

# Spamology: A Study of Spam Origins

Craig A. Shue<sup>\*</sup>  
Oak Ridge National  
Laboratory  
Oak Ridge, Tennessee, USA  
shueca@ornl.gov

Minaxi Gupta  
Computer Science Dept.  
Indiana University  
Bloomington, Indiana, USA  
minaxi@cs.indiana.edu

Chin Hua Kong  
Computer Science Dept.  
Indiana University  
Bloomington, Indiana, USA  
kongch@cs.indiana.edu

John T. Lubia  
Computer Science Dept.  
Indiana University  
Bloomington, Indiana, USA  
jtlubia@cs.indiana.edu

Asim S. Yuksel  
Computer Science Dept.  
Indiana University  
Bloomington, Indiana, USA  
asyuksel@cs.indiana.edu

## ABSTRACT

The rise of spam in the last decade has been staggering, with the rate of spam exceeding that of legitimate email. While conjectures exist on how spammers gain access to email addresses to spam, most work in the area of spam containment has either focused on better spam filtering methodologies or on understanding the botnets commonly used to send spam. In this paper, we aim to understand the origins of spam. We post dedicated email addresses to record how and where spammers go to obtain email addresses. We find that posting an email address on public web pages yields immediate and high-volume spam. Surprisingly, even simple email obfuscation approaches are still sufficient today to prevent spammers from harvesting emails. We also find that attempts to find *open relays* continue to be popular among spammers. The insights we gain on the use of web crawlers used to harvest email addresses and the commonalities of techniques used by spammers open the door for radically different follow-up work on spam containment and even systematic enforcement of spam legislation at a large scale.

## 1. INTRODUCTION

Spam exceeds legitimate email today. It wastes bandwidth and storage, delays valid emails, and hurts human productivity. Given these implications, significant efforts have rightly been devoted to spam containment. While we have made great leaps in developing novel spam-filtering techniques and in understanding the botnets typically used to send spam, our understanding of where spammers get the email addresses to spam is largely confined to “they buy bulk

<sup>\*</sup>Craig Shue performed this work while a Ph.D. candidate at Indiana University.

Final edits of the submitted manuscript have been authored by a contractor of the U.S. Government under contract DE-AC05-00OR22725. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.

email lists for very little money.” This paper is motivated by the question: How are these bulk email lists created? While previous work has explored one avenue for how spammers might be harvesting email addresses [1], our goal is to undertake a systematic study to understand the phenomenon.

Our general approach is to register a dedicated domain, run an email server for that domain, and post email addresses belonging to the domain at strategic places and watch the inflow of spam. The first step we take is to register for free offers on the Web and give each web site a unique email address. This is motivated by the simple observation that Internet users are frequently asked to trust and confide in a variety of web sites. Many sites offer their services, such as email accounts, and news for free. However, to access these services, users must register and provide some personal information. This often includes an email address, allowing the site to contact the user. While users may implicitly assume that the site will securely and responsibly handle the confidential registration information, this assumption may not hold in practice. Our next step is to post unique, site-specific email addresses at popular sites with a blog or a comment section, such as [washingtonpost.com](http://washingtonpost.com) and [nytimes.com](http://nytimes.com). The intent behind this step is to see if the posted email addresses get discovered by web crawlers that eventually lead to spam. To compare the likelihood of getting email addresses harvested from popular sites versus less popular ones, and to understand the behavior of crawlers that lead to spam, we also post multiple email addresses on various web pages belonging to our department.

We exposed 22,230 unique email addresses and monitored the inflow of spam for a period of almost 5 months. The resulting data helped us draw various interesting conclusions and also helped confirm some prevailing wisdom. Specifically, we concluded that:

- **Users must exercise caution when divulging their email addresses.** Though none of the popular web sites where we registered for accounts spammed, a few of the less popular ones did send spam. Publicly posting email addresses at both popular and unpopular sites led to the most spam.
- **Spam arrives instantly.** Our first spam arrived in less than one hour, indicating that new email addresses

get discovered quickly on the Internet.

- **Commonly-used email obfuscation techniques are offering protection (for now).** It is common practice to replace the conventional @ in email addresses by an AT in order to defeat email harvesting. We found that the spammers are still not parsing simple obfuscations as of now. However, one should not count on the protection offered by such simple obfuscation schemes, for they are trivial to defeat.
- **Spamming crawlers exist and can be tracked.** Our scheme of dynamically presenting each new visitor to our department web pages with a new email address allowed us to track which crawlers, if any, led to spam. We found that crawling for harvesting email address is occurring regularly. Thus, blocking access for spamming crawlers could be an important new step in spam containment. This confirms findings from the work in Project Honey Pot [1].
- **Top spammers use multiple email-harvesting strategies.** Top spammers crawled popular and unpopular web pages to harvest email addresses. They also made attempts to check if our mail server could be used as an *open relay*. On the one hand, it shows the aggressiveness of spammers. On the other, it opens up new avenues for fighting spam because anybody who is doing both could be filtered.

**Roadmap:** The remainder of this paper is organized as follows: We explain our data collection technique in Section 2. In Section 3, we provide an overview of the spam we collect and in Section 4, we analyze how quickly spam is received once an email address is exposed. Section 5 examines spamming trends at the domain and IP granularity and Section 6 investigates the behavior of web crawlers that are tied to spam campaigns. We review related work in Section 7 and discuss the implications of this study on spam containment in Section 8.

## 2. DATA COLLECTION METHODOLOGY

Spammers can harvest email addresses from multiple sources on the Internet, including web pages, blogs, newsgroups, social networking sites, and mailing lists. In our study, we examine the ramifications of making email addresses available at a variety of locations on the Internet. Before we could post email addresses to spammers, we had to take several steps. We began by registering a dedicated domain for this project, which we hosted on servers in our department. We then added a mail server record to our domain’s DNS file. To set up this mail server, we used the `qpsmtpd` daemon [2], which is written in Perl and is designed to be easily customized. We set up the mail server to write a copy of each email we received to a file, including the *envelope* headers from the SMTP RCPT TO and MAIL FROM commands. This allowed us to easily examine the body of the email message, determine the host name and IP address of the sending mail server, and record the time the mail was sent.

To observe where spammers harvest emails from, we used both active and passive approaches. In the active approach, we signed up for mailing lists and free newsletters, and posted email addresses at blog sites and other departmental

web sites in a manner that were available to web crawlers. Our general approach was to create a new email address to provide in each case. This allowed us to track the outcome on a per-site basis. In the passive approach, we simply observed the unsolicited spam sent to our mail server. We now describe each in detail.

**1. Signing up for mailing lists and newsletters:** Our first data set was motivated by the question “How often do web sites leak email addresses provided to them?” To collect data to answer this question, we signed up and created accounts at 70 different types of web sites, including those that promise free offers and mailing lists and newsletters. We provided a unique email address belonging to our registered domain to each site in order to track which of these sites lead to spam. We used two different ways to find these sites. First, we used the Alexa Web Information Service [3] to find popular web sites which allowed creation of accounts. 14 of the 70 accounts we created were in Alexa’s top 1000 web sites according to their traffic rankings. Second, we searched for pages that had forms that asked for email addresses. This led to the creation of 56 additional accounts. Combined, these 70 accounts helped us understand the behavior of popular web sites in comparison to the unpopular ones. We refer to this data set as the ACT (standing for *account*) data set throughout the rest of this paper.

**2. Posting emails on popular web sites:** The second data set was motivated by the question “What is the likelihood of receiving spam on an email address posted at a popular site?” To collect data to answer this question, we posted 9 unique email addresses on 4 popular web sites. 3 of these sites were in the Alexa top 1000 popular sites. The email addresses were often added in comment sections of news stories or in blogs. To avoid removal of the email addresses by site moderators, yet still avoid legitimate emails, we included innocuous commentary along with each email address. Since our comments are unlikely to garner responses from regular visitors to these sites, we consider any email sent to these 9 addresses to be spam. Subsequently, this data set is denoted as the PST (standing for *post*) data set.

**3. Posting static email addresses on less popular web pages:** This data set was motivated by the question: “What is the likelihood of receiving spam on an email address at a less popular web site?” To collect this data set, we posted two different email addresses on two departmental course web pages. These email addresses were included in HTML comments on the page to hide them from casual web browsers. However, web crawlers parsing the page may not distinguish commented regions from others and still harvest these email addresses. We took this step to ensure that all emails to these two email addresses are spam. To see if crawlers take the time to get around a commonly used technique to obfuscate email addresses, we posted one email address with an `at` instead of the expected `@`. Thus, one of the email address was in the expected `@` notation (e.g., `static@iucsnrg.com` and the other one with the word `at` separating the user and domain (e.g., `static2 at iucsnrg.com`). We subsequently denote this data set as the STA (standing for *static* email address) data set.

**4. Posting dynamic email addresses on less popular web pages:** This data set was motivated by the same high-level question as the STA data set but was designed

to further understand the behavior of crawlers that harvest email addresses. Specifically, we were interested in understanding how often crawlers visit and which of them lead to spam. Toward this goal, we performed more fine-grained monitoring on our research group web site, which supports server-side scripting languages, such as PHP. Each time two of our tracked pages were loaded, we used a PHP script to randomly generate two new email addresses and embed them in a HTML comment on the page. Since each address was uniquely shown to only one user, this allowed us to record the IP address of the crawler and the time at which the page was crawled, which we then associate with the email address during the analysis. The two generated email addresses on the pages differed only in that one contained the expected @ as a separator between user name and domain and the other contained `at` with spaces on either side. Since these email addresses were dynamically generated, we subsequently denote this data set as the `DYN` data set.

**5. Relay spam:** Upon finding a new registered domain, spammers may attempt to send emails to common first or last names at the domain (e.g., `bob@example.com`). To detect these dictionary-based campaigns, we simply record all email destined to our domain and determine whether it matches any common names. In other cases, spammers may simply look for machines operating email servers and attempt to use them as relays to hide their origins. If a mail server accepts email not destined to its own domain, it is generally referred to as an *open relay*. Open relays have been known to be exploited by spammers [4]. Our domain’s mail server is configured to accept any relayed messages, but instead of delivering the spam messages, it records them to a file so they can be analyzed. We examined the `RCPT TO` field in the spam header toward this goal. If it did not contain our domain, we determined that the sender attempted to use our mail server as an open relay. We subsequently refer to all passively observed spam as the `RAW` data set.

### 3. OVERVIEW OF COLLECTED SPAM

We received a total of 4,033 emails over a period of five months. Of these, 3,475 belonged to the `ACT` data set alone and needed to be examined carefully before being labeled as spam because we authorized the web sites in the `ACT` data sets to send us emails by signing up voluntarily for their offerings, such as newsletters and deals. Thus, emails sent on behalf of these web sites should be considered legitimate.

In order to distinguish legitimate emails in the `ACT` data set from spam, we used SpamAssassin [5], which is a widely-used filter to identify spam. SpamAssassin is designed to be used on a mail server to filter email before it reaches the user’s mailbox. Since we save our emails to files, we simply ran the saved messages through SpamAssassin offline to score them. SpamAssassin uses advanced statistical methods to classify emails. The tests target email headers, email body, attachments, and URLs in the content of email. It assigns scores to each of the rules it uses to distinguish spam from good emails. If the sum of the scores for individual rules meets the set threshold value for an email, it considers it to be spam. We used the default rules and started using the default threshold score of 5.0. We then used the threshold to tune the false negatives and false positives. To decide on an appropriate threshold for the `ACT` data set, we ran SpamAssassin on the `PST` data set which is all spam.

Based on the false negatives produced by SpamAssassin on this data set, we decided to use a threshold of 3.5 for the `ACT` data set.

We encountered a glitch during data collection which caused us to lose email bodies for a period of 53 days. We had complete email headers for this duration which were sufficient for all other analysis conducted in this paper but not for identification of spam emails in `ACT` data set since SpamAssassin needs full email bodies. Hence, we subjected only the 1,857 emails which had full body information to SpamAssassin. 46 of those were labeled as spam. All emails in the rest of the data sets are spam by design.

Table 1 presents an overview of the spam received within each of the five data sets. It also shows the number of email addresses advertised in each data set and the average spam received for email addresses that were spammed. *Clearly, posting email addresses on popular web sites fetched the most spam but even email addresses advertised at less popular sites led to spam. A few of the web sites where we voluntarily signed up for accounts also sent spam.*

Start date: 12th September, 2008					
Duration: 147 days					
	ACT	PST	STA	DYN	RAW
Web sites	70	4	1	2	0
Email addresses advertised	70	8	2	22,150	0
Emails received	1,875	237	23	96	202
Spam emails	46	237	23	96	202

**Table 1: Overview of data collection. Notice that the unique emails circulated in the case of `DYN` data set is the same as the number of crawler visits, since each visit generates a new email address in the comment portion of the HTML of the displayed page.**

### 3.1 Spam Categories

In order to get an idea of the types of spam we received, we used the MeURLin project [6], which performs web page classification based only on its URL. It segments a URL into smaller, meaningful pieces, adds URL components and orthographic features to model prominent patterns, and uses this information to classify the URLs [7]. This approach is faster than typical web page classification since it does not analyze page content in order to classify. We subjected URLs contained in our spam emails to MeURLin in order to classify spam. Figure 1 shows the result of this classification on the 4,052 URLs contained in the body of our spam. Incidentally, the `RAW` data set, where spam attempted to use our mail server as an open relay, did not contain any URLs. Spam related to shopping, games, and computers formed the top 3 categories. Authors in [8] also found these to be the dominant spam categories in their work.

### 3.2 Spam on Obfuscated Emails

The email addresses in `STA` and `DYN` data sets were advertised in two formats: one in the usual format, where an @ separates the user name from the domain name and another where an `at` separates the two. The intention was to see if this commonly used obfuscation technique offers any protection. *To our surprise, none of the crawlers that visited our departmental research and course and research web pages led to any spam on email addresses containing the `at`.* This indicates that simple email obfuscation techniques are offering protection at the moment. This may simply be because

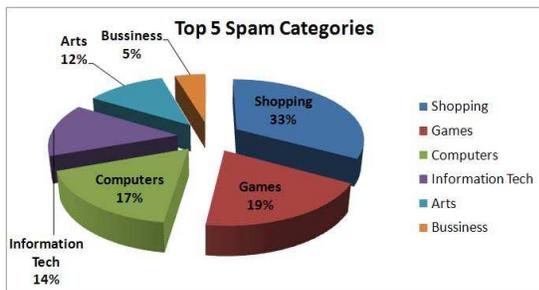


Figure 1: Spam categories

enough emails can be harvested in the expected format that there is no need for spammers to parse additional formats.

#### 4. ONSET OF SPAM

We started signing up for accounts (ACT data set) and posting email addresses (PST, STA, DYN data sets) starting September 12th, 2008, immediately after setting up the mail server for the domain we registered for the project. Not all the email addresses we advertised received spam. We show the arrival statistics for each of the data sets in Table 2. *Of the 70 email addresses we advertised through account registrations (the ACT data set), four received spam. The first spam arrived within 50 minutes of registering for an account.* Fortunately, none of the spammed email addresses were given to web sites belonging to top Alexa web sites, implying that popular web sites tend not to send spam or sell email addresses given to them.

	ACT	PST	STA	DYN	RAW
Email addresses advertised	70	8	2	22,150	0
Email addresses spammed	4	4	1	55	n/a
Average spam per spammed email	15.33	59.25	23	1.7	n/a
First spam	50 min	18.5 hours	84 days	3 days	4 days

Table 2: Arrival of first spam in each data set

Posting email addresses at blogs and comments sections of popular web sites (the PST data set) led to the most spam. 4 of the 8 email addresses there received spam, indicating that crawlers that fish for email addresses are aggressive about crawling popular web sites. Also, the first spam arrived within a day of posting the email address. *This implies that posting email addresses on popular web sites increases exposure to spam more than any of the other places we investigated.*

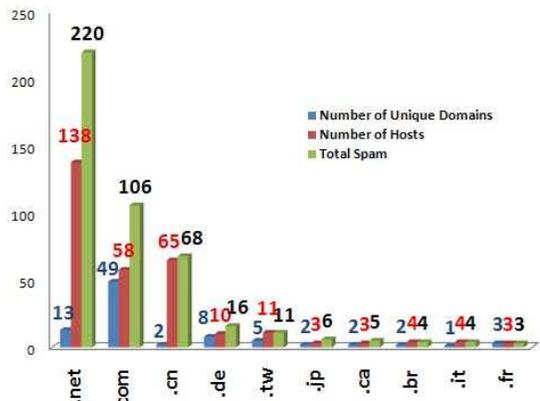
Email addresses on departmental pages also fetched spam (the STA and DYN data sets), albeit it took longer for the spammers to find those addresses. While we would have expected the email addresses on both pages to be spammed around the same time, it took much longer for the web page under the STA data set to be discovered. Nonetheless, it did receive spam after 84 days when the DYN data set was spammed within 3 days.

The first spam in the RAW data set arrived within 4 days of setting up the mail server for the account. *Thus, new mail servers are quickly discovered by spammers.* Further, none of the spam emails in this data set were targeted to our domain.

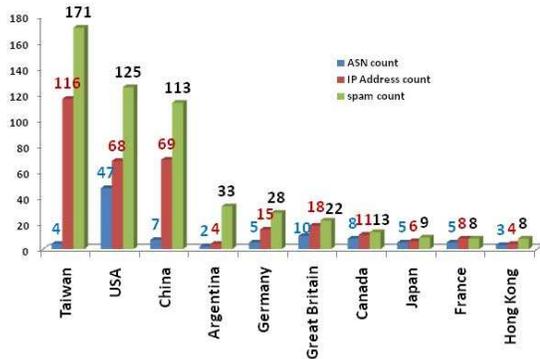
Each spam email was an attempt to relay spam through our mail server, implying that spammers are constantly on the look out for open relays.

#### 5. SPAMMING DOMAINS & IP ADDRESSES

Spam is a truly global phenomenon. We collected a total of 603 spam emails across our five data sets. These emails came from 331 host names belonging to 129 domains and 31 top-level domains (TLDs). The spamming machines had 384 unique IP addresses belonging to 142 autonomous systems (ASes) that were spread over 35 countries.



(a) Top-10 spamming TLDs, their domains, and their hosts



(b) Top-10 spamming countries, their IP addresses, and their ASNs

Figure 2: Top-10 spammers

Figure 2 shows the top-10 spammers from two perspectives. Figure 2(a) shows the spam from the top-10 spamming TLDs and the number of domains and hosts that were involved in sending spam. This Figure has a few interesting aspects. First, even though the .com TLD accounts for almost half the domains in the Internet, .net TLD sends more spam. This is due to a particular spamming domain, `hinet.net`, which contributed to 28% of all the spam in our data sets. Another noteworthy observation is that only the .net and .cn TLDs have a high ratio of spam emails to domains sending spam. In case of .net, it is due to `hinet.net` and in case of .cn, it is due to the domain that contributed the second highest amount of spam (10%), `163data.com.cn`.

Figure 2(b) shows the spam from the top-10 spamming countries and the number of autonomous system numbers

(ASNs) and IP addresses that were involved in sending spam. Taiwan, USA, and China contributed the most spam in our data. In fact, the top spamming domains, `hinet.net` and `163data.com.cn`, were registered in Taiwan and China respectively.

	ACT	PST	STA	DYN	RAW
Unique sending domains	10	56	13	38	12
Average spam per domain	4.7	4.23	1.76	2.52	16.66
Unique sending IPs	11	164	23	55	133
Average spam per IP	4.27	1.44	1	1.74	1.5

Table 3: Spamming domains and IP addresses

We now look at spamming domains and IPs across the different data sets. Table 3 shows unique spamming domains and IP addresses for each data set. *A key observation is that average spam per domain and IP address is low across all data sets, indicating that blacklisting based on spammer IP addresses and domains may not be fruitful.*

### 5.1 Top Spamming Domains

Figure 3 shows the spamming behavior of the top spamming domain, `hinet.net`. It sent a total of 169 emails (28% of all spam) using 113 IP addresses and was active throughout our data collection period. It appears that `hinet.net` is a Taiwanese gateway site that provides news, blogs, and other services.

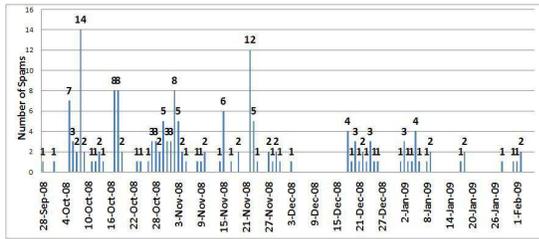


Figure 3: Number of spam per day for the top spamming domain

A closer examination of `hinet.net` led to several interesting observations. First, a vast majority of `hinet.net`'s spam was directed to the RAW data set. Specifically, 163 of the spam emails from this domain were sent to the RAW data set and contained no email bodies, implying that most of them were attempts to find *open relays*. The rest of the 6 spam emails from `hinet.net` showed up in the PST data set, implying that it crawled the popular blogs to find email addresses to spam. While `hinet.net` did not send spam to any other data sets, it crawled the web sites in our DYN data set 30 times! In all of these activities, `hinet.net` used IP addresses from many different prefix ranges, all belonging to its own domain. We verified this by mapping the IP addresses in the various data sets to corresponding ASNs using the Cymru ASN lookup service [9], using *whois* lookups to find out the owners of the ASNs, and then looking up BGP prefixes announced by the various ASNs using BGP routing tables [10]. *Clearly, hinet.net actively tries to find open relays to send spam and also crawls the web to find email addresses to spam. This analysis strongly points to blocking this domain and its IP address range from crawling or sending spam.*

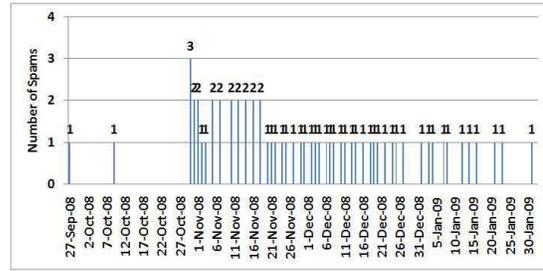


Figure 4: Number of spam per day for second biggest spamming domain

To see if `hinet.net`'s behavior is unique or typical, we look at the next biggest spamming domain in our data. Figure 4 shows the activity of the next biggest spamming domain, `163data.com.cn`. It sent a total of 61 emails (10% of all spam) using 61 different IP addresses and was active during most of our data collection period. `163data.com.cn` shared characteristics with `hinet.net` in that it attempted to relay spam through our mail server and spammed an email address we posted at a popular blog site. However, unlike `hinet.net`, which only crawled our pages in the DYN data set but did not send any spam, `163data.com.cn` crawled the DYN pages 267 times and also sent spam (just one, however). The two domains differ in that while much of `hinet.net`'s activity was focused around relaying spam through our mail server, 58 of the 61 spam sent by `163data.com.cn` were a result of posting our email address at a popular blog site. Incidentally, `163data.com.cn` was also reported as serving malware and running command and control servers for controlling botnets in a recent study that investigated cyber espionage against Tibetan institutions [11].

The two top spammers shared another interesting property – they both used their own IP addresses to send spam. This is somewhat counter-intuitive since many works have pointed to the use of botnets to send spam [12, 13, 14]. We conjecture that the spammers in our data are mostly using their own machines for three purposes: 1) to find open relays, 2) to find new email addresses to spam, and 3) to test the fruits of their labor. *We conclude that many spammers are employing common strategies, including relay attempts and web crawling, to find spamming targets. Further, since these activities require continuous monitoring, they are likely to use IP addresses belonging to their own prefixes in these endeavors. This opens up new avenues spam containment, as we discuss in Section 8.*

## 6. WEB CRAWLERS AND SPAM

To obtain the email addresses in our STA, DYN, and PST data sets, spammers must use web crawlers to access the web pages and then extract the email addresses on each page. While the PST data set emails are located on third-party servers, the STA and DYN data sets are hosted on web servers we operate, allowing us to examine the behavior of the web crawlers the spammers use. In particular, since the DYN data set provides a unique email address to each machine that accesses the page, we can use it to perform detailed analysis. In this Section, we examine the web crawlers used by spammers in the DYN data set at the IP, Autonomous System Number (ASN), and country granularity. We further

examine whether the web crawling infrastructure used by spammers is also used to send the spam emails themselves.

## 6.1 Overview of Spamming Crawlers

Each time a client accesses one of the web pages providing email addresses in the DYN data set, we record the client’s IP address, email address we provided, and the time the page was accessed. We then examined which of these email addresses later received spam. From this, we can determine which web crawlers accessed our web pages, harvested an email address, and provided it to spammers. To learn more about our crawlers, we used the Cymru IP to ASN lookup tool [9], which allowed us to also examine crawlers from an ASN granularity and determine its country.

In Table 4, we provide statistics about the number of clients that accessed our pages, at the IP and ASN granularity, and the number of times they downloaded the pages. Note that only about 3.23% of the unique clients that visited our DYN pages led to spam. They originated from a small number of ASNs and revisited the web pages an average of 11.62 times, indicating that they are likely to be crawlers. We refer to them as *spamming crawlers* subsequently. We infer that *spamming crawlers regularly revisit pages to detect new email addresses*.

	Led to spam	No Spam	Total
Unique IP Addresses	60	1,486	1,546
ASNs	14	378	392
Page Downloads	697	21,453	22,150

Table 4: Web crawler statistics

When we examined the crawlers that led to the most spam, we found that most of them are the sole spamming crawler in that ASN. However, one ASN from China had four different hosts involved in crawling to find email address to spam while all other ASNs had only one top spamming web crawler. Of the top 15 spamming web crawlers, 7 were from China, 3 were in the US, and Argentina, Brazil, Canada, Germany, and the United Kingdom each had one. However, the volume of spam resulting from each of these harvested addresses was a different story. The one web crawler from the UK resulted in the most emails at 27. Upon manual inspection, all of these emails seemed to be related to a financial scam. The Argentinian web crawler resulted in 21 spam mails. The three US web crawlers distributed a combined 26 spam messages while the 7 Chinese crawlers resulted in only 13 spam messages. In Figure 5, we plot the geographical locations of the good and bad crawlers and the mail servers used by spammers. From this, we see that good web crawlers are widely distributed, as are the mail servers used by spammers. However, we note that the malicious web crawlers tend to be more tightly clustered.

We further note that there was no overlap between the ASNs used by crawlers that led to spam and the ASNs used by legitimate visitors to our sites. This indicates that it may be feasible to block a small number of ASNs associated with spammer web crawlers to eliminate the harvesting of email addresses on a site.

## 6.2 Relationship between Crawlers and Spamming Mail Servers

Given that we know that some web crawlers led to spam, we now look to see if we can establish relationships between



Figure 5: Geolocation of Good Crawlers (G), Bad Crawlers (B), and Spamming Mail Servers (S)

the crawling machines and the mail servers used to transmit the actual spam messages.

We first look at whether the same machine performs the web crawling and operates the spam mail server. In four of the 60 cases these were functions performed on the same machine. We next look at whether the web crawler and spam mail server were located in the same ASN. We find 10 of these relationships in 3 ASNs. The ASN with the most of these pairings had 6 instances while the other two ASNs had two of these relationships. In all but one of these cases, the spam volume was only a single message with the exception sending 3 spam mails. From this, we see that web crawlers and mail sending servers tend to be in the same ASN, but these functions are often performed on separate machines.

Finally, we note that one special web crawler coordinated with 6 different spam mail servers located in the same /24 IP prefix, but different from that of the web crawler, to send 14 spam messages. However, this crawler only coordinated with one other mail server from a different prefix and ASN. This type of setup with a particular prefix may indicate an explicit decision by the spam operator to divide crawling and spam serving infrastructure or that they have agreements to pass on the harvested email addresses to partners.

## 6.3 Aggressiveness of Spamming Crawlers

Next, we looked at the number of times a crawling visited at its most frequent point. We note that good web crawlers rate limited their visits, averaging at most two visits per minute. However, crawlers tied to spamming were much more aggressive in their crawling, with the two most active crawlers visiting over 50 times in a single minute. This indicates that aggressiveness may itself be an important criterion to identify spamming crawlers.

Since spammers’ web crawlers regularly revisit pages, we examined whether how many of the email addresses we provided to these crawlers resulted in spam. The spamming web crawler that visited us most regularly, a total of 142 times, received a unique email address each visit. However, only 16 of these email addresses received spam after the crawler visited. The email addresses are located in exactly the same place in the page and use the same format each time the page is loaded. Accordingly, the web crawler was likely to be successful at harvesting each email address. While we cannot be sure why only some of the email addresses were spammed, it may be that the crawling infrastructure is shared across spam campaigns, with some campaigns having a shorter delivery time line than others.

## 7. RELATED WORK

In an early study by Cranor and LaMacchia [15], the authors point out the risk of spam email and analyze spam in terms of categorizing spam content. Since this study, spam has eclipsed legitimate mail. In a recent work by Goodman, Cormack and Heckerman [16], the authors discuss the state of spam and anti-spam technologies and techniques. In the work by Ramzan and Wüest [17], the authors examine the behavior of phishing campaigns during 2006. They analyze fluctuations in phishing emails and their targeted populations. With the high profile of spam and the economic impact associated with it, spam has been the subject of significant research. Here, we focus on how spamming infrastructure, analysis of the spam economy, and anti-spam efforts designed to combat the growth of spam.

### 7.1 Spamming Infrastructure

Prince *et al.* [1] use email honey pots to correlate web crawling clients with spam campaigns. We confirm some of these findings using generated addresses in a single domain, but use additional data sets to determine how spammers find addresses. Anderson *et al.* [8] develop a technique called SpamScatter to identify the infrastructure used in email scams. They characterize scam infrastructure to understand the spammer behavior. They extract the URLs from spam emails and used a commercial web content filtering software to categorize the spam. We did not have access to this filtering product for our own study; instead, we use the URL-based classification of web pages described in the work by Kan and Thi [7]. Xie *et al.* [12] designed a signature generation system that can detect botnet-based spam emails. The authors use this to detect new botnets and trends in obfuscation approaches used in spam emails. In the work by Zhao *et al.* [13], the authors used graph algorithms to detect web provider email accounts used by botnets to send spam. This analysis helped identify accounts registered by bots during an interval when CAPTCHAs were subverted, allowing automatic bot registration. In the work by Konte *et al.* [14], the authors examine the infrastructure used by phishers and other scamsters. They examine the DNS entries from host names in URLs contained in spam emails and find high turn-over among the hosting machines. This phenomenon, called fast flux, is used by phishers to provide resilient scam sites even when the scam hosting infrastructure uses compromised machines. In the work by McGrath *et al.* [18], the authors find that even the DNS servers used by these spam domains flux, indicating that the DNS infrastructure for the domains use compromised machines. Our work complements each of these by providing a means to detect the spam infrastructure used to harvest target email addresses. This infrastructure is separate from the spam mail sending or phishing site hosting infrastructure. Accordingly, our work can detect and expose additional compromised machines used for spamming.

### 7.2 Spam Economy

Spam email transmission has formed its own economy with a business model. In the work by Kanich *et al.* [19], the authors explore this economy and attempt to determine the amount of money made by spammers. To do so, they infiltrate a large botnet and determine the volume of responses to spam emails and the rate at which these emails are converted into sales. In the work by Wang *et al.* [20],

the authors track spam campaigns that exploit redirects to improve search engine rankings and identify advertisers that display ads on these spam web pages. Franklin *et al.* [21] analyze an underground online marketplace to determine the extent of profit associated with online scams. The authors conclude these markets are capable of stealing millions of dollars from victims in less than a year.

### 7.3 Anti-Spam Techniques

Approaches such as DomainKeys [22], Sender ID [23], and the Sender Policy Framework [24] attempt to reduce spam by providing origin authenticity. Each approaches uses cryptographic primitives to make it easier for a destination to confirm that only the indicated source sent the message. This can be used to prioritize legitimate senders while blocking spamming organizations. Other tools, such as SpamAssassin [5], provide email filtering functionality to separate spam email before it reaches the user's mailbox. Other works focus on spam from an end-user perspective. The study by Seigneur and Jensen [25] proposes a solution for spam by hiding one's email address using a ephemeral email addresses. If spams begin to arrive via the ephemeral email address, the address is retired. This type of service is offered commercially to end users. In our approach, we deliberately expose disposable email addresses to detect how our email addresses were obtained by spammers.

## 8. DISCUSSION

The results of our study have direct implications for web users and open up new avenues for future work on spam containment that differs from the filtering strategies used today. We discuss each of these implications below.

### 8.1 Preventing Exposure

Web users frequently share their email addresses with web sites. They also post them on web pages without realizing the results of their actions. Our study finds that even a single exposure of an email address can result in immediate and high-volume spam campaigns. While sharing email addresses with popular Internet sites does not appear to result in spam, less reputable sites appear willing to distribute email addresses to spammers, leading to greater exposure. Thus, users must exercise caution in exposing their email addresses. Fortunately, users can use simple obfuscation when posting an email address to defeat current email harvesters.

Web site operators can also take steps to protect their users' credentials. They can limit crawler-based harvesting by simply prohibiting posts with email addresses in a public forum or automatically obfuscating the email addresses. While some of these steps are already in place at web forums, there is a need for wider adoption.

While some small-scale commercial email service providers offer disposable email addresses to users, they are not offered by major email providers. Large-scale adoption of disposable email accounts would reduce the risk in providing an email address: users can receive important information but disconnect themselves from spam messages because the address becomes invalid soon after. Such an approach can also be combined with automated junk mail filtering.

### 8.2 New Avenues in Spam Filtering

Our work experimentally validates that spammers use web crawlers to harvest email addresses on a variety of web pages.

We found that many crawlers revisit to harvest new email addresses and that they are often aggressive in how quickly they return. This implies that blocking Web access for aggressive crawlers in general or repeat crawlers that previously led to spam may curtail spam. We also found that spamming crawlers rarely shared their ASNs with good crawlers. This supports blocking Web access for crawlers from ASNs who have sent spam in the past. Thus, a symbiotic relationship between spam filters and web servers is an interesting area of future exploration for spam containment.

We also found that top spammers aggressively crawl to harvest new email addresses and constantly search for open relays to transmit spam. To do so, they appear to be using machines belonging to their own IP prefixes. This is intuitive since botnets are typically rented out for a short duration and while they can be useful for sending spam for a short period, they are not suited for continuous and regular crawling and open-relay discovery. These observations have important implications for curtailing spam. First, any prefix that has attempted to relay spam can be blocked Web access. An aggressive strategy may even block web access for the entire ASN. This strategy can also work in the reverse direction: The IP or ASN of any web crawler that has led to spam in the past can also be blocked from sending any email, relay attempt or otherwise. The effectiveness of these approaches remains a future area of investigation.

### 8.3 Future Work

While our study yields its own interesting results, it invites further large-scale exploration. With aggressive seeding of bait email addresses scattered throughout the Web, we can better characterize spammer web crawling behavior and identify previously undetectable machines used in spamming. Likewise, disposable email addresses can help determine web site compliance with local laws, such as the United States CAN-SPAM Act [26], and to run “name and shame” campaigns to force better web site privacy practices.

## 9. REFERENCES

- [1] M. B. Prince, L. Holloway, E. Langheinrich, B. M. Dahl, and A. M. Keller, “Understanding how spammers steal your e-mail address: An analysis of the first six months of data from Project Honey Pot,” in *Conference on Email and Anti-Spam (CEAS)*, 2005.
- [2] “qpsmtpd: SMTP server,” <http://smtpd.developer.com/>.
- [3] Amazon.com, Inc, “Alexa Web information service (AWIS),” 2008, <http://aws.amazon.com/awis>.
- [4] A. Pathak, Y. C. Hu, and Z. M. Mao, “Peeking into spammer behavior from a unique vantage point,” in *USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*, 2008.
- [5] “SpamAssassin wiki,” <http://wiki.apache.org/spamassassin/SpamAssassin/>.
- [6] M.-Y. Kan, “MeURLin: URL-based classification of web pages,” <http://wing.comp.nus.edu.sg/meurlin/index.html>.
- [7] M.-Y. Kan and H. O. N. Thi, “Fast webpage classification using URL features,” in *ACM International Conference on Information and Knowledge Management*, Nov. 2005.
- [8] D. S. Anderson, C. Fleizach, S. Savage, and G. M. Voelker, “Spamscatter: Characterizing Internet scam hosting infrastructure,” in *USENIX Security*, 2007.
- [9] Team Cymru, “IP to ASN lookup v1.0,” <http://asn.cymru.com/>.
- [10] U. of Oregon Advanced Network Technology Center, “Route Views project,” <http://www.routeviews.org/>.
- [11] R. D. et al, “Tracking ghostnet: Investigating a cyber-espionage network,” *Information Warfare Monitor*, Tech. Rep., 2009.
- [12] Y. Xie, F. Yu, K. Achan, R. Panigrahy, G. Hulthen, and I. Osipkov, “Spamming botnets: Signatures and characteristics,” in *ACM SIGCOMM*, 2008.
- [13] Y. Zhao, Y. Xie, F. Yu, Q. Ke, Y. Yu, Y. Chen, and E. Gillum, “Botgraph: Large scale spamming botnet detection,” in *USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2009.
- [14] M. Konte, N. Feamster, and J. Jung, “Dynamics of online scam hosting infrastructure,” in *Passive and Active Measurement Conference (PAM)*, 2009.
- [15] L. F. Cranor and B. A. LaMacchia, “Spam!” in *Communications of the ACM. Vol. 41, No. 8 Pages 74-83*, aug 1998.
- [16] J. Goodman, G. V. Cormack, and D. Heckerman, “Spam and the ongoing battle for the inbox,” *Communications of the ACM*, vol. 50, no. 2, pp. 24-33, 2007.
- [17] Z. Ramzan and C. Wüest, “Phishing attacks: Analyzing trends in 2006,” in *Conference on Email and Anti-Spam (CEAS)*, 2007.
- [18] D. K. McGrath, A. J. Kalafut, and M. Gupta, “Phishing infrastructure fluxes all the way,” *IEEE Security and Privacy Magazine special issue on DNS Security*, 2009.
- [19] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. Voelker, V. Paxson, and S. Savage, “Spamalytics: An empirical analysis of spam marketing conversion,” in *ACM Conference on Computer Security (CCS)*, 2008.
- [20] Y.-M. Wang, M. Ma, Y. Niu, and H. Chen, “Spam double-funnel: Connecting Web spammers with advertisers,” in *International World Wide Web Conference (WWW)*, 2007.
- [21] J. Franklin, V. Paxson, A. Perrig, and S. Savage, “An inquiry into the nature and causes of the wealth of Internet miscreants,” in *ACM Conference on Computer Communication Security (CCS)*, 2007.
- [22] E. Allman, J. Callas, M. Delany, M. Libbey, J. Fenton, and M. Thomas, “DomainKeys identified mail (DKIM) signatures,” IETF RFC 4871, May 2007.
- [23] J. Lyon and M. Wong, “Sender ID: Authenticating e-mail,” IETF RFC 4406, April 2006.
- [24] M. Wong and W. Schlitt, “Sender policy framework (SPF) for authorizing use of domains in e-mail, version 1,” IETF RFC 4408, Apr. 2006.
- [25] J.-M. Seigneur and C. D. Jensen, “Privacy recovery with disposable email addresses,” in *EEE Security and Privacy, vol. 1, no. 6, pp. 35-39*, nov 2003.
- [26] Public Law 108 - 187, “The CAN-SPAM act of 2003,” [http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=108\\_cong\\_public\\_laws&docid=f:publ187.108.pdf](http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=108_cong_public_laws&docid=f:publ187.108.pdf).