Available online at www.sciencedirect.com
ScienceDirect
Journal of Network and
Computer Applications 31 (2008) 966–985
Journal of NETWORK and COMPUTER APPLICATIONS
www.elsevier.com/locate/jnca

On the properties of spam-advertised URL addresses

Eleni Georgiou[a] , Marios D. Dikaiakos[a,*], Athena Stassopoulou[b]

[a]Department of Computer Science, University of Cyprus, CY-1678 Nicosia, Cyprus
[b]Department of Computer Science, Intercollege, CY-1700 Nicosia, Cyprus

Received 18 August 2006; received in revised form 30 December 2006; accepted 31 January 2007

Abstract

The main purpose of most spam e-mail messages distributed on Internet today is to entice recipients into visiting World Wide Web pages that are advertised through spam. In essence, e-mail spamming is a campaign that advertises URL addresses at a massive scale and at minimum cost for the advertisers and those advertised. Nevertheless, the characteristics of URL addresses and of web sites advertised through spam have not been studied extensively. In this paper, we investigate the properties of URL-dissemination through spam e-mail, and the characteristics of URL addresses disseminated through spam. We conclude that spammers advertise URL addresses non-repetitively and that spam-advertised URLs are short-lived, elusive, and therefore hard to detect and filter. We also observe that reputable URL addresses are sometimes used as decoys against e-mail users and spam filters. These observations can be valuable for the configuration of spam filters and in order to drive the development of new techniques to fight spam.

Keywords: Spam e-mail; Characterization study; URI

1. Introduction

Spam e-mail refers to unsolicited e-mail messages that are sent with automated methods to millions of recipients (The Spamhaus Project, 2006). Spam messages are annoying, offensive, fraudulent and incur significant cost to their recipients, in terms of wasted processing time, bandwidth, storage space and loss of productivity (Goodman et al., 2005).

*Corresponding author. Tel.: +357 22892700.
E-mail addresses: cs01ge@cs.ucy.ac.cy (E. Georgiou), mdd@cs.ucy.ac.cy (M.D. Dikaiakos), stassopoulou.a@intercollege.ac.cy (A. Stassopoulou).

1084-8045/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2007.01.003

Typically, e-mail spammers seek financial profit through the promotion of products and services. Sending large amounts of e-mail is neither difficult nor expensive. However, spammers need to do more than that: as e-mail users try to protect themselves from spam by installing filters that seek to identify spam e-mails either by their source or by their content, spammers need to invent new ways to hide their identity and avoid filter detection.

Currently, the majority of spam messages are encoded in the hypertext markup language (HTML). The use of HTML can improve the presentation of e-mail content on HTML-aware e-mail clients, making it more appealing to its recipients thanks to the use of different fonts, colors, pictures, etc. HTML encoding helps also in the evasion from spam filters in a number of exploits described as HTML-based obfuscation: using HTML, spammers can render their e-mail content undetectable by inserting in it invisible text with zero font size, by splitting up the e-mail content inside HTML tables, and so on. Hence, spammers manage to alter the lexical patterns that are detectable by text-based spam filters while maintaining the information they wish to present to e-mail recipients intact. Last, but not least, spammers adopt HTML encoding to entice e-mail recipients into visiting web sites that are advertised through spam. To this end, HTML-encoded spam e-mails carry URL addresses hidden behind "call-to-action" text or image anchors. E-mail recipients are lured into clicking upon these URLs when reading their e-mail on web-enabled e-mail clients, in order to reach spam-advertised web-sites and services.

Arguably, e-mail spam serves nowadays as the cheapest and easiest mechanism for disseminating spam-advertised URLs and their respective web sites to millions of Internet users. Consequently, the advertisement of these web sites can be considered as the root cause behind the problem of e-mail spam. Nevertheless, little attention has been given to the properties and the characteristics of URLs contained inside spam messages, although such an investigation could lead to a better understanding of the spam problem. For example, it would be interesting to estimate the lifetime of spam-advertised sites and the recurrence of URL advertisements; also, to identify any distinct characteristics of URLs circulated through spam e-mails: are they short and mnemonic or long and cryptic? Do they refer to static or dynamic content? Do they point directly to the advertised content or hide behind redirection? Furthermore, it would be interesting to investigate the statistics that spam messages have vis-à-vis the URLs contained therein: for instance, the average number of URLs found inside spam messages, the co-existence of trustworthy or random URLs inside the spam messages, and so on.

This information could help us understand the mechanisms that spammers use to advertise web sites and the tricks they employ to avoid spam-filter detection or prosecution by legal authorities. Moreover, it could expose distinctive features of spam messages. Such an understanding could be proven useful for policy makers who seek effective strategies to regulate e-mail spam (Moustakas et al., 2005), to spam-filter developers who are looking into extending the coverage of spam filters (Albrecht et al., 2005), and to researchers who look for improved ways to fight spam (Androutsopoulos et al., 2005; Li and Hsieh, 2006; Nelson et al., 2006).

In this paper, we present a characterization study that focuses on the characteristics of HTML-encoded spam e-mails and the URLs disseminated through such messages. To this end, we analyze four (4) sets of spam messages. To the best of our knowledge, this is the only study so far that focuses on the properties of URL addresses advertised through spam. The remaining of this paper is organized as follows: Section 2 presents a brief overview of the problem of e-mail spam and discusses related work. In Section 3, we present a system that we built to analyze HTML-encoded spam e-mails and extract properties of the URLs carried inside those messages; we also describe the spam archives used in our study. In Section 4, we present our statistical analysis. A summary of our findings is given in Section 5. Finally, Section 6, presents our main conclusions and suggestions for future work.

2. E-mail spamming

Despite strong efforts to regulate and eventually eliminate spam, during the last few years the volume of spam messages has been increasing continuously. In June 2003, BrightMail reported that 49% of all e-mail was spam. In May 2004, this figure had increased to 64%, whereas, according to Postini, 75–80% of all e-mail was spam in 2004. Recent estimates suggest that currently 81% of e-mail traffic is spam. Due to its volume, spam not only is annoying to individual e-mail users, but also incurs a significant financial cost to institutions, due to productivity loss that results from the effort that users need to spend in handling spam messages and dealing with their side effects. Moreover, the cost in wasted processing power, storage space and consumed bandwidth is not negligible: according to Ferris Research, the overall cost of receiving spam e-mails for USA companies in 2002 was 8.9 billion dollars. This estimate took into account productivity loss and consumption of network resources. In a similar study, in July 2004, Nucleus Research estimated that spam was costing 1934 dollars per employee a year in loss of productivity, while the cost in July 2003 was 874 dollars per employee per year. According to estimates by Ferris Research, the cost of spam in 2005 was expected to reach 17 billions dollars for US companies and to 50 billion dollars worldwide. Finally, in early 2005, The Spamhaus Project (2005) predicted that by mid-2006 spam could reach 95% of all e-mail traffic.

2.1. *Tricks of the trade*

E-mail messages are transferred between a sender and a recipient e-mail server through the simple mail transfer protocol (SMTP) (Klensin, 2001; Postel, 1982). SMTP was published in 1982; it was designed in a way that it would allow anybody to use an SMTP server to send e-mails without authentication of her identity. This vulnerability facilitates the spoofing of e-mail sender IDs and has been exploited by spammers who can easily spoof or hide their real e-mail address. Spammers also take advantage of open proxy servers to mask their identity. Open proxy servers are mis-configured Internet hosts that allow traffic from any Internet service to be routed through them. Spammers identify and hijack such insecure proxy servers and route spam e-mails through them.

To achieve their goals, spammers need to have their messages received by as many recipients as possible. To this end, spammers must acquire very large collections of operational e-mail addresses and evade the spam filters that are commonly deployed by Internet Service Providers and end-users. Spammers use a number of techniques to collect e-mail addresses, such as: (a) using crawlers to harvest e-mail addresses from web pages, (b) buying databases with e-mail addresses, (c) generating user names for known domain names by using dictionary attacks or even simple brute force approaches (Prince et al., 2005).

Spammers employ a large and evolving variety of techniques to evade spam filters. Among other filter-evasion techniques, the encoding of spam messages in HTML is often used to obfuscate the real content of spam e-mails: using tables, inserting illegal tags, inserting white text on white background are just a few of the exploits found in HTML-encoded spam. The use of HTML can also make the content of each spam message unique and, thus, undetectable by content-based filters through the insertion of random modifications in message text, which are not readable by the human e-mail recipient. A detailed description of these techniques can be found in Graham-Cumming (2003).

In recent years, several legal and technical methods for fighting spam have been proposed and/or implemented. Legal methods try to regulate spam through legislation (Moustakas et al., 2005; Park et al., 2005; Sorkin). For instance, the American legislation (Can Spam Act 2004) requires that bulk mailers maintain "opt-out" lists, i.e., e-mail advertising lists in which recipients are signed up without their knowledge or permission, but may request to be removed therefrom (Arora, 2006). In contrast, EU legislation (EU directive 2002/58/EC) legalizes only "opt-in" lists, i.e., e-mail advertising lists which users must deliberately sign on to. Although an increasing number of countries take legislative measures to fight spam, the legislative regulation is actually very difficult since spammers very often choose to distribute spam from countries with no legal restrictions.

Technical methods that address the problem of spam focus either on filtering of spam messages or on wider changes to the e-mail system. Proposed changes to the e-mail system include the adoption of authentication mechanisms (Lawton, 2005), micro-payment schemes (Abadi et al., 2003; Kraut et al., 2005), challenge-response schemes (Iwanaga et al., 2003; Roman et al., 2006), schemes to encapsulate policy within the e-mail address (Ioannidis, 2003), etc. Such proposals typically entail major upgrades or replacement of existing e-mail protocols and systems; therefore, their wide adoption is delayed by serious concerns about their cost. Filtering techniques seek to filter the incoming mail either at the level of the e-mail server (real-time black lists, reverse DNS lookup) or at the level of the e-mail client program (content filters, fingerprint filters, call-to-action filters, header filters), using heuristics, black-lists, signatures or machine learning approaches (see for instance (Surbl; Albrecht et al., 2005; de Freitas and Levene, 2004; Hird, 2002; Li and Hsieh, 2006; Webb et al., 2006; Nelson et al., 2006); for a survey of spam-filtering tools, the interested reader may look at, Carpinter and Hunt, 2006).

It is worth noting that spammers continuously evolve and adapt their exploits following the evolution of spam filters (for a description of the evolution of spam see Hulten et al., 2004; Pu and Webb, 2006) and vice versa. This "competition" between spammers and filter developers has been paralleled to an arms race (Pu and Webb, 2006).

2.2. *Related work*

Spam e-mail has attracted significant attention from the IT industry and the media, as it represents the primary annoyance of the Internet experience nowadays. Nevertheless, very few published studies have examined the characteristics of spam-messages carrying HTML code and URL addresses. One of the first extensive characterization studies of spam e-mail traffic was published in Gomes et al. (2004). In this work, the authors examined the e-mails that arrived at a central University mail server in Brazil during a period of eight days in January 2004 (over 360,000 spam and non-spam messages). Their investigation focused primarily on the characterization of the resulting SMTP traffic, i.e., the e-mail arrival process and size distribution, as well as distributions of popularity and temporal locality. The same group, conducted a graph-theoretical analysis of e-mail traffic in Gomes et al.

(2005), using a log of 615, 102 e-mail messages received by the Brazilian University mail server in late 2004. In that work, the authors constructed a user and a domain graph for the spam log. The vertices of the user graph were e-mail senders and recipients while edges represented the sender/recipient relationship introduced by e-mail messages. In the domain graph, the vertices were either domains of external senders or e-mail recipients. The authors calculated the structural properties of these graphs and identified differences between spam and non-spam messages. A short reference to the properties of spam-advertised URLs were published in Albrecht et al. (2005), Pu and Webb (2006). In the former study, the authors analyzed 13,750 spam messages and discovered that about 42.2% of them contained more than one URL, whereas nearly 7.3% had 10 or more distinct URLs. In the latter study, the authors examined the evolution of the exploits used by spammers during a period of three years, from January 2003 to January 2006, and found that, every month, 85–95% of the spam messages examined contained at least one URL. Finally, a number of other works examined the characteristics of various aspects related to spam, such as the semantics of spam content (Hulten et al., 2004), the anatomy of phishing e-mail (Drake et al., 2004), and web spam (Webb et al., 2006).

In contrast to previous work, in this paper we focus exclusively on spam messages that carry URL addresses and investigate various static and dynamic properties of "spam-advertised" URLs. To this end, we use spam messages retrieved from Spamarchive, a community-maintained database that contains spam messages contributed by various recipients around the world, and personal spam e-mail folders. We examine a total of 236,000 spam messages sent between early 2004 and late 2005. Our results agree with the observations of earlier studies in that the large majority of spam messages (73–90% in our case) contain at least one URL. However, we also examine a range of other features that have not been studied before: the distribution of URLs found inside spam messages, the temporal distribution of the appearance of spam-advertised URLs, the properties of the actual URLs advertised through spam, the temporal characteristics of the URL advertisement through spam, etc.

3. Analyzing spam e-mail

3.1. Spam data sets

To derive the characteristics of spam messages and of spam-advertised URLs, we need to have access to a large corpus of spam e-mails. To this end, we collected e-mails from two main sources: (i) personal spam folders, contributed by users of the University of Cyprus e-mail relay server. This server is protected by the Spam Assassin filter (The Apache SpamAssassin Project), which filters all incoming messages, computes a "SPAM score," and labels e-mail messages as spam when their score is higher than a given threshold (Koutsioupis). The mail server keeps messages labeled as spam into personal spam folders of individual recipients. (ii) Spamarchive.org, a large online repository of spam e-mails. Spamarchive collects spam messages contributed by users throughout the world, and makes its repository available for testing, developing, and benchmarking anti-spam tools.

For the purposes of our study, we used four different collections (sets) of spam e-mails: three data sets (named L, M1 and M2) were retrieved from the spamarchive and one data set (S) was derived from personal spam folders. These four data sets contain collectively 234,000 e-mail messages that have been labeled as spam. The dates of these messages cover periods between 44 and 74 days in 2004 and early 2005. More details on our data sets are given in Table 1.

3.2. SPAT: a spam analysis toolset

To proceed with our analyses, we need to pre-process our data sets, identify and extract all URLs included inside the spam messages, and associate them with relevant metadata. To this end, we developed *SPAT* (spam analysis toolset), a collection of PERL scripts and JAVA classes, connected with a back-end mySQL database. SPAT provides support for pre-processing, information extraction, storage and analysis. The structure of SPAT is presented in Fig. 1. SPAT processing involves the following steps:

Step 1—Information extraction: A Perl script is used to read all the spam e-mails of a given data set and to process each spam message accordingly. In particular:

- The body and the headers of the e-mails are extracted and stored in the database. Various message metadata are also extracted (e-mail subject, sender, date, time stamp) and stored in the database.
- The body e-mails are composed usually of two parts, a plain text part and an HTML part. Most of the URLs are located in the HTML part as values of HTML-tag arguments, such as HREF, XRC, URL, SRC, ACTION, and BACKGROUND. The URLs contained inside the e-mail bodies are extracted and stored in the database.

Table 1
Data set properties

Data set	Number of e-mail messages	Period	Duration (days)	Source
L(arge)	150,021	01/12/2004–17/01/2005	48	spamarchive.org
M1	41,447	30/07/2004–12/10/2004	74	spamarchive.org
M2	41,129	25/09/2005–07/11/2005	44	spamarchive.org
S(mall)	1961	01/12/2004–24/01/2005	55	Personal inboxes

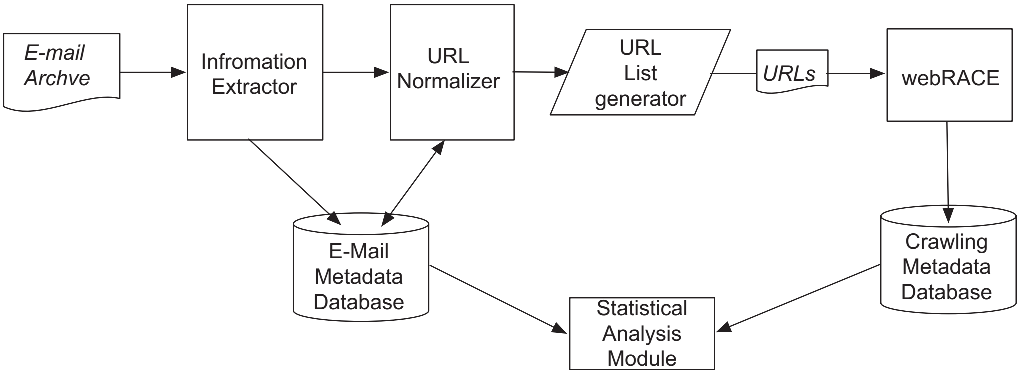

Fig. 1. SPAT: spam analysis toolset.

Step 2—URL normalization: Very often, URLs with small lexical differences point to the same web resource. For example, `http://www.hostname.com` and `http://www.HOSTNAME.com/index.html` correspond to the same web page. The purpose of the normalization process is to transform all URLs into a canonical form that facilitates the detection of lexically different but equivalent URLs, without having to undertake a full HTTP resolution thereof. The second step in SPAT processing is implemented as a JAVA program that normalizes all collected URLs and stores the result back to the database.

Step 3—Creation of URL lists: Subsequently, we sort the normalized URLs by the date of their appearance in the spam data set and group the URLs in *weekly lists*. Each weekly list includes all URLs contained in the spam messages of a given week, in the time frame of the data set under consideration. We select the granularity of one week, as this is the reference time that we use later to estimate web site lifetime; this value can, however, be tuned easily to hours, days, or months.

Step 4—Verification of URL accessibility: In this last step, we verify the accessibility of each URL in our database. For that purpose, we use *webRACE*, a JAVA-based, configurable, multithreaded web crawler (Zeinalipour-Yazti and Dikaiakos, 2002). webRACE receives as "seed" the weekly lists of URLs created in Step 3, and resolves each of these URLs using HTTP. To speedup this process and to save disk space and network bandwidth, we configured webRACE to only maintain metadata about each HTTP resolution, rather than to download the corresponding web resource.

3.3. Metrics

Using SPAT, we collect information that will help us capture key static and dynamic properties of the URL addresses that are located inside spam messages. By static properties, we mean traits that are determined by the syntax, the location, and the name of a URL address found inside a spam e-mail; for instance, whether the URL address represents a link to some other page and if it contains a valid domain name. Dynamic properties, on the other hand, are URL-address traits that correspond either to the temporal behavior of URL addresses (e.g., the frequency of their reappearance inside spam logs) or to the behavior that URL addresses exhibit when we try to resolve them using HTTP (e.g., accessibility, redirection).

4. Statistical observations

4.1. Data set statistics

We used SPAT to isolate the spam messages that carry URLs. We found out that over 73% of e-mails contained in the four data sets carry URLs; also, that these e-mails contain a total of 1,048,040 URLs, 340,308 of which are distinct and belong to 166,324 distinct domains. The details of these remarks for each separate data set are given in Table 2. In the following sections, we focus on the subset of spam e-mails that carry URLs in their body.

In Table 3, we present statistics about the distribution of URLs inside the spam e-mails. From this table, we conclude that each spam e-mail carries an average of 5.03–5.98 URLs; there is, however, a very high variability in this figure, as witnessed by the large difference between the mean and the median number of URLs per e-mail, and the high standard deviation. This variability suggests that, in all four data sets examined, there is a small

Table 2
Data set properties

Data set	E-mails	E-mails with URLs	URLs	Unique URLs
L	150,021	115,697 (77.12%)	691,987	217,747 (31%)
M1	41,447	37,358 (90.38%)	195,049	73,647 (38%)
M2	41,129	30,185 (73.40%)	151,870	43,696 (29%)
S	1961	1594 (81.29%)	9134	5218 (57%)

Table 3
Number of URLs per e-mail

Data set	Max	Mean	Median	Standard deviation
L	1287	5.98	2	14.69
M1	481	5.17	2	12.24
M2	622	5.03	2	12.85
S	291	5.73	2	14.93

number of spam messages carrying a very large number of URLs and a large number of spam messages carrying only a couple of URLs.

4.2. URL distribution

In this section, we study the distribution of URLs found inside the spam messages of our four data sets. One question we are trying to clarify is how frequently spam-advertised URLs appear in a given data set. To this end, for each distinct URL address found inside our spam logs, we calculate the average number of spam e-mails inside which this URL is located. It turns out that each spam-advertised URL appears on average in 3.18, 2.57,3.48 and 1.75 spam messages, for data sets L, M1, M2 and S, respectively. We observed, however, that the median value of spam e-mails equals 1; in other words, most of the spam-advertised URLs are found only in one spam message, although there is a small number of URLs that are heavily advertised via spam.

Similarly, the study of the distribution of the daily URL appearance shows that a spam-advertised URL appears in a spam message on average 3.18 out of the 48 days of data set L, 2.65 out of the 74 days of data set M1, 3.47 out of the 44 days of data set M2, and 1.75 out of the 55 days of data set S. Again, most URLs appear only in one day during the time frame of our data sets; there is, however, a small number of URLs that appear more often.

4.3. Static properties of spam-advertised URLs

4.3.1. HTML tagging

URLs are inserted in HTML documents as argument-values of various HTML tags. The type of the HTML argument used to place a URL inside an HTML document is an indication of the purpose of that URL's use inside the document: URLs found in HREF arguments are meant to be links to some web page; in contrast, URLs found in XRC,

Table 4
URLs types

Arguments	HREF (%)	SRC (%)	XRC (%)	BACKGROUND (%)	ACTION (%)	URL (%)
L	64.61	33.84	0.61	0.63	0.20	0.11
M1	66.29	30.93	1.93	0.59	0.18	0.07
M2	71.02	26.20	1.49	1	0.17	0.10
S	77.63	20.43	0	1.12	0.05	0.76

Table 5
Static properties of spam-advertised URLs

Data set	L	M1	M2	S
Unique URLs	217,747 (31%)	73,647 (38%)	43,696 (29%)	5,218 (57%)
Dynamic URLs	24.44%	28.77%	17.29%	21.53%
Domain Names	111,627	36,700	14,305	3,692
`.com`	59.27%	46.6%	66.26%	81.66%
`.info`	17.63%	19.12%	10.59%	7.99%
`.org`	8.86%	10.69%	1.67%	4.93%
`.net`	5.75%	6.16%	11.02%	2.03%
`.biz`	0%	9.64%	1.7%	1.05%
`.us`	2.67%	2.90%	0.32%	0.13%
`.uk`	1.91%	1.65%	0.21%	0.19%

URL, SCR or BACKGROUND arguments are probably the source of a picture or some other multimedia item (e.g. a sound file). The distribution of these arguments for the four data sets of our study are shown in Table 4; it turns out that about the $\frac{2}{3}$ of the spam-advertised URLs are links to web resources.

4.3.2. Static vs. dynamic

A percentage of 17.29–28.77% of the URLs found in spam e-mails were dynamic, i.e., they have the following syntax: ⟨`http://www.host.com/?input=value`⟩ (see Table 5). Dynamic web pages are generated by scripts that collect data from a back-end application or database, and generate HTML content depending on the user requests. Commercial web-sites and especially spam advertised web-sites are very often identified by such dynamic URLs since their content must be formatted according to the user ids or preferences.

4.3.3. URL origin

We measure the percentage of URLs whose ‘‘authority component’’ (host component) (Berners-Lee et al., 2004; W3C Technical Architecture Group, 2004) is an IP address instead of a domain name, in order to investigate the case of spam-advertised web sites using short-lived IP addresses in order to make their detection harder. We found out, however, that the majority of spam-advertised sites (about 99%) are represented using domain names. Therefore, those spam-advertised URLs that are not fake, must be registered with the domain name service (DNS).

We also studied the distribution of spam-advertised sites per top-level domain of the DNS and present our findings in Table 5. As expected, across all four data sets, almost

90% of the spam-advertised sites belong to the .com, .info, .org, and .net domains, with .com alone hosting between 46.6% and 81.66% of the spam-advertised sites.

4.3.4. *URL path length*

The path length of a URL represents the number of segments found in its "path component." This component is usually organized in hierarchical form and serves to identify a resource within the scope of the URL's scheme and naming authority (Berners-Lee et al., 2004). For example, the path component of "http://www.example.com/dir1/sdir2/foo.html" is the "dir1/sdir2/foo.html" substring, and has a length equal to 3. Typically, URL addresses are chosen to be short in order to facilitate their memorization by humans. In contrast, URLs with a long path component are hard to memorize, correspond to resources that are stored in very large and complex information collections, and have been shown to fail more frequently (Spinellis, 2003). The distribution of URL lengths for data set M1 is given in Fig. 2; the distribution for the other data sets is similar. From Fig. 2, we observe that 50% of the spam-advertised URLs have a path length of 1 or 2. There is, however, a percentage of 10.67% of very long URL addresses, with a path length larger than 10.

4.3.5. *URL reputation*

Inside the set of spam-advertised URLs that we discovered in our study, we located several addresses that belong to well-known web sites, such as popular search engines (Google, Yahoo, MSN), IT companies (Microsoft, Apple, IBM, AOL, Adobe), news web sites (news.com.com, CNN, BBC, New York Times), US Universities, governmental agencies (US Department of Labor, United States Senate), large companies (SonyEricsson, WarnerBros, Paramount, Disney) and popular e-commerce sites (Amazon, Ebay). Spammers often include reputable URLs inside their messages either to trick users into

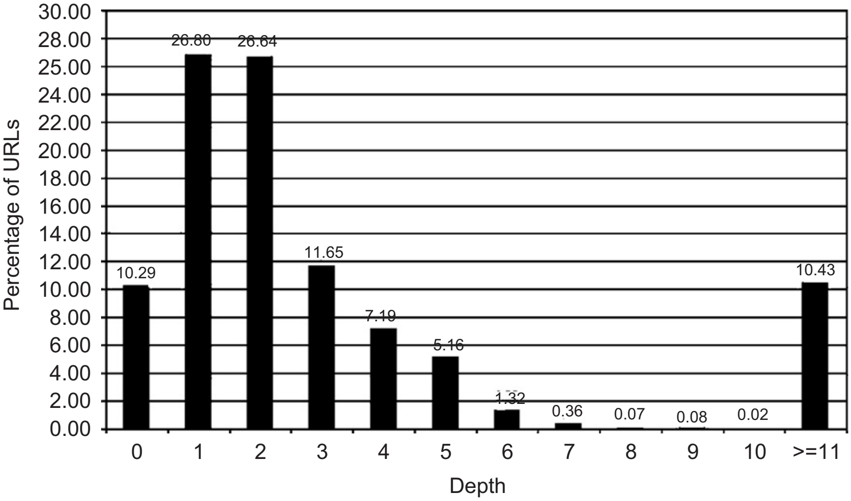

Fig. 2. URL path length (M1).

believing that the sender of the spam is legitimate or to evade spam filters that rely on URL filtering.

Due to the very large number of URL addresses of our study, it is not feasible to estimate manually the extent to which spammers use reputable URL addresses as decoys. To address this problem, we used the open directory project (ODP) Database, a very large and highly visible human-edited directory of the World Wide Web. Given that the classification of sites and pages in ODP is done by human editors, it is safe to assume that its listed web sites have useful content, are not fraudulent and, hence, can be considered as "reputable." ODP has the additional advantage that it is accessible online and can be queried through the Web automatically. By accessing ODP, we were able to examine whether a *URL* or a *host-name* extracted by SPAT was listed in ODP's directory. The outcome of this study is summarized in the diagrams of Fig. 3; it is interesting to observe that 3.5–8.64% of the URL addresses found inside the spam messages of our data sets belong to host-names classified by ODP. Obviously, this is a lower bound of reputable URL addresses, since ODP does not cover the whole Web. What these percentages tell us, however, is that a sizeable number of the URL addresses distributed through spam are reputable and that they have been inserted in junk e-mails probably without the consent of their owners.

4.4. *Dynamic properties of spam-advertised URLs*

4.4.1. *Reappearance frequency*

In this section, we investigate how often URLs reappear inside subsequent spam e-mail campaigns. To estimate the reappearance frequency, we calculate the percentage of URLs that belong to each weekly URL list of our data sets *and* are found inside the spam e-mails of later weeks. We present these percentages in Tables 6–9.

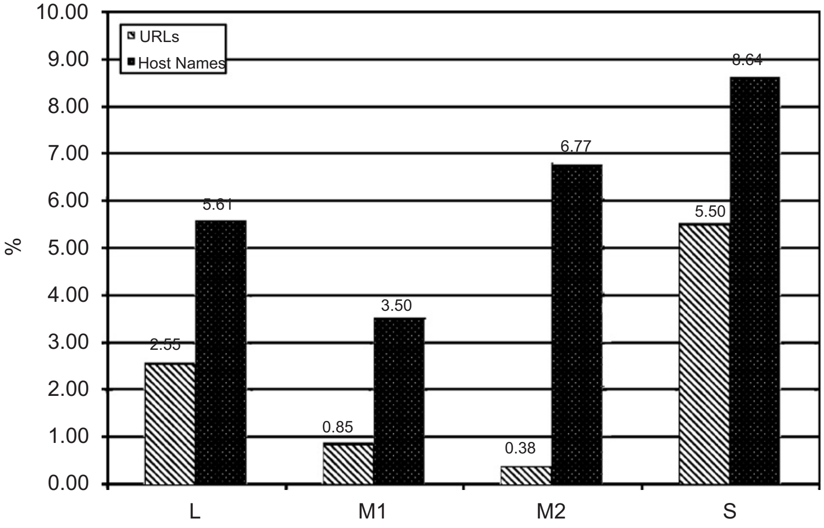

Fig. 3. Percentages of URL addresses and host-names contained in ODP directory.

Table 6
L—URL reappearance percentage (%)

Appearing in	Percentage of URLs of					
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6
Week 2	3.60					
Week 3	1.70	2.73				
Week 4	1.01	11.62	16.05			
Week 5	1.46	4.56	23.63	15.27		
Week 6	2.05	0.91	0.58	0.46	2.39	
Week 7	1.16	1.45	23.07	9.85	11.43	4.26

Table 7
M1—URL reappearance percentage (%)

Appearing in	Percentage of URLs of						
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7
Week 2	1.89						
Week 3	4.13	1.04					
Week 4	2.85	2.79	8.58				
Week 5	0.59	0.81	0.53	1.21			
Week 6	0.09	0.02	0.05	3.02	3.49		
Week 7	0	0.02	0	0.49	0.04	4.94	
Week 8	0.39	0.59	0.09	1.12	2.11	3.27	0.93

Table 8
M2—URL reappearance percentage (%)

Appearing in	Percentage of URLs of				
	Week 1	Week 2	Week 3	Week 4	Week 5
Week 2	3.86				
Week 3	11.99	4.59			
Week 4	6.88	3.61	13.65		
Week 5	6.51	1.65	9.42	20.89	
Week 6	1.05	9.86	0.67	8.37	1.64

From these tables, we can conclude that the reappearance frequency is low for all the data sets, and occurs mainly within one or two weeks after the appearance of a spam-advertised URL. Even then, the reappearance frequency is not higher than 20%, with the notable exception of data set L, where between 16 and 23% of URLs distributed in week 3 appear again after one, two, and four weeks. Reappearance of URLs completely fades out after 2–3 weeks from the first appearance of a URL in our data sets. In summary, we conclude that over 75% of the web sites that are advertised through spam, do not do so more than once. In other words, the corpus of spam-advertised web-sites changes with a fast pace.

Table 9
S—URL reappearance percentage (%)

Appearing in	Percentage of URLs of						
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7
Week 2	8.35						
Week 3	3.89	8.02					
Week 4	2.51	10.59	4.33				
Week 5	0.28	1.67	1.22	6.14			
Week 6	0	0.22	0.37	1.19	4.45		
Week 7	0	1	0.94	1.48	1.2	5.01	
Week 8	0	0.33	0.47	0.59	0.27	0	2.14

4.4.2. URL accessibility

It is interesting to estimate the temporal behavior of spam-advertised URLs, i.e., for how long a URL is expected to stay accessible after its first appearance inside a SPAM message. A URL is considered accessible if an HTTP request for it returns the corresponding web page without the appearance of any error message; error messages occur either because the URL's host-name cannot be mapped to a valid IP address, or the requested resource is not found on the web server, or there is some web server failure during the time of the request. To examine the temporal behavior of spam-advertised URLs, we feed the weekly URL lists derived by SPAT as "seeds" to the webRACE crawler, and use the crawler to automatically examine which URLs are accessible via HTTP. When the accessibility crawl is finished for all the week lists available, we measure the percentage of URLs from each weekly list successfully resolved during the crawl. The time lapse between the accessibility crawls and the last e-mail time stamp recorded in the corresponding data sets range from a couple of days to two weeks. We report these percentages in Figs. 4 and 5.

From these diagrams, we observe that one-third (32.92%) to one-half (50.60%) of spam-advertised URLs that belong to weekly lists closer to the time frame of our accessibility crawls, are found inaccessible (see the "Failed" percentages for weeks 8, 7, 6, and 8 in the diagrams of Figs. 4a, b, 5a, and b, respectively). For instance, in the case of data set M1 (Fig. 4(a)), the accessibility crawl started on October 14, 2004; the last e-mail entry in that data set was from October 18, 2004. During this accessibility crawl, webRACE failed to fetch 42.59% of the URLs that belonged to weekly list 8 of M1. As we move to "older" URLs, that is URLs which appeared in earlier weekly lists, we observe that URL accessibility remains low but does not drop too much. Only when examining URLs that are 6 to 7 weeks "old," are we able to see a noticeable drop in accessibility by nearly 9%. Similar remarks can be derived from the accessibility crawls of the remaining data sets.

4.4.3. Redirecting URLs

As part of the dynamic behavior of spam-advertised URLs, we also examine the percentage of accessible URLs that *redirect* automatically their requests to other web pages. This redirection behavior was detected with the help of the webRACE crawler, during the accessibility experiment described above. The diagrams of Figs. 6 and 7 present the percentage of spam-advertised URLs that redirect user requests to some other web

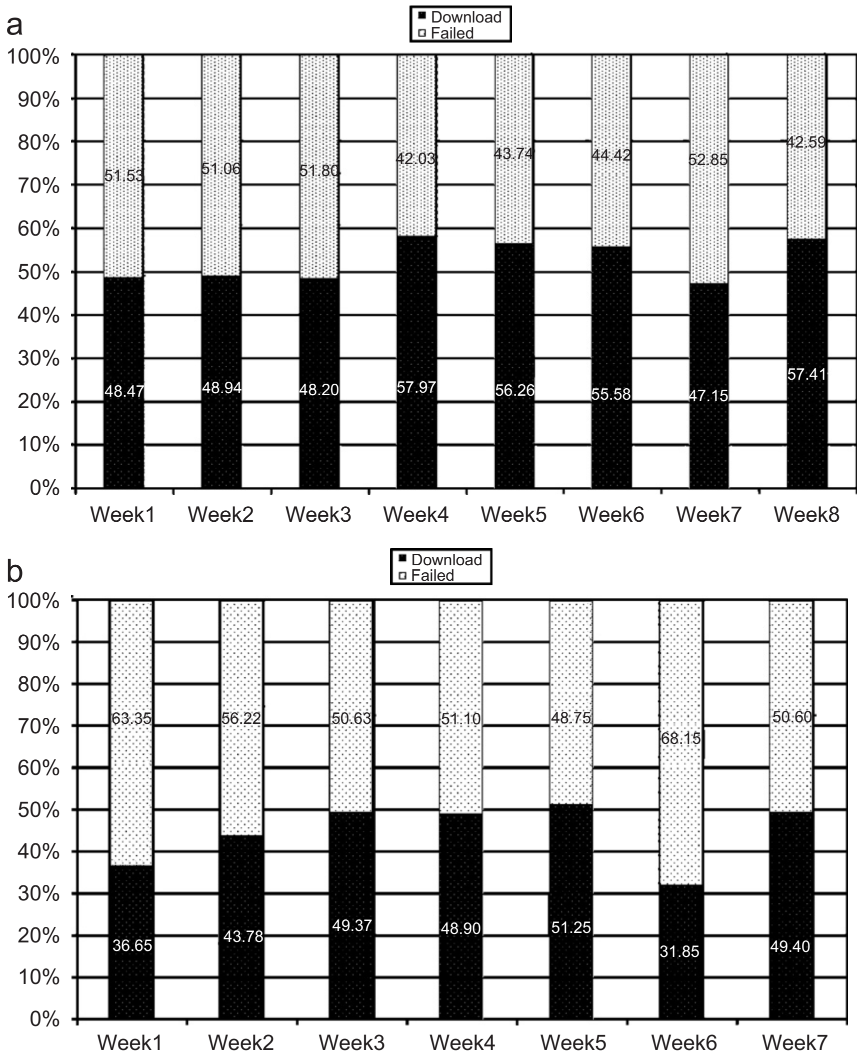

Fig. 4. (a) Accessibility test for data set M1, conducted between 14/10/04 and 18/10/04 (M1 time frame: 30/7/04–12/10/04); (b) Accessibility test for data set L, conducted between 3/2/05 and 15/2/05 (L time frame: 1/12/04–17/1/05).

page. From these diagrams, we observe that a significant number of spam-advertised URLs redirect their traffic to other web pages; this number varies from 10.52% (fifth week of data set M2 in Fig. 7) to 39.67% (eighth week of data set S in Fig. 7). This observation suggests that, quite often, spammers try to hide the identity of the web resource they seek to attract attention to, using the redirection functionality of HTML and HTTP. Also, from the diagrams in Figs. 6 and 7, we observe that, in three out of the four data sets, the percentage of redirecting URLs decreases significantly as we move to ‘‘older’’ URLs. This can be an indication that redirecting URLs decay much faster than non-redirecting ones.

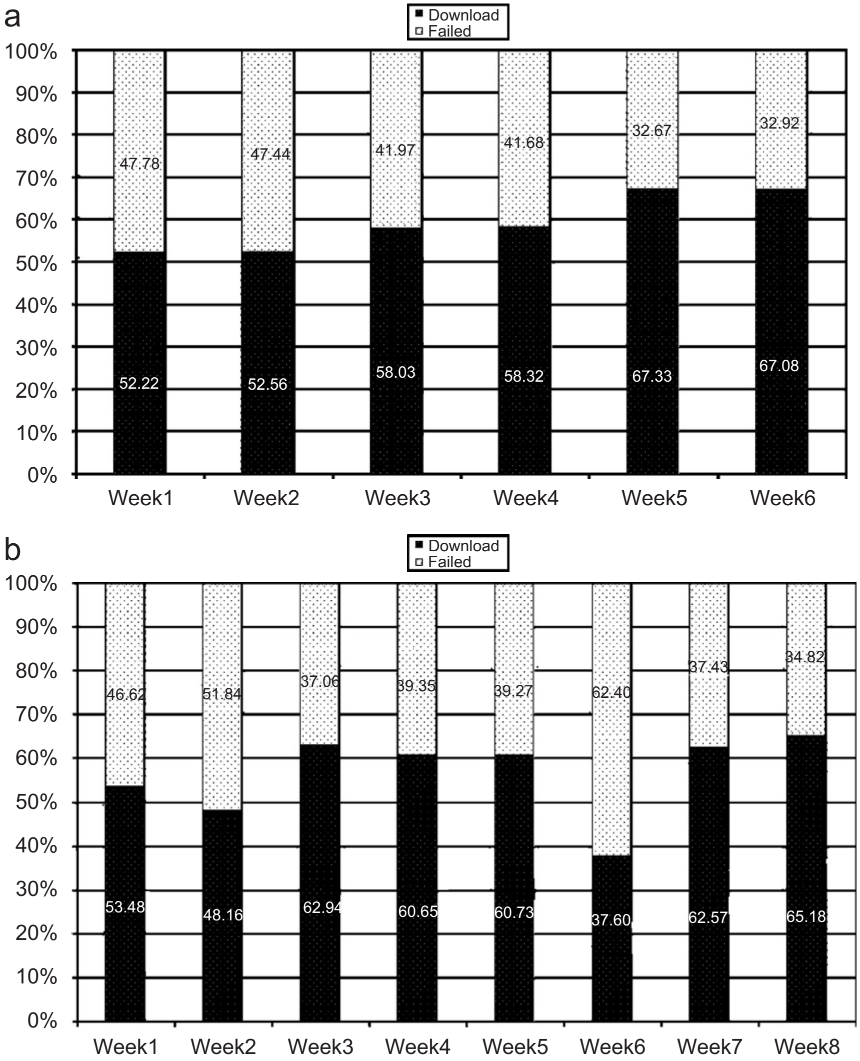

Fig. 5. (a) Accessibility test for data set M2, conducted between 11/11/05 and 17/11/05 (M2 time frame: 25/9/05–7/11/05); (b) Accessibility test for data set S, conducted on 1/2/2005 (S time frame: 1/12/04–24/1/05).

5. Summary of findings

In this work we examined four logs of spam e-mails, focusing on the characteristics of the URL addresses found inside most of these messages. From this analysis, we observe that:

- The large majority of spam messages are encoded in HTML and/or carry URL addresses in their bodies. In the data sets examined, 73–90% *of spam messages carry at least one URL*. It seems that the *dissemination of URL addresses to e-mail recipients is*

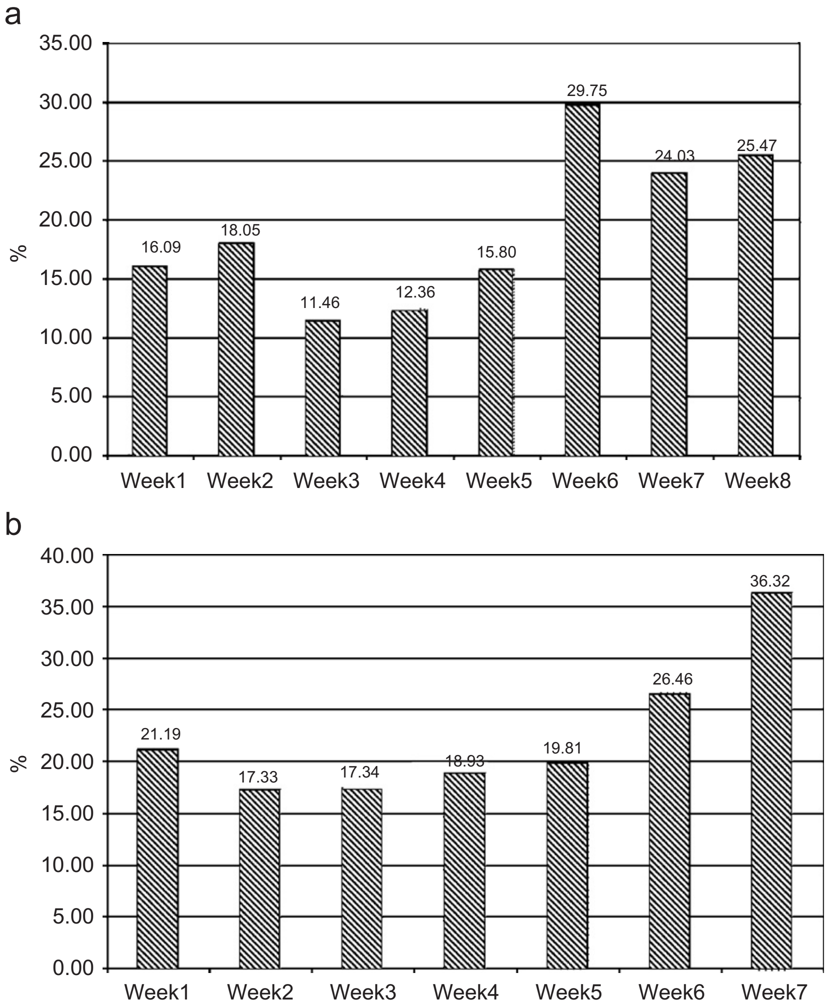

Fig. 6. Percentage of redirecting URLs for data sets M1 (a) and L (b).

the main driver behind e-mail spam. This trend is expected to continue, as the World Wide Web becomes more ubiquitous and better integrated with e-mail systems; also, as fraudulent activities such as phishing and Web spamming are spread further on the Internet.

- Each spam e-mail carries in its body 5–6 URL addresses on average. Nevertheless, there is a high variability in the distribution of URL addresses per spam message: a small percentage of spam messages carry several hundred URLs in their bodies, whereas a much larger percentage of e-mails carries only 1–2 URL addresses. Therefore, the *mere number of URLs found inside an e-mail message cannot be used effectively to capture the majority of spam messages*; here, we make the assumption that e-mails carrying a very large number of URL addresses have a very small probability of being non-spam.

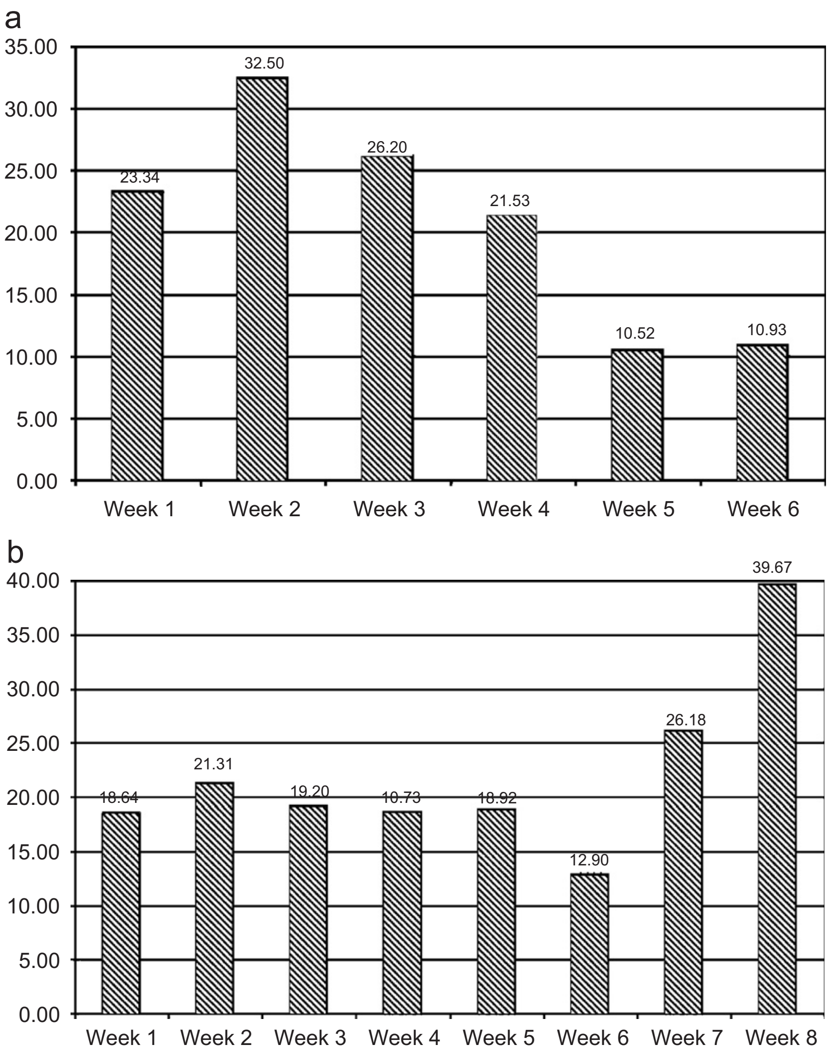

Fig. 7. Percentage of redirecting URLs for data sets M2 (a) and S (b).

- Around *two-thirds* (64.61–77.63%) *of the URL addresses found inside spam messages are encoded as links to Web resources*. The remaining one-third is used primarily to enrich the presentation of spam messages in HTML-enabled e-mail clients through the inclusion of images, multimedia files, etc.

Focusing on the characteristics of the URL addresses found inside spam messages, we conclude that:

- The *majority of these URL addresses correspond to static resources*, with a sizeable minority of 17.29–28.77% being dynamic. Over 90% of spam-advertised URL addresses belong to the `.com`, `.info`, `.org`, and `.net` domains.

- *The encoding of nearly all spam-advertised URL addresses includes the domain name of some host component*; therefore, URL addresses that are real and operational must be hosted on servers registered with Internet's DNS.
- Around 75% *of spam-advertised URLs have a path length at most equal to* 3. Therefore, one out of four URLs can be characterized as rather non-mnemonic and/or belonging to large web site hierarchies.
- Often, spam messages carry reputable URLs in an effort to evade spam filters, to confuse e-mail recipients, and/or to achieve phishing or Web spamming. *At least* 3.5–8% *of the host-names of spam-advertised URL addresses can be considered reputable.*
- The dissemination of a specific URL address through spam e-mail is *not* a recurrent activity: *the large majority of URL addresses that appear in spam messages do not appear again in subsequent spam campaigns*. Furthermore, accessibility tests show that nearly 50% of spam-advertised URLs have a lifetime that is less than a couple of weeks. Also, our tests show that trying to resolve a sizeable percentage of spam-advertised URLs results to an HTTP redirection.

6. Conclusions

The distribution of URL addresses to e-mail recipients is becoming the root cause behind the existence and the expansion of spam e-mail. URL addresses are found in all kinds of spam, from promotional to fraudulent (phishing and Web spamming). Therefore, spamming is essentially an incessant campaign that advertises URL addresses at a massive scale and at minimum cost for the advertisers (spammers) and those advertised. There are, however, notable differences between spamming and conventional advertising campaigns. The latter are paid, mediated, forms of communication from *identifiable* sources, designed to persuade as many receivers as possible to take some action, *now or in the future* (Richards and Curran, 2002). Conventional campaigns seek to make brand names of products and services well known and easily recognizable. In contrast, spamming campaigns promote URL addresses that are intensionally *short-lived*, *elusive*, and therefore *not easily identifiable*. Still, this information must be presented in an attractive and believable manner, so as to lure receivers into *immediate* action.

So far, efforts to address the problem of spam have focused on inventing effective techniques for detecting and filtering spam e-mails at the recipient side, and in trying to keep spam filters up-to-date about the new exploits devised by spammers that seek to escape detection. Our conjecture is that, besides filtering, we need to shift the focus of the anti-spam effort to the root-cause of spam, i.e., to spam-advertised sites. Our study shows, however, that this is challenging since spam-advertised sites are a moving and constantly changing target. Therefore, schemes to blacklist dubious URL addresses through collaborative filtering, such as those deployed by Surbl and used by SpamAssassin, may not be effective: spammers will be able to set up and distribute new URLs faster than the updating of blacklists. Also, the definition of what constitutes a dubious URL cannot be unequivocal; therefore, blacklisting can have serious side effects, especially if it turns against sites whose URL addresses were inserted in spam messages without their consent and against their will. Similar concerns are raised with the implementation of countermeasures against spam-advertised sites, such as duping (Nelson et al., 2006). To address these concerns, we are investigating the application of distributed, collaborative voting mechanisms for the identification and isolation of dubious URL addresses that adopt spamming as an advertising mechanism.

References

Abadi M, Birrell A, Burrows M, Dabek F, Wobber T. Bankable postage for network services. In: Advances in computing science—ASIAN 2003 programming languages and distributed computation, Eighth Asian computing science conference, Mumbai, India, December 10–14, 2003, Proceedings, Lecture notes in computer science, vol. 2896. Springer, Berlin; 2003. p. 72–90.

Albrecht K, Burri N, Wattenhofer R, Spamato. An extendable spam filter system. In: Proceedings of the second conference on email and anti-spam (CEAS05), July 2005. Available online at ⟨http://www.ceas.cc/papers-2005/156.pdf⟩ (last accessed June 2006).

Androutsopoulos I, Magirou EF, Vassilakis DK. A game theoretic model of spam e-mailing. In: Proceedings of the second conference on email and anti-spam (CEAS05), July 2005. Available online at ⟨http://www.ceas.cc/papers-2005/113.pdf⟩ (last accessed June 2006).

Arora V. The CAN-SPAM act: an inadequate attempt to deal with a growing problem. Columbia Journal of Law and Social Problems 2006;39(3):299–330.

Available online at: ⟨http://www.ferris.com/⟩.

Berners-Lee T, Fielding RT, Masinter L. Uniform resource identifier (URI): generic syntax. Internet draft, September 2004. ⟨http://gbiv.com/protocols/uri/rev-2002/draft-fielding-uri-rfc2396bis-07.html⟩ (last accessed: July 2006).

Brightmail inc. ⟨http://www.brightmail.com⟩.

Carpinter J, Hunt R. Tightening the net: a review of current and next generation spam filtering tools. Computers & Security 2006;25:566–78.

de Freitas S, Levene M. Spam on the Internet: is it here to stay or can it be eradicated? Technical Report, Joint Information Systems Committee Technology and Standards Watch Reports; 2004.

Drake CE, Oliver JJ, Koontz EJ. Anatomy of a phishing email. In: Proceedings of the first conference on email and anti-spam (CEAS04), July 2004. Available online at: ⟨http://www.ceas.cc/papers-2004/114.pdf⟩.

Gomes LH, Cazita C, Almeida JM, Almeida V, Meira Jr W. Characterizing a spam traffic. In: Proceedings of the fourth ACM SIGCOMM conference on internet measurement. ACM; 2004. p. 356–69.

Gomes LH, Almeida RB, Bettencourt LMA, Almeida V, Almeida JM. Comparative graph theoretical characterization of networks of spam and legitimate email. In: Proceedings the second conference on email and anti-spam (CEAS05), July 2005. Available online at ⟨http://www.ceas.cc/papers-2005/131.pdf⟩ (last accessed June 2006).

Goodman J, Heckerman D, Rounthwaite R. Stopping spam. Scientific American, April 2005.

Graham-Cumming J. The Spammers' compendium. Spam conference; 2003. Available online at: ⟨http://www.jgc.org/tsc/⟩.

Hird S. Technical solutions for controlling spam. In: Proceedings of the annual technical conference of the Australian UNIX and open systems user group; 2002.

Hulten G, Penta A, Seshadrinathan G, Mishra M. Trends in spam products and methods. In: Proceedings of the first conference on email and anti-spam (CEAS04), July 2004. Available online at: ⟨http://www.ceas.cc/papers-2004/165.pdf⟩.

Ioannidis J. Fighting spam by encapsulating policy in email addresses. In: Proceedings of the network and distributed system security symposium, NDSS; 2003.

Iwanaga M, Tabata T, Sakurai K. Evaluation of anti-spam method combining Bayesian filtering and strong challenge and response. In: Proceedings of IASTED international conference on communication, network, and information security (CNIS 2003), vol. 12; 2003. p. 214–9.

Klensin J. Simple mail transfer protocol; 2001. IETF, RFC 2821. ⟨http://www.ietf.org/rfc/rfc2821.txt⟩.

Koutsioupis C. Anti-spam at the University of Cyprus. Available online at: ⟨http://www.ucy.ac.cy/security/docs/WhydoIgetSPAM.pdf⟩ (last accessed: July 2006).

Kraut RE, Sunder S, Telang R, Morris J. Pricing electronic mail to solve the problem of spam. Human-Computer Interaction 2005;20(1–2):195–223.

Lawton G. E-mail authentication is here, but has it arrived yet? Computer 2005;38(11):17–9.

Li F, Hsieh M-H. An empirical study of clustering behavior of spammers and group-based anti-spam strategies. In: Proceedings of the third conference on email and anti-spam (CEAS06), July 2006. Available online at ⟨http://www.ceas.cc/2006/1.pdf⟩ (last accessed August 2006).

Moustakas E, Ranganathan C, Duquenoy P. Combating spam through legislation: a comparative analysis of US and European approaches. In: Proceedings of the second conference on email and anti-spam (CEAS05), July 2005. Available online at ⟨http://www.ceas.cc/papers-2005/146.pdf⟩ (last accessed June 2006).

Nelson PC, Dallmeyer KP, Szybalski LM, Palarz TP, Wieher M. Spamalot: a toolkit for consuming spammers resources. In: Proceedings of the third conference on email and anti-Spam (CEAS06), July 2006. Available online at ⟨http://www.ceas.cc/2006/6.pdf⟩ (last accessed August 2006).

Nucleus research. ⟨http://www.nucleusresearch.com⟩.

ODP: Open Directory Project. Available online at: ⟨http://www.dmoz.org/⟩.

Park SY, Kim JT, Kang SG. Proposal of a new effective spam mail regulation. In: The seventh international conference on advanced communication technology, ICACT 2005, vol. 2; 2005. p. 1081–4.

Postel JB. Simple mail transfer protocol, 1982. IETF, RFC 821. ⟨http://www.ietf.org/rfc/rfc821.txt⟩.

Prince MB, Holloway L, Keller AM. Understanding how spammers steal your e-mail address: an analysis of the first six months of data from project honey pot. In: Proceedings of the second conference on email and anti-spam (CEAS05), July 2005. Available online at ⟨http://www.ceas.cc/papers-2005/163.pdf⟩ (last accessed June 2006).

Pu C, Webb S. Observed trends in spam construction techniques: a case study of spam evolution. In: Proceedings of the third conference on email and anti-spam (CEAS06), July 2006. Available online at ⟨http://www.ceas.cc/2006/4.pdf⟩ (last accessed August 2006).

Richards JI, Curran CM. Oracles on 'advertising': searching for a definition. Journal of Advertising 2002;31(2):63–77.

Roman R, Zhou J, Lopez J. An anti-spam scheme using pre-challenges. Computer Communications 2006;29:2739–49.

Sorkin DE. Spam laws. Available online at: ⟨http://www.spamlaws.com/⟩ (last revised March 2007).

SpamArchive. Available online at: ⟨http://www.spamarchive.org/⟩.

Spinellis D. The decay and failures of web references. Communications of the ACM 2003;46(1):71–7.

Surbl—spam uri realtime blocklists. ⟨http://www.surbl.org/⟩ (last accessed: August 2006).

The Apache SpamAssassin Project. ⟨http://spamassassin.apache.org⟩ (last accessed: July 2006).

The Spamhaus Project. Increasing spam threat from proxy hijackers, ⟨http://www.spamhaus.org/news.lasso?article=156⟩, February 2005.

The Spamhaus Project. The definition of spam. ⟨http://www.spamhaus.org/definition.html⟩ (last accessed July 2006).

Webb S, Caverlee J, Pu C. Introducing the webb spam corpus: using email spam to identify web spam automatically. In: Proceedings of the third conference on email and anti-spam (CEAS06), July 2006. Available online at ⟨http://www.ceas.cc/2006/1.pdf⟩ (last accessed August 2006).

W3C Technical Architecture Group. Architecture of the World Wide Web, Volume One, W3C Recommendation, December 2004. Available at: ⟨http://www.w3.org/TR/webarch/⟩.

Zeinalipour-Yazti D, Dikaiakos MD. Design and implementation of a distributed crawler and filtering processor. In: Halevy A, Gal A, editors. Proceedings of the fifth international workshop on next generation information technologies and systems (NGITS 2002), Lecture notes in computer science, vol. 2832. Springer, Berlin, June 2002. p. 58–74.