with a valid unique identifier, only 10.2% were blacklisted as crawlers. In contrast, 55.3% of all unique identifiers used in requests originated from these crawlers. For all nonimage requests made, 87.43% were made by blacklisted IP addresses.

The number of conversions (E) is the number of visits to the purchase page of the pharmacy site, or the number of executions of the fake self-propagation program.

Figure 5. The spam conversion pipeline.

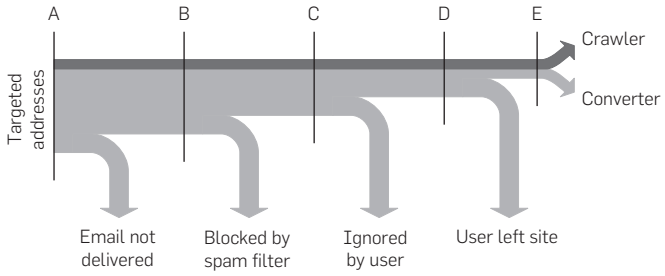


Table 2. Filtering at each stage of the spam conversion pipeline for the self-propagation and pharmacy campaigns. Percentages refer to the conversion rate relative to Stage A.

Stage	Pharmacy		Postcard		April Fool	
A—Spam Targets	347,590,389	100%	83,655,479	100%	40,135,487	100%
B—MTA delivery(est.)	82,700,000	23.8%	21,100,000	25.2%	10,100,000	25.2%
C—Inbox delivery	–	–	–	–	–	–
D—User site visits	10,522	0.00303%	3,827	0.00457%	2,721	0.00680%
E—User conversions	28	0.0000081%	316	0.000378%	225	0.000561%

Table 3. Number of messages delivered to a user's inbox as a fraction of those injected for test accounts at free email providers and a commercial spam filtering appliance. The test account for the Barracuda appliance was not included in the Postcard campaign.

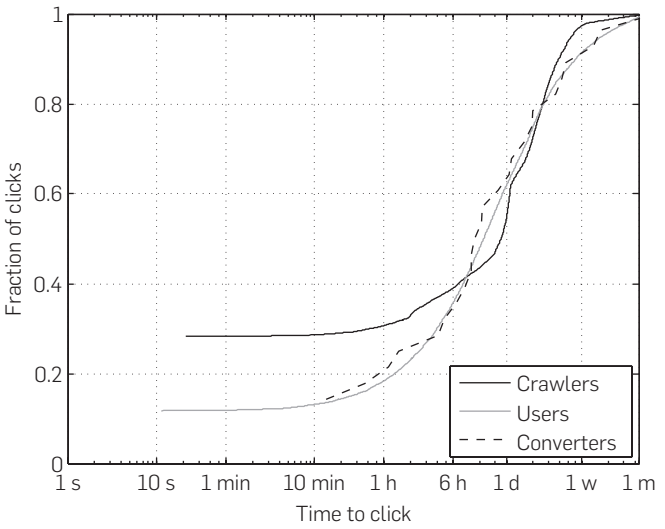
Spam Filter	Pharmacy	Postcard	April Fool
Gmail	0.00683%	0.00176%	0.00226%
Yahoo	0.00173%	0.000542%	None
Hotmail	None	None	None
Barracuda	0.131%	N/A	0.00826%

Our results for Storm spam campaigns show that the spam conversion rate is quite low. For example, out of 350 million pharmacy campaign emails only 28 conversions resulted (and no crawler ever completed a purchase so errors in crawler filtering plays no role). However, a very low conversion rate does not necessary imply low revenue or profitability. We discuss the implications of the conversion rate on the spam conversion proposition further in Section 8.

Time-to-Click: The conversion pipeline shows what fraction of spam ultimately resulted in visits to the advertised sites. However, it does not reflect the latency between when the spam was sent and when a user clicked on it. The longer it takes users to act, the longer the scam hosting infrastructure will need to remain available to extract revenue from the spam.[2] Put another way, how long does a spam-advertised site need to be online to collect potential revenue?

Figure 6 shows the cumulative distribution of the "time-to-click" for accesses to the pharmacy site. The time-to-click is the time from when spam is sent (when a proxy forwards a spam workload to a worker bot) to when a user "clicks" on the URL in the spam (when a host first accesses the Web site). The graph shows three distributions for the accesses by all users, the users who visited the purchase page ("converters"), and the automated crawlers (14,716 such accesses).

Figure 6. Time-to-click distributions for accesses to the pharmacy site.



The user and crawler distributions show distinctly different behavior. Almost 30% of the crawler accesses are within 20 s of worker bots sending spam. This behavior suggests that these crawlers are configured to scan sites advertised in spam immediately upon delivery. Another 10% of crawler accesses have a time-to-click of 1 day, suggesting crawlers configured to access spam-advertised sites periodically in batches. In contrast, only 10% of the user population accesses spam URLs immediately, and the remaining distribution is smooth without any distinct modes. The distributions for all users and users who "convert" are roughly similar, suggesting little correlation between time-to-click and whether a user visiting a site will convert. While most user visits occur within the first 24 h, 10% of times-to-click are a week to a month, indicating that advertised sites need to be available for long durations to capture full revenue potential.

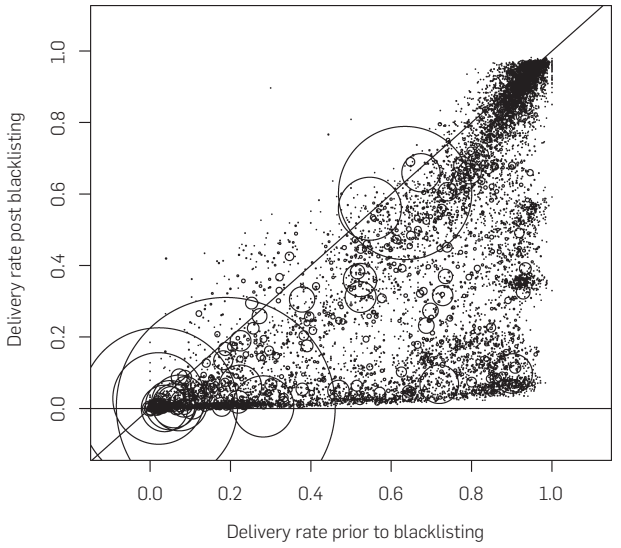
6. EFFECTS OF BLACKLISTING

A major effect on the efficacy of spam delivery is the employment by numerous ISPs of address-based blacklisting to reject email from hosts previously reported as sourcing spam. To assess the impact of blacklisting, during the course of our experiments we monitored the *Composite Blocking List* (CBL),[6] a blacklist source used by the operators of some of our institutions. At any given time the CBL lists on the order of 4–6 million IP addresses that have sent email to various spamtraps. We were able to monitor the CBL from March 21–April 2, 2008, from the start of the pharmacy campaign until the end of the April Fool campaign.

We downloaded the current CBL blacklist every half hour, enabling us to determine which worker bots in our measurements were present on the list and how their arrival on the list related to their botnet activity. Of 40,864 workers that sent delivery reports, fully 81% appeared on the CBL. Of those appearing at some point on the list, 77% were on the list prior to our observing their receipt of spamming directives, appearing first on the list 4.4 days (median) earlier. Of those not initially listed but then listed subsequently, the median interval until listing was 1.5 h, strongly suggesting that the spamming activity we observed them being instructed to conduct quickly led to their detection and blacklisting. Of hosts never appearing on the list, more than 75% never reported successful delivery of spam, indicating that the reason for their lack of listing was simply their inability to effectively annoy anyone.

We would expect that the impact of blacklisting on spam delivery strongly depends on the domain targeted in a given email, since some domains incorporate blacklist feeds such as the CBL into their mailer operations and others do not. To explore this effect, Figure 7 plots the per-domain delivery rate: the number of spam emails that workers reported as successfully delivered to the domain divided by number attempted to that domain. The *x*-axis shows the delivery rate for spams sent by a worker prior to its appearance in the CBL, and the *y*-axis shows the rate after its appearance in the CBL. We limit the plot to the 10,879 domains to which workers attempted to deliver at least 1,000 spams. We plot

Figure 7. Change in per-domain delivery rates as seen prior to a worker bot appearing in the blacklist (*x*-axis) vs. after appearing (*y*-axis). Each circle represents a domain targeted by at least 1,000 analyzable deliveries, with the radius scaled in proportion to the number of delivery attempts.



delivery rates for the two different campaigns as separate circles, though the overall nature of the plot does not change between them. The radius of each plotted circle scales in proportion to the number of delivery attempts, the largest corresponding to domains such as hotmail.com, yahoo.com, and gmail.com.

From the plot we clearly see a range of blacklisting behavior by different domains. Some employ other effective antispam filtering, indicated by their appearance near the origin—spam did not get through even prior to appearing on the CBL blacklist. Some make heavy use of either the CBL or a similar list (*y*-axis near zero, but *x*-axis greater than zero), while others appear insensitive to blacklisting (those lying on the diagonal). Since points lie predominantly below the diagonal, we see that either blacklisting or some other effect related to sustained spamming activity (e.g., learning content signatures) diminishes the delivery rate seen at most domains. Delisting followed by relisting may account for some of the spread of points seen here; those few points above the diagonal may simply be due to statistical fluctuations. Finally, the cloud of points to the upper right indicates a large number of domains that are not targeted much individually, but collectively comprise a significant population that appears to employ no effective antispam measures.

7. CONVERSION ANALYSIS

We now turn to a preliminary look at possible factors influencing response to spam. For the present, we confine our analysis to coarse-grained effects.

We start by mapping the geographic distribution of the hosts that "convert" on the spam campaigns we monitored. Figure 8 maps the locations of the 541 hosts that execute the emulated self-propagation program, and the 28 hosts that visit the purchase page of the emulated pharmacy site. The map shows that users around the world respond to spam.

Figure 8. Geographic locations of the hosts that "convert" on spam: the 541 hosts that execute the emulated self-propagation program (light gray), and the 28 hosts that visit the purchase page of the emulated pharmacy site (black).

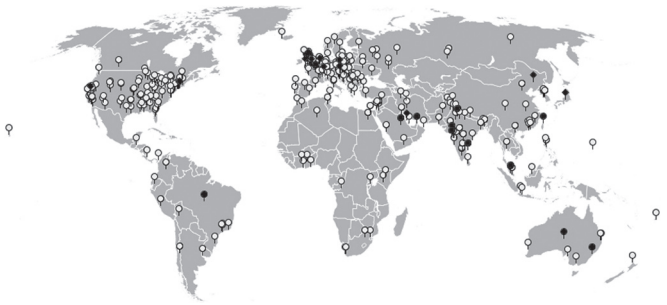
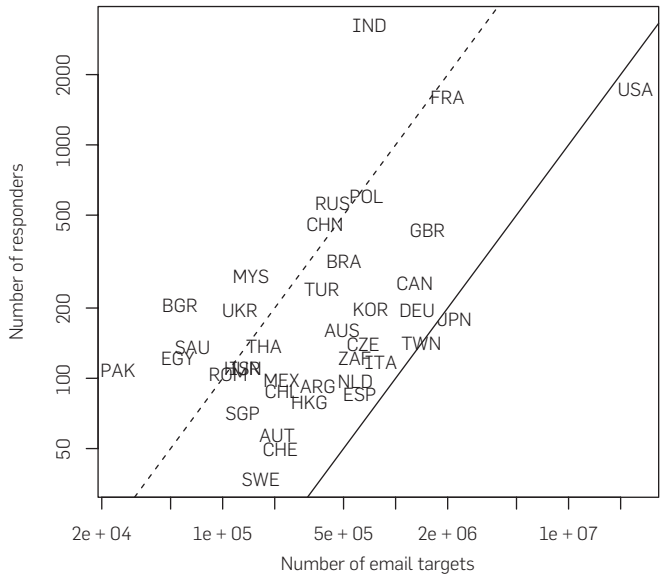


Figure 9 looks at differences in response rates among nations as determined by prevalent country-code email domain TLDs. To allow the inclusion of generic TLDs such as .com, for each email address we consider it a member of the country hosting its mail server; we remove domains that resolve to multiple countries, categorizing them as "international" domains. The *x*-axis shows the volume of email (log-scaled) targeting a given country, while the *y*-axis gives the number of responses recorded at our Web servers (also log-scaled), corresponding to Stages A and D in the pipeline (Figure 5), respectively. The solid line reflects a response rate of 10^{-4} and the dashed line a rate of 10^{-3}. Not surprisingly, we see that the spam campaigns target email addresses in the United States substantially more than any other

Figure 9. Volume of email targeting (*x*-axis) vs. responses (*y*-axis) for the most prominent country-code TLDs. The *x* and *y* axes correspond to Stages A and D in the pipeline (Figure 5), respectively.



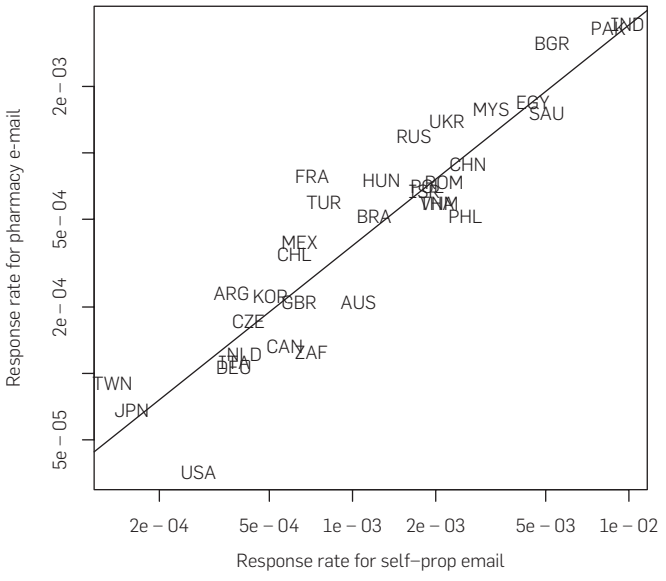
country. Further, India, France, and the United States dominate responses. In terms of response *rates*, however, India, Pakistan, and Bulgaria have the highest response rates than any other countries (furthest away from the diagonal). The United States, although a dominant target and responder, has the lowest resulting response rate of any country, followed by Japan and Taiwan.

However, the countries with predominant response rates do not appear to reflect a heightened interest in users from those countries in the specific spam offerings. Figure 10 plots the rates for the most prominent countries responding to self-propagation vs. pharmacy spams. The median ratio between these two rates is 0.38 (diagonal line). We see that India and Pakistan in fact exhibit almost exactly this ratio (upper-right corner), and Bulgaria is not far from it. Indeed, only a few TLDs exhibit significantly different ratios, including the United States and France, the two countries other than India with a high number of responders; users in the United States respond to the self-propagation spam substantially more than pharmaceutical spam and vice versa with users in France. These results suggest that, for the most part, per-country differences in response rate are due to *structural* causes (quality of spam filtering, user education) rather than differing degrees of cultural or national interest in the particular promises or products conveyed by the spam.

8. CONCLUSION

This paper describes what we believe is the first large-scale quantitative study of spam conversion. We developed a methodology that uses botnet infiltration to indirectly instrument spam emails such that user clicks on these messages are taken to replica Web sites under our control. Using this methodology we instrumented almost 500 million spam messages, comprising three major campaigns, and quantitatively characterized both the delivery process and the conversion rate.

Figure 10. Response rates (stage D in the pipeline) by TLD for executable download (*x*-axis) vs. pharmacy visits (*y*-axis).



We would be the first to admit that these results represent a single data point and are not necessarily representative of spam as a whole. Different campaigns, using different tactics and marketing different products will undoubtedly produce different outcomes. Indeed, we caution *strongly* against researchers using the conversion rates we have measured for these Storm-based campaigns to justify assumptions in any other context. At the same time, it is tempting to speculate on what the numbers we have measured might *mean*. We succumb to this temptation below, with the understanding that few of our speculations can be empirically validated at this time.

After 26 days, and almost 350 million email messages, only 28 sales resulted—a conversion rate of well under 0.00001%. Of these, all but one was for male-enhancement products and the average purchase price was close to $100. Taken together, these conversions would have resulted in revenues of $2,731.88—a bit over $100 a day for the measurement period or $140 per day for periods when the campaign was active. However, our study interposed on only a small fraction of the overall Storm network—we estimate roughly 1.5% based on the fraction of worker bots we proxy. Thus, the total daily revenue attributable to Storm's pharmacy campaign is likely closer to $7000 (or $9500 during periods of campaign activity). By the same logic, we estimate that Storm self-propagation campaigns can produce between 3500 and 8500 new bots per day.

Under the assumption that our measurements are representative over time (an admittedly dangerous assumption when dealing with such small samples), we can extrapolate that, were it sent continuously at the same rate, Storm-generated pharmaceutical spam would produce roughly 3.5 million dollars of revenue in a year. This number could be even higher if spam-advertised pharmacies experience repeat business, a bit less than "millions of dollars every day," but certainly a healthy enterprise.

The next obvious question is, "How much of this revenue is profit?" Here things are even murkier. First, we must consider how much of the gross revenue is actually recovered on a sale. Assuming the pharmacy campaign drives traffic to an affiliate program (and there are very strong anecdotal reasons to believe this is so) then the gross revenue is likely split between the affiliate and the program (an annual net revenue of $1.75 million using our previous estimate). Next, we must subtract business costs. These include a number of incidental expenses (domain registration, bullet-proof hosting fees, etc.) that are basically fixed sunk costs, and the cost to distribute the spam itself.

Anecdotal reports place the *retail* price of spam delivery at a bit under $80 per million.[14] In an examination we conducted of some spam-for-hire service advertisements, we found prices ranging from $70 to over $100 per million for delivery to US addresses, with substantial discounts available for large volumes. This cost is an order of magnitude less than what legitimate commercial mailers charge, but is still a significant overhead; sending 350M emails would cost more than $25,000. Indeed, given the net revenues we

estimate, retail spam delivery would only make sense if it were 20 times cheaper still.

And yet, Storm continues to distribute pharmacy spam—suggesting that it is in fact profitable. One explanation is that Storm's masters are vertically integrated and the purveyors of Storm's pharmacy spam are none other than the operators of Storm itself (i.e., that Storm does not deliver these spams for a third-part in exchange for a fee). There is some evidence for this, since the distribution of target email domain names between the self-propagation and pharmacy campaigns is virtually identical. Since the self-propagation campaigns fundamentally must be run by the botnet's owners, this suggests the purveyor of the pharmacy spam is one and the same. A similar observation can be made in the harvesting of email addresses from the local hard drives of Storm hosts. These email addresses subsequently appear in the target address lists of the pharmacy campaign and self-propagation campaigns alike. Moreover, neither of these behaviors is found in any of the other (smaller) campaigns distributed by Storm (suggesting that these may in fact be fee-for-service distribution arrangements). If true, then the cost of distribution is largely that of the labor used in the development and maintenance of the botnet software itself. While we are unable to provide any meaningful estimates of this cost (since we do not know which labor market Storm is developed in), we surmize that it is roughly the cost of two or three good programmers.

If true, this hypothesis is heartening since it suggests that the third-party *retail* market for spam distribution has not grown large or efficient enough to produce competitive pricing and thus, that profitable spam campaigns require organizations that can assemble complete "soup-to-nuts" teams. Put another way, the profit margin for spam (at least for this one pharmacy campaign) may be meager enough that spammers must be sensitive to the details of how their campaigns are run and are economically susceptible to new defenses.

Acknowledgments

This was one of the most complex measurement studies our group has ever conducted and would have been impossible without the contributions of a large and supportive cast. Here we offer our thanks for their insightful feedback and individual contributions to our effort.

Jordan Hayes provided decidedly nontrivial help with site domain registration. Peter Blair, Paul Karkas, Jamie Knight, and Garrick Lau at Tucows supported this activity (once we convinced them we were not spammers) and allowed us to use reputable registrars. Randy Bush provided overall guidance and help concerning Internet operations and policy issues while Erin Kenneally advised us on legal issues. Brian Kantor set up and managed our DNS, Web, and SMTP servers, while Scott Campbell and Stephen Chan performed massive DNS lookups for us. Jef Poskanzer provided data access for debugging our experiment, Stephen Chenette provided technical assistance and Fallon Chen was our in-house graphic designer. Bill Young and Gregory Ruiz-Ade set up target email accounts in UCSD's CSE department. Special thanks to Gabriel Lawrence and Jim Madden of UCSD's ACT for supporting this activity on UCSD's systems and networks. Finally, our thanks to the anonymous reviewers for their time and commentary.

This work was made possible by the National Science Foundation grants NSF-0433702 and NSF-0433668 and by generous research, operational and in-kind support from Cisco, Microsoft, HP, Intel, VMWare, ESnet, the Lawrence Berkeley National Laboratory, and UCSD's Center for Networked Systems. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors or originators and do not necessarily reflect the views of these organizations.

References

1. Akass, C. Storm worm 'making millions a day.' http://www.pcw.co.uk/personal-computer-world/news/2209293/strom-worm-making-millions-day, February 2008.
2. Anderson, D.S., Fleizach, C., Savage, S., Voelker, G.M. Spamscatter: Characterizing internet scam hosting infrastructure. In *Proceedings of the USENIX Security Symposium* (Boston, MA, August 2007).
3. Angwin, J. Elusive Spammer Sends EarthLink on Long Chase. http://online.wsj.com/article_email/SB10522559382372600.html, May 2003.
4. D. M. Association. DMA Releases 5th Annual 'Response Rate Trends Report.' http://www.the-dma.org/cgi/disppressrelease?article=1008, October 2007.
5. Boehme, R., Ho, T. The effect of stock spam on financial markets. In *Proceedings of the 5th Workshop on the Economics of Information Security (WEIS)* (June 2006).
6. Composite Blocking List (CBL). http://cbl.abuseat.org/, March 2008.
7. Frieder, L., Zittrain, J. Spam works: evidence from stock touts and corresponding market activity. *Berkman Center Research Publication*, 2006.
8. Goodman, J., Rounthwaite, R. Stopping outgoing spam. *Proceedings of the 5th ACM Conference on Electronic Commerce* (2004), 30–39.
9. Hanke, M., Hauser, F. On the effects of stock spam emails. *J. Financ. Mark. 11*, 1 (2008), 57–83.
10. Kirk, J. Former Spammer: 'I Know I'm Going to Hell.' http://www.macworld.com/article/58997/2007/07/spammer.html, July 2007.
11. Kreibich, C., Kanich, C., Levchenko, K., Enright, B., Voelker, G.M., Paxson, V., Savage, S. On the Spam Campaign Trail. In *First USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET'08)*, April 2008.
12. Judge, W.Y.P., Alperovitch, D. Understanding and Reversing the Profit Model of Spam. In *Workshop on Economics of Information Security 2005 (WEIS 2005)* (Boston, MA, USA, June 2005).
13. Watson, D. All Spammers Go to Hell (posting to funsec list). http://www.mail-archive.com/funsec%40linuxbox.org/msg03346.html, July 2007.
14. Wilson, T. Competition May Be Driving Surge in Botnets, Spam. http://www.darkreading.com/document.asp?doc_id=142690, 2008.

Chris Kanich, Kirill Levchenko, Brandon Enright, Geoffrey M. Voelker, and Stefan Savage ({ckanich,klevchen,voelker,savage}@cs.ucsd.edu bmenrigh@ucsd.edu), Department of Computer Science and Engineering University of California, San Diego.

Christian Kreibich and Vern Paxson (christian@icir.org, vern@cs.berkeley.edu), International Computer Science Institute Berkeley.