Longtime Behavior of Harvesting Spam Bots

Oliver Hohlfeld
TU Berlin / Telekom Innovation Laboratories
oliver@net.t-labs.tu-berlin.de

Thomas Graf
Modas GmbH
post@thomas-graf.de

Florin Ciucu
TU Berlin / Telekom Innovation Laboratories
florin@net.t-labs.tu-berlin.de

ABSTRACT

This paper investigates the origins of the spamming process, specifically concerning address harvesting on the web, by relying on an extensive measurement data set spanning over three years. Concretely, we embedded more than 23 million unique spamtrap addresses in web pages. 0.5% of the embedded trap addresses received a total of 620,000 spam messages. Besides the scale of the experiment, the critical aspect of our methodology is the uniqueness of the issued spamtrap addresses, which enables the mapping of crawling activities to the actual spamming process.

Our observations suggest that simple obfuscation methods are still efficient for protecting addresses from being harvested. A key finding is that search engines are used as proxies, either to hide the identity of the harvester or to optimize the harvesting process.

Categories and Subject Descriptors

C.2.3 [**Computer-communication networks**]: Network operations—*Network monitoring*; H.4.3 [**Information Systems Applications**]: Communications Applications—*Electronic mail*

General Terms

Measurement, Security

Keywords

Spam, E-Mail, Address Harvesting

1. INTRODUCTION

The presence of unsolicited bulk e-mail (spam), which has exceeded the volume of legitimate e-mail, remains a costly economic problem. Notwithstanding existing counteracting measures, spamming campaigns advertising products are profitable even when the amount of purchases being made is small relative to the amount of spam [12]. The apparent success of spamming campaigns motivates the understanding of spamming trends and their economics, which may provide insights into more efficient counteracting measures.

Many studies address the properties of spam e-mails, traffic, and campaigns [5, 31, 19, 22], infrastructures for spam dissemination (e.g., botnets) [6, 27, 11, 19, 31, 22, 30, 10], and detection and classification methods [29, 2, 6, 7, 9, 11, 30, 14, 13]. For instance, it has been shown that spam and non-spam traffic have significantly different properties which can be used for spam classification [5]. Much fewer studies address the origins of the spamming process, e.g., concerning *address harvesting*, which remains the primary means for spammers to obtain new target addresses. Addresses can be harvested in multiple ways, e.g., from public web pages by using crawlers [20] or by malicious software locally running on compromised machines [16]. Investigating the harvesting processes is particularly relevant as it leads to new insights about spammers, according to studies revealing the social network of spammers [28] or a rather superficial effort to conceal identity [20].

To explore the origins of the spamming process, this paper conducts a large scale study involving addresses harvested from public web pages. Concretely, to identify address harvesting crawlers, we have embedded more than 23 million unique spamtrap addresses in more than 3 million visits to web pages over the course of more than three years, starting in May of 2009. 0.5% of the embedded addresses received a total of 620,000 spam e-mails. The uniqueness property of the embedded spamtrap addresses enables the mapping between the crawling activity to the actual spamming process.

Our main findings can be summarized as follows: *i)* search engines are used as proxies, either for hiding the identity of the harvester or for optimizing the harvesting process and *ii)* simple obfuscation methods are still efficient for protecting addresses from being harvested.

In addition, we find that harvesting on our web sites is on the decline. Harvested addresses are mainly spammed in batches and are only used for a short time period. We show that harvester bots are still mainly run in access networks. One interpretation of our results is that only a few parties are involved in address harvesting, each causing different spam volumes. Our findings also suggest that the usage of some harvesting software is stable. Also, harvesters make little use of Tor as anonymity service to hide their identity. Our overall study provides thus an up-to-date view on spam origins which further reveals guidelines for webmasters to protect e-mail addresses.

Our data set can be obtained from [8].

IMC'12, November 14–16, 2012, Boston, Massachusetts, USA.

2. RELATED WORK

The method of identifying harvesting bots by issuing dynamically created addresses that are unique to each page request has been used for spam prevention and the identification of harvesters [24, 20, 25]. The first attempts in understanding the behavior of harvesters have been undertaken by Prince et al. [20] and Schryen [24] in 2005. Based on 2500 spam e-mails, Schryen [24] investigates whether the top level domain of an e-mail address is relevant for spammers and finds that .com addresses attract more spam. A more systematic study of address harvesting was done by Prince et al. [20] by using a distributed platform using 5000 participants to advertise spamtrap addresses and receive spam (Project Honey Pot). The authors present preliminary results on the average turnaround time of e-mails, User Agent strings used by harvester bots, and their geolocation. The data has been obtained over a period of six months and is based on an unstated number of spam e-mails. In particular, the paper classifies harvesters into two distinct categories by the message turnaround time: hucksters and fraudsters.

Aspects of address harvesting were revisited by Shue et al. [25] in 2009. Their study is based on 96 spam e-mails and studies the geolocation of harvesters, strength of presentation methods, turnaround times, and the aggressiveness of harvester bots expressed by the frequency of page visits.

Several spam prevention studies [3, 23] propose to pollute spammers' databases and thus to inflate the number of available recipients in order to reduce spam on legitimate accounts. In contrast, our study is concerned with dynamically generating spamtrap addresses for identifying the properties of address harvesting.

As trends in the world of spam and malware are changing fast, this paper presents an up-to-date view on address harvesting and content spamming. To the best of our knowledge, we are the first to present a large-scale data set spanning over more than three years that combines aspects of harvesting and comment spamming. In contrast to [24] and [25], our spam body consists of 620,000 spam e-mails and is larger by magnitudes. While we confirm previous findings, we also study new aspects such as *i)* the connection between harvesting and comment spamming activities, *ii)* the efficiency of blacklisting, *iii)* the usage of the Tor anonymity service, *iv)* host-level properties of bots, and *v)* the usage of search engines as proxies to hide the identity of harvesters.

3. METHODOLOGY & DATASETS

To study the properties of the address harvesting process of harvesters using web crawlers, we use a methodology relying on issuing unique spamtrap e-mail addresses via the web. As the addresses are uniquely generated for each page request, their usage can be directly mapped to a specific page request. The generated addresses are embedded into nine low-profile web pages of various types (gaming, private web pages, research group, etc., see Table 1) and popularities. This methodology is implemented in web sites by including a dynamic script that generates unique e-mail addresses for each page request and logs information about the visitors. The resulting distributed platform to advertise our spamtrap addresses and to receive spam is illustrated in Figure 1.

Webmasters are typically confronted with the dilemma of choosing a method for displaying e-mail addresses on the web: Should addresses be presented in a user-friendly or obfuscated way to prevent spam? Which presentation method is the most robust against address harvesters? To shed light on this dilemma, the information included in the web pages of our study consists of six different spamtrap addresses, each being displayed with one of the following presentation and obfuscation techniques: *i*) a *mailto:* link (**MTO**), *ii*) non-linked, plain-text address (**TXT**), *iii*) e-mail obfuscated in the form of *user [at] domain [dot] tld* (**OBF**), *iv*) obfuscated using Javascript code (**JS**), *v*) included in a hidden data field of a web form (**FRM**), and *vi*) plain-text address inside an HTML comment (**CMT**). All of the above described addresses consist of random strings of 10 characters each (*RND IDs*, e.g., "jdi4gj8bzx"). We use random strings as they are sufficiently hard to guess. Table 1 shows the total number of embedded random IDs per web page, as well as the respective measurement periods. Note that the number of random IDs correlates with the number of monitored page requests for each web site.

In addition to random strings, we issue realistic looking addresses containing random combinations of first and last names generated from phone book records (*Name IDs*, e.g., "john.doe"). These addresses were introduced in January 2010, six months after the random IDs. The number of embedded addresses per web page is shown in Table 1. Compared to random strings, the assumption is that realistic looking addresses are harder to identify as spamtrap addresses, but are also easier to guess. As the total number of possible firstname × lastname combinations is much smaller than the total number of possible random IDs, we only issue name IDs using the *MTO* embedding method, to avoid running out of addresses. Webmasters often append strings to displayed addresses that are to be removed by users, causing bots to extract non-existent addresses. Therefore, by using the MTO method, we embed name addresses twice in each web page: once by using the regular address and once by appending a "_remove_" tag.

E-mail addresses are advertised by appending different domains and TLDs. Our e-mail domains are handled by several mail exchange servers located in different networks. Servers which are under our control run a qsmtpd SMTP server that captures the complete SMTP transactions and accepts any incoming mail. Other servers provide us the unfiltered e-mail feed via IMAP. We consider any e-mail sent to trap addresses as spam.

As harvesters can only be identified once the first spam is received, we log basic information such as the requesting IP for all page visits. In addition, web site operators provided us with complete access logs since January 2010. This extended information allows us to analyze further properties such as user agent strings submitted by visitors.

As our web pages cover a variety of different genres and popularities, this selection is arguably representative. By monitoring a relatively small number of web pages concentrated in Germany, the conclusions of this study are conceivably biased. However, this bias creates the opportunity to look at a focussed set of web pages and study locality in the harvesting process.

4. HARVEST AND SPAM ACTIVITIES

This section presents the main properties of address harvesting bots. We present statistics on page requests made by bots, geolocation of bots, the usage of our spamtrap addresses, fingerprints of bots, the robustness of methods

Site	Type	Country	Start of Rnd IDs	Issued Rnd IDs (% spammed)	Issued MTO Rnd IDs (% spammed)	Issued Name IDs (% spammed)	End
A	Private blog	DE	2009-05-16	791,890 (0.23%)	144,769 (0.45%)	211,851 (0.12%)	2010-11-29
B	Gaming web site	DE	2009-05-16	2,807,925 (0.06%)	469,804 (0.19%)	929,147 (0.03%)	2012-08-24
C	Private web site	DE	2009-05-16	21,558 (0.53%)	3,890 (1.54%)	5,938 (0.12%)	2011-03-28
D	Mail archive	DE	2009-05-16	5,191,288 (1.75%)	917,836 (3.20%)	1,518,105 (0.68%)	2012-08-24
E	Private web page	DE	2009-05-17	1,097 (0.00%)	197 (0.00%)	320 (0.00%)	2012-08-17
F	Private web page	DE	2009-05-16	400,490 (0.54%)	70,424 (1.47%)	118,481 (0.09%)	2011-10-30
G	Spamtap page	DE	2010-01-14	998132 (0.29%)	166,408 (0.54%)	332,694 (1.07%)	2012-08-24
H	Research group	DE	2010-01-24	7,582,332 (0.07%)	1,372,051 (0.17%)	2,094,329 (0.04%)	2012-08-24
I	Fake email provider	US	2010-07-09	34,500 (0.19%)	5,750 (0.26%)	11,500 (0.03%)	2011-05-16

Table 1: Data Set Overview

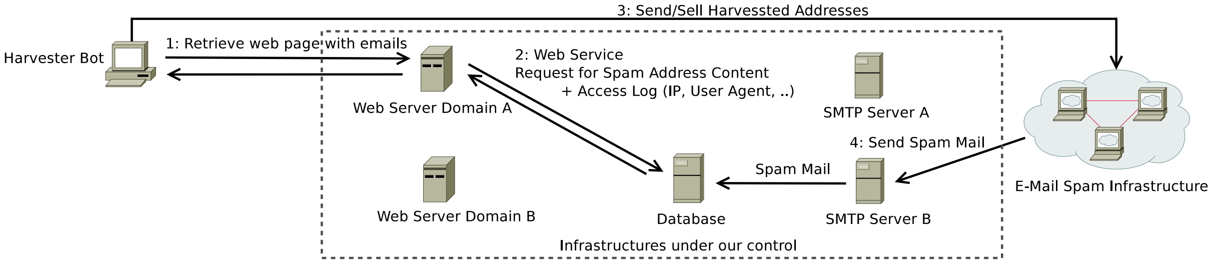

3: Send/Sell Harvessted Addresses
Harvester Bot
1: Retrieve web page with emails
Web Server Domain A
2: Web Service
Request for Spam Address Content
+ Access Log (IP, User Agent, ..)
SMTP Server A
4: Send Spam Mail
Spam Mail
E-Mail Spam Infrastructure
Web Server Domain B
Database
SMTP Server B
Infrastructures under our control

Figure 1: Measurement Methodology: Multiple web servers offer unique spamtrap addresses. Spam e-mails are received by multiple SMTP servers in multiple locations. Note that infrastructures used for harvesting and for sending spam might be run by different entities.

used to display e-mail addresses on the web, the efficiency of blacklisting, the usage of anonymity services, the relationship to comment spam, and the role of search engines.

4.1 Network Level Properties

We start by analysing the requests made by harvesting bots to the monitored web sites. By request we denote a page retrieval which resulted in spam to at least one of the retrieved e-mail addresses. Figure 2(a) shows *i)* the total number of page requests per month and *ii)* the page requests by harvesters. The figure shows a decline in harvesting activity at the monitored sites, especially compared to comment spam as malicious activity (cf. § 4.6). Conceivable reasons for this decline are: *i)* our sites get blacklisted over time, *ii)* increasing e-mail turnaround time or, *iii)* less usage of web crawlers for address harvesting.

In total, we classified 1251 hosts as harvesters and obtained DNS records for 90% of the hosts. For the remaining 10%, no DNS record could be obtained from the authoritative DNS servers, but *whois* information. Inspecting a random subset of hosts led to mostly access networks. To our surprise, we classified 20% of the hosts as search engines whose requests originated from legitimate address spaces associated to Google, Microsoft, and Yahoo. We discuss this issue further in § 4.7. Requests by a search engine crawler that resulted in spam are shown in Figure 2(a).

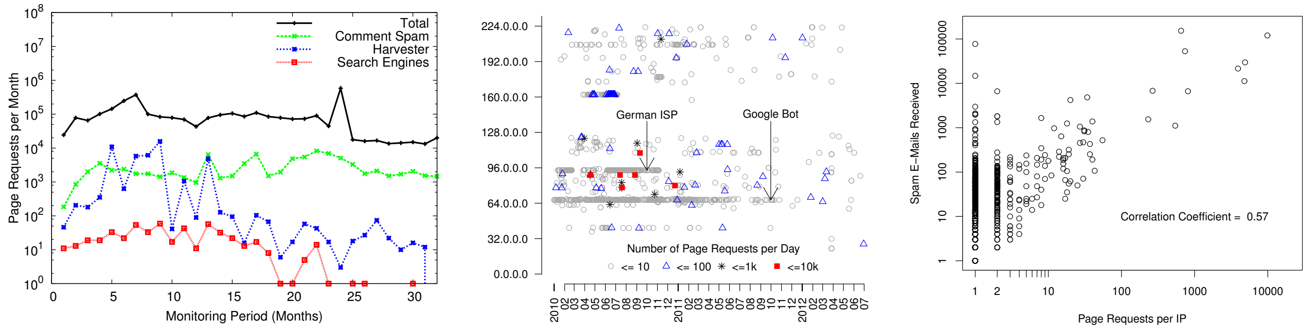
To study how requests by harvesting bots are spread over time, address space, and volume, Figure 2(b) shows a volume classification for the requests per day and IP. For most of the bots, only a small number of page requests resulting in spam can be observed (maximum 9871 requests per IP and day). A few regions in the IP address space show activity over multiple months, visible as horizontal bars. We found DSL customers by a German ISP to be the most dominant ones in March and July to August of 2010. However, the Google bot showed the longest time stability.

The figure also shows several heavy-hitters; six hosts retrieved around 10,000 pages—corresponding to 80,000 e-mail IDs—each on a single day. Manually inspecting these hosts revealed that most of the IP addresses belong to a single provider in Romania. We found 24 distinct IPs originating from this network, none of them having a DNS record. Page requests by these IPs span over almost the entire monitoring period and are responsible for a major fraction of the received spam (see Figure 3(b)). We observed requests to five of our web sites, of which 99% belong to web site D (mail archive).

To connect access statistics with the actual spam volume, we show the number of received spam e-mails vs. the page requests per IP in Figure 2(c). In many cases, only one or two page requests per IP are observed. However, the spam volume sent to addresses advertised in those requests was substantial.

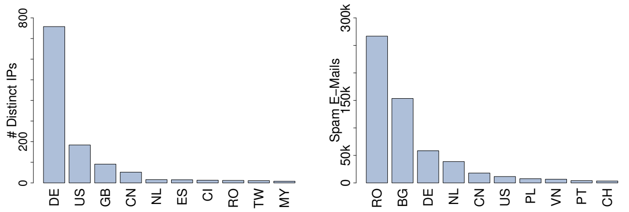
We were further interested in whether harvesting machines are primarily hosted by infected machines located in residential or business access lines, or by operating dedicated servers. For this classification, we *i)* firstly apply a reverse DNS lookup to obtain host names and *ii)* secondly look for specific text patterns in the obtained host names. We classify hosts as DSL or Cable hosts if their host name contains key words such as “dsl”, “customer”, “dialin”, etc. According to our classification heuristic, 73% of the IPs belong to ADSL or Cable access providers. This shows that harvester bots are still primarily run in residential access networks.

To study further properties of the hosts running bot software, we collected statistics about open TCP/UDP ports by port scans since mid 2011. To reduce the traffic caused by port scans and to focus our scans on harvesters, we limited our port scans to hosts blacklisted by Project Honeypot (note that we can only mark harvesters in retrospect after the first spam arrival). Only 13 hosts that we scanned harvested e-mail addresses from our web sites. Six of the scanned hosts had port 3389 open (typically used for remote



(a) Total Page Requests (b) Page Requests per Day and IP (c) Page Requests vs. Received Spams

Figure 2: Bot Visits to our web sites that lead to spam



(a) by requesting IPs (b) by spam volume

Figure 3: Top 10 Harvesting Countries

control of Windows systems), four port 22 (remote control of Unix systems), and five port 80 (HTTP).

Lastly, we study the geolocation of the observed IP addresses. By solely looking at the number of distinct IPs per country (see Figure 3(a)), the bias in our data set towards web sites in Germany is reflected in the geolocation of harvesting machines: 60.6% of all bot IPs are located in Germany. Looking at AS information, we find that the majority of harvesting requests originate from AS3320 (Deutsche Telekom residential access lines) in Germany.

However, are the German harvester bots also responsible for most of the spam volume? By looking at the total spam volume caused per harvesting location leads to a different distribution (see Figure 3(b)); harvester bots in Romania and Bulgaria caused 72% of the received spam. All the 675 Bulgarian page requests were made by a single IP located in a Bulgarian ISP in November 2010 (we observed 24 distinct IPs from Romania, as mentioned earlier). The German bots that made up for 60% of all the distinct IPs were only responsible for 10% of the spam.

4.2 E-Mail Address Usage

What happens to e-mail addresses after they were harvested? We investigated this aspect by focusing on the usage of harvested addresses. Concretely, we denote the time between the address being harvested and their first usage as the *turnaround time* and show its distribution in Figure 4. 50% of the addresses received the first spam e-mail within four days after being harvested. The slowest observed turnaround was 1068 days, whereas the fastest was 64 seconds. Focusing the analysis only on addresses advertised to search engines led to slower turnaround times (not shown in the figure); 50% of the addresses were spammed within 11 days after a visit by a search engine bot. We found the fastest turnaround to be one day, while the slowest was 611 days.

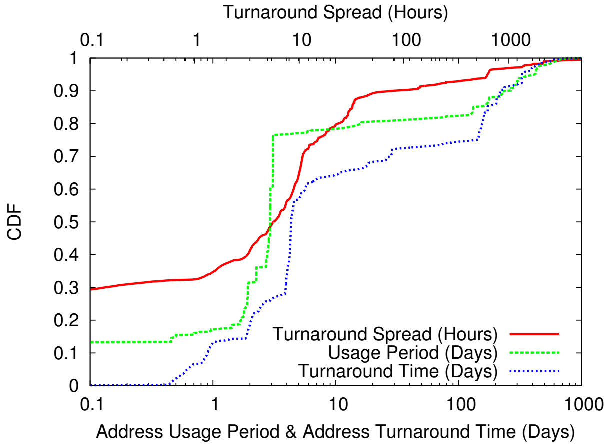


Figure 4: E-Mail Address Usage

As we embedded multiple addresses in one page, we were interested whether they were also simultaneously used for the first time. Therefore, we selected all RND addresses which received more than one spam e-mail (78%) and grouped them by page requests. We denote the *spread* in turnaround times as the range (max-min) of turnaround times for the addresses embedded in one request. A low spread indicates that all addresses in one request were firstly spammed within the same period. The distribution of the spread in hours is shown in Figure 4 (note the upper axis). 80% of the pages show a spread of less than a day (99% for search engines), and 27% of 0 seconds (94% for search engines) meaning that all extracted addresses simultaneously received their first spam. This finding suggests that spam to our spamtrap addresses was mainly sent in batches.

We also computed the amount of time that our addresses receive spam, denoted here as the *usage period*. 11% (16% for the search engines) of all addresses which received at least two spam e-mails were used for less than a second, 17% (40%) for less than a day, and 78% (51%) for less than a week. The longest observed usage of an e-mail address was 1068 days (749 days). We mention that our monitoring period spanned over 1202 days.

Comparing addresses advertised to search engine bots reveals various usage patterns; these addresses tend to be sent more often in batches to addresses embedded within a page. They also tend to have a slower turnaround time and a longer usage period. We further mention that only 558 addresses advertised to search engines were spammed, making this subset less representative than the whole data set.

4.3 Fingerprinting: User Agent Strings

In Figure 5 we show the usage of user agent strings submitted by harvesters bots in the HTTP header of the page request to our web sites. Figure 5(a) shows the use of user agents on a per-request basis. Note that the variability is modulated by heavy-hitters as shown in Figure 2(b); thus a per-IP classification shifts the popularity of user agents.

Note the variation over time visible in Figure 5(a) and 5(b) (also in Figure 6). One inference from this observation is the existence of only a few parties that harvest addresses from our sites. Depending on their activity, they can strongly skew a given quarter's statistics. Also, differences in Figure 5(a) and 5(b) suggest that the addresses harvested by different parties are not homogeneously used and caused different spam volumes.

We observed that 19% of the classified hosts as harvesters submitted a user agent string mimicking those of major search engines. By resolving IP addresses to AS numbers, we find that only 5% of the hosts using the Google bot user agent do not originate from the Google AS and are thus mimicking the Google bot. These 5% of the hosts are located in various ISPs and hosting sites (including Amazon EC2) located in seven different countries. Checking the *whois* records for each IP revealed various providers that cannot be associated directly to Google. 95% are indeed legitimate requests from the true Google AS. We did not observe a case of faked user agents for Yahoo's Slurp and Microsoft's Bing bots.

Seven years after the study of Prince [20], we find the Java user agent (e.g. "Java/1.6.0_17"), classified in our figures as "Script" and reported by [20], still to be present. Visits by this user agent span over the entire data set. We find this user agent to be used by 3% of the hosts classified as harvester. However, these hosts account for 88% of the page requests leading to spam, whereas spamtrap addresses harvested by these hosts account for 55% of the total spam, indicating that our data is skewed by one type of harvesting. In particular, the majority of the hosts located in Romania (cf. § 4.1) supplied the Java user agent string. These visits caused 99.9% of the spam volume that can be traced back to harvester bots in Romania. We found only a single host submitting six requests using "Mozilla/2.0 (compatible; NEWT ActiveX; Win32)" as user agent. This finding indicates that the usage of some harvesting software is fairly stable.

Out of curiosity, we personally replied to incoming spam in a few cases. In one case, a personalised response was received 10 minutes after our inquiry, originating from a residential access line located in the Netherlands. To our surprise, the IP from which the e-mail was sent matched the IP to which the address was issued a few days before (the harvester bot did use the "Java" user agent). It is often speculated that harvesting and mass e-mailing are two different processes, which might be conducted by different entities. However, this example shows the contrary, as the spammer did run the harvesting bot on his/her computer or used the same bot as proxy. This observation calls for further investigation.

4.4 Address Presentation Method Robustness

One aspect concerning webmasters is how to display e-mail addresses on the web to prevent spam: in a user-friendly or an obfuscated way? To address this issue, we study the robustness of presentation techniques by displaying our spamtrap addresses using a set of different presentation and obfuscation methods. For each spamtrap address which received spam, we show the relative share of spammed addresses for the used presentation methods in Figure 6.

As expected, a significant portion of spam was received by addresses presented in easy to parse plain text or as *mailto:* link. While some of the plain text obfuscated addresses (OBF) were harvested, none of the addresses presented using Javascript code received any spam. Concerning addresses advertised to search engine bots, the majority of the spammed addresses were presented using MTO (60.7%) and TXT (38.4%). Bots using the Java user agent only parsed addresses presented using MTO and TXT. These findings suggest that simple obfuscation methods, in particular Javascript, are still quite efficient to protecting addresses from being harvested.

4.5 Efficiency of Blacklisting & Usage of Anonymity Services

We query the IP based spam blacklist provided by Project Honeypot for each page request to our monitored sites at the time of the visit. Blacklist data has been collected over a period of 13 months since July 2011 and aims to evaluate the efficiency of blacklisting for blocking harvester bots. During this period, we received visits by 318 hosts that were classified as harvesting spam bots. 26% of the visiting hosts were marked by Project Honeypot as harvester.

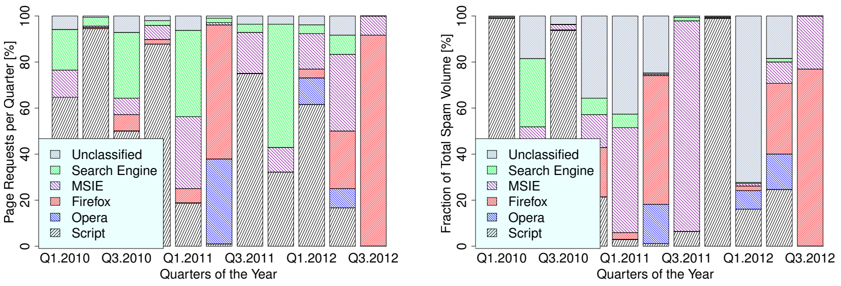
In addition, we were interested if harvesters use anonymity services—such as Tor—to hide their identity. While the default configuration of Tor exit nodes blocks traffic to port 25—used to send spam—access to port 80—used to retrieve e-mail addresses from the web—is not prohibited by default. This could make Tor more attractive to harvesters than to spammers. To check if requests originated from the Tor network, we queried the list of Tor exit nodes when a page was requested. Tor usage statistics were collected over a period of five months starting in 2012. In this period, only 0.03% of the total requests to our web sites originated from the Tor network. However, we did not receive any request using Tor that was classified as harvesting activity.

We note that the short evaluation time conceivably biases these observations, but suggests that harvesters do not make an effort to conceal their identity.

4.6 Are Comment Spammers Harvesters?

Comment spam [1, 17, 26, 21, 15] exploits the existence of web forms that are designed to let users upload content. Examples of legitimate use are: i) commenting to blogs, discussion boards, or Youtube videos, ii) messaging to webmasters, and iii) uploading files such as videos or images. Crawlers that traverse the web, download and parse web pages are needed for both activities, i.e., comment spamming and address harvesting. Given the required effort, it would be efficient to simultaneously run both activities, i.e., harvesting addresses while sending comment spam. But do spammers make a business out of both harvesting addresses *and* delivering comment spam?

To test this hypothesis, we include trap web forms in addi-



(a) by page requests

(b) by spam volume

Figure 5: User Agents Strings of Harvesting Bots

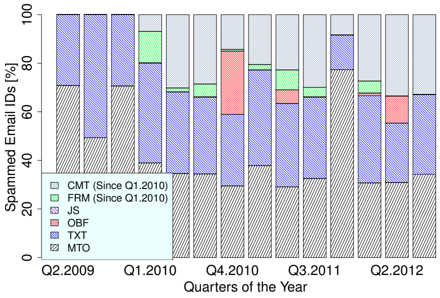


Figure 6: Spammed Email IDs by Presentation Method

tion to spamtrap addresses in web pages. Like the spamtrap addresses, our trap forms are not rendered by the browser and are thus invisible to normal users. We therefore assume any data sent over trap forms to be sent by bots. As these forms can have any structure, we replicated forms used for placing comments by the Wordpress software, frequently used to run blogs and web sites. Over a period of more than two years, we received 89,158 comment spams from 9312 distinct IP addresses. In the period of July 2010 to May 2011, five harvester hosts originating from four different countries submitted empty forms in which they could have technically sent comment spam. These hosts submitted five different user agents strings, including the Java user agent we discussed earlier. However, none of the e-mail addresses issued to comment spam bots was spammed. This suggests that comment spam bots do not harvest e-mail addresses.

To further study the differences between comment spam bots and harvesting bots, we repeated the analysis presented in § 4.1 - § 4.5 for the comment spam data set (not shown here). Our findings suggest that harvesting and comment spamming are uncorrelated activities, run using different software, and are most likely run by different entities. In this way, comment spammers do not (yet?) exploit the feasibility of simultaneously extracting and selling e-mail addresses on the market.

4.7 Harvesting: Role of Search Engines

We now address the role of search engines in the context of address harvesting. To our surprise, we received spam to spamtrap addresses advertised only to major search engine bots, i.e., Google, Microsoft, and Yahoo. In particular, 0.5% of the spamtrap addresses delivered *only* to search engines received 0.2% of the total spam. We define visits by search engines as requests made by crawlers that originate from the Google, Microsoft, and Yahoo ASes. All of the visits originating from those ASes used the proper user agent of the respective search engine bot. Concretely, 13% of the hosts classified as harvester originated from the Google AS, 3.7% from the Microsoft AS, and 1.2% from the Yahoo AS. We observe this behavior on all sites over the entire measurement period (cf. Figure 2(b) for the Google bot).

While the impact of the harvesting techniques to the overall spam volume and the number of harvested addresses is rather small, that very existence is a surprising result that has not been previously reported. It suggests that harvesters use search engines as a proxy to either *i)* hide their own identity or *ii)* optimize the harvesting process itself. As harvesters did not try to hide their identities by either using anonymity services or by masquerading as legitimate browsers by sending common user agent strings, option *ii)* seems more likely. In fact, we were able to find harvesting software that offers the functionality of querying search engines. Concretely, the advertisement for ECrawl v2.63 [18] states: "Access to the Google cache (VERY fast harvesting)". The Fast Email Harvester 1.2 "collector supports all major search engines, such as Google, Yahoo, MSN" [4]. This finding suggests that web site operators should not advertise e-mail addresses to search engine bots. It also calls for a further systematic investigation.

5. CONCLUSIONS

We have presented a longitudinal study of address harvesting that is based on a large-scale data set and that gives an up-to-date view on spam origins. We show that some aspects of harvesting are fairly stable over time, e.g., the existence of a certain user agent that has been observed for years, and the poor performance of harvesting software in breaking obfuscation methods. One interpretation of our results suggest that only a few harvesting parties are active, each causing different spam volumes. We also find that new aspects arise in the harvesting process, such as the emerging trend in the usage of legitimate search engines as proxies for address harvesting. Other observations point to the decline of harvesting activity on our sites and the existence of only a small set of hosts being responsible for a major fraction of the received spam.

Our findings reveal several guidelines for webmasters, e.g., *i)* to continue using obfuscation methods for displaying e-mail addresses on the web, e.g., by using Javascript code, *ii)* to restrict embedding e-mail addresses in web sites sent to legitimate browsers, and in particular not to search engine bots, *iii)* to rely on blacklists, e.g., provided by Project Honey Pot, to limit the likelihood of address harvesting.

6. ACKNOWLEDGEMENTS

We thank Bernhard Ager, Gregor Maier, Enric Pujol, and Nadi Sarrar for their insightful comments. Further, we thank the webmasters for including our scripts. We also thank the anonymous IMC reviewers and our shepherd Vern Paxson for their valuable comments and suggestions to improve this manuscript. Last but not least, we thank all the anonymous spammers and harvesters for making this study possible.

7. REFERENCES

[1] S. Abu-Nimeh and T. Chen. Proliferation and detection of blog spam. *IEEE Security and Privacy*, 8(5):42–47, Sept. 2010.

[2] I. Androutsopoulos, J. Koutsias, K. V. Chandrinos, G. Paliouras, and C. D. Spyropoulos. An evaluation of Naive Bayesian anti-spam filtering. In *Workshop on Machine Learning in the New Information Age*, pages 9–17, 2000.

[3] A. Antonopoulos, K. Stefanidis, and A. Voyiatzis. Fighting spammers with spam. In *International Symposium on Autonomous Decentralized Systems*, 2009.

[4] eMarkSofts. Fast email harvester 1.2. `http://fast-email-harvester.smartcode.com/info.html`, 2009.

[5] L. H. Gomes, C. Cazita, J. M. Almeida, V. Almeida, and W. Meira, Jr. Characterizing a spam traffic. In *ACM IMC*, pages 356–369, 2004.

[6] G. Gu, J. Zhang, and W. Lee. BotSniffer: Detecting botnet command and control channels in network traffic. In *Network and Distributed System Security Symposium*, 2008.

[7] S. Hao, N. A. Syed, N. Feamster, A. G. Gray, and S. Krasser. Detecting spammers with snare: spatio-temporal network-level automatic reputation engine. In *USENIX security symposium*, pages 101–118, 2009.

[8] O. Hohlfeld. IMC 2012 address harvesting data set. `http://www.net.t-labs.tu-berlin.de/~oliver/harvesting/`, 2012.

[9] T. Holz, C. Gorecki, K. Rieck, and F. C. Freiling. Measuring and detecting fast-flux service networks. In *Network and Distributed System Security Symposium*, 2008.

[10] X. Hu, M. Knysz, and K. G. Shin. Measurement and analysis of global IP-usage patterns of fast-flux botnets. In *IEEE INFOCOM*, pages 2633–2641, 2011.

[11] J. P. John, A. Moshchuk, S. D. Gribble, and A. Krishnamurthy. Studying spamming botnets using botlab. In *USENIX NSDI*, pages 291–306, 2009.

[12] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage. Spamalytics: An empirical analysis of spam marketing conversion. In *ACM Conference on Computer and Communications Security*, pages 3–14, 2008.

[13] J. Kim, K. Chung, and K. Choi. Spam filtering with dynamically updated URL statistics. *IEEE Security and Privacy*, 5(4):33–39, July 2007.

[14] M. Knysz, X. Hu, and K. G. Shin. Good guys vs. bot guise: Mimicry attacks against fast-flux detection systems. In *IEEE INFOCOM*, pages 1844–1852, 2011.

[15] P. Kolari, A. Java, and A. Joshi. Spam in Blogs and Social Media, Tutorial . In *International Conference on Weblogs and Social Media*, 2007.

[16] C. Kreibich, C. Kanich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage. Spamcraft: an inside look at spam campaign orchestration. In *USENIX LEET*, 2009.

[17] Y. Niu, H. Chen, F. Hsu, Y.-M. Wang, and M. Ma. A quantitative study of forum spamming using context-based analysis. In *Network and Distributed System Security Symposium*, 2007.

[18] Northworks Solutions Ltd. Ecrawl v2.63. `http://www.northworks.biz/software.html`, 2012.

[19] A. Pathak, F. Qian, Y. C. Hu, Z. M. Mao, and S. Ranjan. Botnet spam campaigns can be long lasting: evidence, implications, and analysis. In *ACM SIGMETRICS*, pages 13–24, 2009.

[20] M. B. Prince, B. M. Dahl, L. Holloway, A. M. Keller, and E. Langheinrich. Understanding how spammers steal your e-mail address: An analysis of the first six months of data from project honey pot. In *Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*, 2005.

[21] A. Rajadesingan and A. Mahendran. Comment spam classification in blogs through comment analysis and comment-blog post relationships. In *Computational Linguistics and Intelligent Text Processing*, pages 490–501, 2012.

[22] A. Ramachandran and N. Feamster. Understanding the network-level behavior of spammers. In *ACM SIGCOMM*, pages 291–302, 2006.

[23] S. Roy, A. Pathak, and Y. C. Hu. Mitigating the impact of spams by internet content pollution. In *ACM SIGCOMM Poster*, 2007.

[24] G. Schryen. An e-mail honeypot addressing spammers' behavior in collecting and applying addresses. In *IEEE Information Assurance Workshop*, 2005.

[25] C. A. Shue, M. Gupta, J. J. Lubia, C. H. Kong, and A. Yuksel. Spamology: A study of spam origins. In *Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*, 2009.

[26] A. Thomason. Blog spam: A review. In *Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*, 2007.

[27] Y. Xie, F. Yu, K. Achan, R. Panigrahy, G. Hulten, and I. Osipkov. Spamming botnets: signatures and characteristics. *ACM CCR*, 38(4):171–182, Oct. 2008.

[28] K. S. Xu, M. Kliger, Y. Chen, P. J. Woolf, and A. O. Hero. Revealing social networks of spammers through spectral clustering. In *IEEE ICC*, 2009.

[29] L. Zhang, J. Zhu, and T. Yao. An evaluation of statistical spam filtering techniques. *ACM Transactions on Asian Language Information Processing*, 3(4):243–269, Dec. 2004.

[30] Y. Zhao, Y. Xie, F. Yu, Q. Ke, Y. Yu, Y. Chen, and E. Gillum. Botgraph: large scale spamming botnet detection. In *USENIX NSDI*, pages 321–334, 2009.

[31] L. Zhuang, J. Dunagan, D. R. Simon, H. J. Wang, and J. D. Tygar. Characterizing botnets from email spam records. In *USENIX LEET*, 2008.