Characterizing Web Spam Using Content and HTTP Session Analysis

Steve Webb
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
webb@cc.gatech.edu

James Caverlee
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
caverlee@cc.gatech.edu

Calton Pu
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
calton@cc.gatech.edu

ABSTRACT

Web spam research has been hampered by a lack of statistically significant collections. In this paper, we perform the first large-scale characterization of web spam using content and HTTP session analysis techniques on the Webb Spam Corpus – a collection of about 350,000 web spam pages. Our content analysis results are consistent with the hypothesis that web spam pages are different from normal web pages, showing far more duplication of physical content and URL redirections. An analysis of session information collected during the crawling of the Webb Spam Corpus shows significant concentration of hosting IP addresses in two narrow ranges as well as significant overlaps among session header values. These findings suggest that content and HTTP session analysis may contribute a great deal towards future efforts to automatically distinguish web spam pages from normal web pages.

1. INTRODUCTION

Web spam has grown to a significant percentage of all web pages (between 13.8% and 22.1% of all web pages [2, 8]), threatening the dependability and usefulness of web-based information in a manner similar to how email spam has affected email. Unfortunately, previous research on the nature of web spam [2, 5, 8, 10, 11, 13] has suffered from the difficulties associated with manually classifying and separating web spam pages from legitimate pages. As a result, these previous studies have been limited to a few thousand web spam pages, which is insufficient for an effective content analysis (as customarily performed in email spam research).

In this paper, we provide the first large-scale experimental study of web spam pages by applying content and HTTP session analysis techniques to the Webb Spam Corpus [12] – a collection of almost 350,000 web spam examples that is two orders of magnitude larger than the collections used in previous evaluations. Our main hypothesis in this study is that web spam pages are fundamentally different from "normal" web pages. To evaluate this hypothesis, we characterize the content and HTTP session properties of web spam pages using a variety of methods. The web spam content analysis is composed of two parts. The first part quantifies the amount of duplication present among web spam pages. Previous studies [1, 4, 6] have shown that only about two thirds of all web pages are unique; thus, we expected to find a similar degree of duplication among our web spam pages. To evaluate duplication in the corpus, we constructed clusters (equivalence classes) of duplicate or near-duplicate pages. Based on the sizes of these equivalence classes, we discovered that duplication is twice as prevalent among web spam pages (i.e., only about one third of the pages are unique).

The second part of the content analysis focuses on a categorization of web spam pages. Specifically, we identify five important categories of web spam: **Ad Farms**, **Parked Domains**, **Advertisements**, **Pornography**, and **Redirection**. The **Ad Farms** and **Parked Domains** categories consist of pages that are comprised exclusively of advertising links. These pages exist solely to generate traffic for other sites and money for web spammers (through pay-per-click advertising programs). The **Advertisements** category contains pages that advertise specific products and services, and the pages in the **Pornography** category are pornographic in nature. The **Redirection** category consists of pages that employ various redirection techniques. Within the **Redirection** category, we identify seven redirection techniques (HTTP-level redirects, 3 HTML-based redirects, and 3 JavaScript-based redirects), and we find that 43.9% of web spam pages use some form of HTML or JavaScript redirection.

The third component of our research is an evaluation of the HTTP session information associated with web spam. First, we examine the IP addresses that hosted our web spam pages and find that 84% of the web spam pages were hosted on the 63.* – 69.* and 204.* – 216.* IP address ranges. Then, we evaluate the most commonly used HTTP session headers and values. As a result of this evaluation, we find that many web spam pages have similar values for numerous headers. For example, we find that 94.2% of the web spam pages with a "Server" header were hosted by Apache (63.9%) or Microsoft IIS (30.3%). These results are particularly interesting because they suggest that HTTP session information might be extremely valuable for automatically distinguishing between web spam pages and normal pages.

The rest of the paper is organized as follows. Section 2 describes our web spam corpus and summarizes its collection methodology. In Section 3, we report the results of a content analysis of web spam, which consists of two parts. The first part evaluates the amount of duplication that appears in web spam. The second part identifies concrete web spam cate-

CEAS 2007 - Fourth Conference on Email and Anti-Spam, August 2-3, 2007, Mountain View, California USA

gories and provides an extensive description of the redirection techniques being used by web spammers. In Section 4, we report the results of an analysis of web spam HTTP session information, which identifies the most common hosting IP addresses and HTTP header values associated with web spam. Section 5 summarizes related work, and Section 6 concludes the paper and provides future research directions.

2. THE WEBB SPAM CORPUS

In our previous research [12], we developed an automatic technique for obtaining web spam examples that leverages the presence of URLs in email spam messages. Specifically, we extracted almost 1.2 million unique URLs from more than 1.4 million email spam messages. Then, we built a crawler to obtain the web pages that corresponded to those URLs. Our crawler attempted to access each of the URLs; however, many of the URLs returned HTTP redirects (i.e., 3xx HTTP status codes). The crawler followed all of these redirects until it finally accessed a URL that did not return a redirect.

Our crawler obtained two types of information for every successfully accessed URL (including those that returned a redirect): the HTML content of the page identified by the URL and the HTTP session information associated with the page request transaction. As a result, we created a file for every successfully accessed URL that contains all of this information. After our crawling process was complete, we had 348,878 web spam pages and 223,414 redirect files (i.e., files that correspond to redirect responses). These files are collectively referred to as the Webb Spam Corpus[1], and they provide the basis for our analysis in this paper. For a more detailed description of our collection methodology and the format of the files in the Webb Spam Corpus, please consult [12].

We acknowledge that our collection of web spam examples is not representative of all web spam; however, it is two orders of magnitude larger than any other available source of web spam to date, and as such, it currently provides the most realistic snapshot of web spammer behavior. Thus, although the characteristics of our corpus might not be indicative of all web spam, our observations still provide extremely useful insights about the techniques being employed by web spammers.

3. CONTENT ANALYSIS

In this section, we provide the results of our large-scale analysis of web spam content. This analysis consists of two parts. The first part, discussed in Section 3.1, quantifies the amount of duplication present among web spam pages. The second part, discussed in Section 3.2, presents a categorization of web spam pages.

3.1 Web Spam Duplication

Previous research has shown that approximately one third of all web pages are duplicates or near-duplicates of a web page in the remaining two thirds [1, 4, 6]. To evaluate the amount of duplication among web spam pages, we analyzed three forms of duplication in our corpus: URL duplication, content duplication, and content near-duplication.

[1]The Webb Spam Corpus can be found at `http://www.webbspamcorpus.org/`.

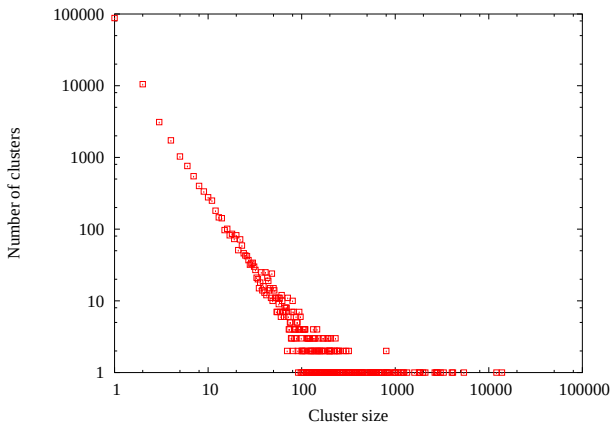


Figure 1: Number and size of the shingling clusters.

In our previous work [12], we identified the existence of duplicate URLs in the corpus (i.e., multiple web spam pages with the same URL), and we explained that these duplicate URLs are the result of multiple unique HTTP redirect chains that lead to the same destination. Specifically, we found that the corpus contains 263,446 unique URLs, which means about one fourth of the web spam pages have a URL that is the same as one of the web spam pages in the remaining three fourths.

To identify content duplication, we computed MD5 hashes for the HTML content of all of the web spam pages in our corpus. After evaluating these results, we found 202,208 unique MD5 values. Thus, 146,670 of the web spam pages (42%) have the exact same HTML content as one of the pages in a collection of 202,208 unique web spam pages. Many of these duplicates are explained by the URL duplication that exists in the corpus (described above), but since each of the duplicate URLs represents a distinct entry point (i.e., a unique HTTP redirect chain) to a given page, we consider them to be functionally equivalent to content duplicates.

To evaluate the amount of near-duplication in our corpus, we used the shingling algorithm that was developed by Fetterly et al. [3, 4, 6] to construct equivalence classes of duplicate and near-duplicate web spam pages. First, we preprocessed every web spam page in the corpus. Specifically, the HTML tags in each page were replaced by white space, and every page was tokenized into a collection of words, where a word is defined as an uninterrupted series of alphanumeric characters.

Then, for every page, we created a fingerprint for each of its n words using a Rabin fingerprinting function [9] (with a degree 64 primitive polynomial p_A). Once we had the n word fingerprints, we combined them into 5-word phrases. The collection of word fingerprints was treated like a circle (i.e., the first fingerprint follows the last fingerprint) so that every fingerprint started a phrase, and as a result, we obtained n 5-word phrases. Next, we generated n phrase fingerprints for the n 5-word phrases using a Rabin fingerprinting function (with a degree 64 primitive polynomial p_B). After we obtained the n phrase fingerprints, we applied 84 unique Rabin fingerprinting functions (with degree 64 primitive polynomials p_1, ..., p_{84}) to each of the n phrase fingerprints. For every one of the 84 functions, the smallest

Table 1: Most of the 50 largest equivalence classes.

Rank	Size	Categories	Most Common Domain (Count)
1	13,806	Ad Farms, Redirection	techbuyer.com (6,877)
2	12,090	Redirection	www.bizrate.com (578)
3	5,420	Pornography, Redirection	www.ezinetracking.com (832)
4	4,138	Parked Domains, Redirection	migada.com (1,553)
5, 8, 41, 46	4,034, 2,837, 594, 567	Ad Farms, Redirection	mx07.com (4,034)
6	3,294	Parked Domains, Redirection	pntaa.com (452)
7	3,053	Advertisements	www.macmall.com (3,053)
9, 10, 11, 21	2,791, 2,749, 2,646, 1,142	Ad Farms	ew01.com (6,579)
12	2,096	Advertisements	yoursmartrewards.com (2,096)
13	1,983	Parked Domains, Redirection	www.optinspecialists.info (426)
14, 32	1,837, 784	Ad farms, Redirection	click.recessionspecials.com (1,836)
15	1,828	Redirection	mailer.ebates.com (1,828)
17	1,606	Parked Domains, Redirection	www.flgstff.com (777)
18	1,336	Parked Domains	www.gibox.com (133)
19, 38	1,239, 622	Ad Farms, Redirection	lb3.netster.com (1,821)
22	1,069	Advertisements	morozware.com (62)
23, 28, 30, 34, 43, 47	1,014, 822, 802, 712, 583, 562	Ad Farms, Redirection	www.thehdhd.com (1,014)
24	995	Advertisements	www.personaloem.info (995)
25	977	Advertisements	ratedoem.info (112)
26	857	Parked Domains	pn01.com (402)
27	831	Advertisements	www.netidentity.com (831)
33	724	Parked Domains, Redirection	apps5.oingo.com (724)
35	674	Parked Domains, Redirection	www.demote.com (2)
37	630	Advertisements	www.pimsleurapproach.com (630)
42	588	Parked Domains, Redirection	new.hostcn2.com (84)
44	580	Parked Domains, Redirection	www.zudak.com (282)
45	570	Pornography	www.centerfolds4free.com (29)
48	554	Parked Domains, Redirection	landing.domainsponsor.com (542)
49, 50	536, 532	Ad Farms	dbm.consumer-marketplace.com (451)

of the n fingerprints was stored. Once this process was complete, each web spam page was reduced to 84 fingerprints, which are referred to as that page's *shingles*.

Once all of the pages were converted to a collection of 84 shingles, we clustered the pages into equivalence classes (i.e., clusters of duplicate or near-duplicate pages). Two pages were considered duplicates if all of their shingles matched, and they were near-duplicates if their shingles agreed in two out of the six possible non-overlapping collections of 14 shingles. For a more detailed description of this shingling algorithm, please consult [3, 4, 6].

After running the shingling algorithm on our web spam pages, we were left with 109,157 unique clusters of duplicate and near-duplicate pages. Figure 1 shows the distribution of the number and size of the shingling clusters. From the figure, we see that 87,819 clusters contain a single web spam page (the point at the top-left of the figure). These pages are truly unique because none of the other pages in the corpus duplicate their content. On the opposite end of the spectrum, one cluster contains 13,806 web spam pages (the point at the bottom-right of the figure). All of these pages are either duplicates or near-duplicates of each other. The main observation from these results is that two thirds of web spam pages are duplicates or near-duplicates of a web spam page in the remaining one third. Thus, duplication is twice as prevalent among web spam pages as it is among web pages in general.

3.2 Web Spam Categorization

To categorize the content of web spam pages, we manually investigated the 50 largest equivalence classes (as defined by the clustering algorithm described in Section 3.1). These 50 clusters contain 93,595 web spam pages, accounting for 26.8% of the web spam pages in our corpus. Based on our investigation, we identified five categories that describe the pages we reviewed:

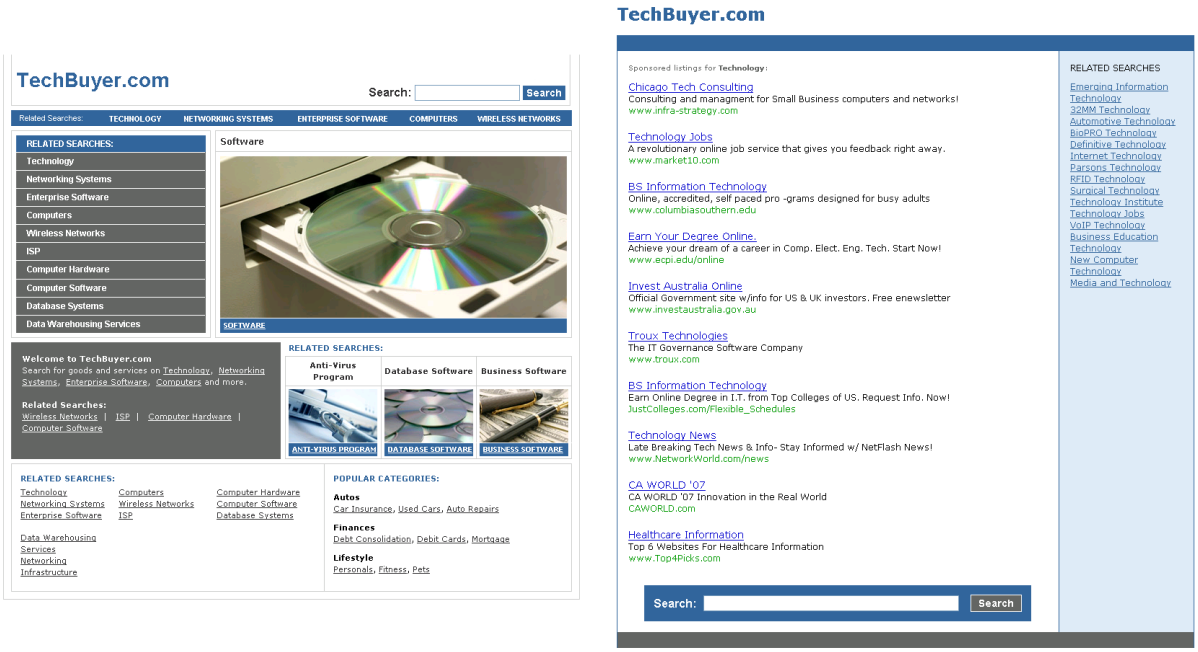
- **Ad Farms**
- **Parked Domains**
- **Advertisements**
- **Pornography**
- **Redirection**

These categories help describe the purpose of the pages in each of the shingling clusters as well as the goals of the spammers who created them. Table 1 lists most of the 50 largest equivalence classes. For each cluster, we provide its rank (in terms of size), size, and categorization. We also provide the most common domain name found in each of the listed clusters. In the remainder of this section, we will describe our five web spam categories and detail the important characteristics of their representative pages.

3.3 Ad Farms

Ad farms are pages that only contain advertising links (usually in the form of ad listings). These pages are of little value to visitors because they do not contain any original content. Additionally, many (if not all) of the links that appear in their ad listings are low quality because they are not ordered by traditional ranking algorithms (e.g., Google's PageRank). In fact, a large fraction of the links are controlled by the web spammers themselves.

To deceive visitors into believing ad farms are valuable and legitimate, most web spammers create elaborate entry pages that appear to be legitimate directories. Figure 2(a) shows an example of an ad farm's entry page. Once visitors



(a) (b)

Figure 2: Ad Farm Examples.

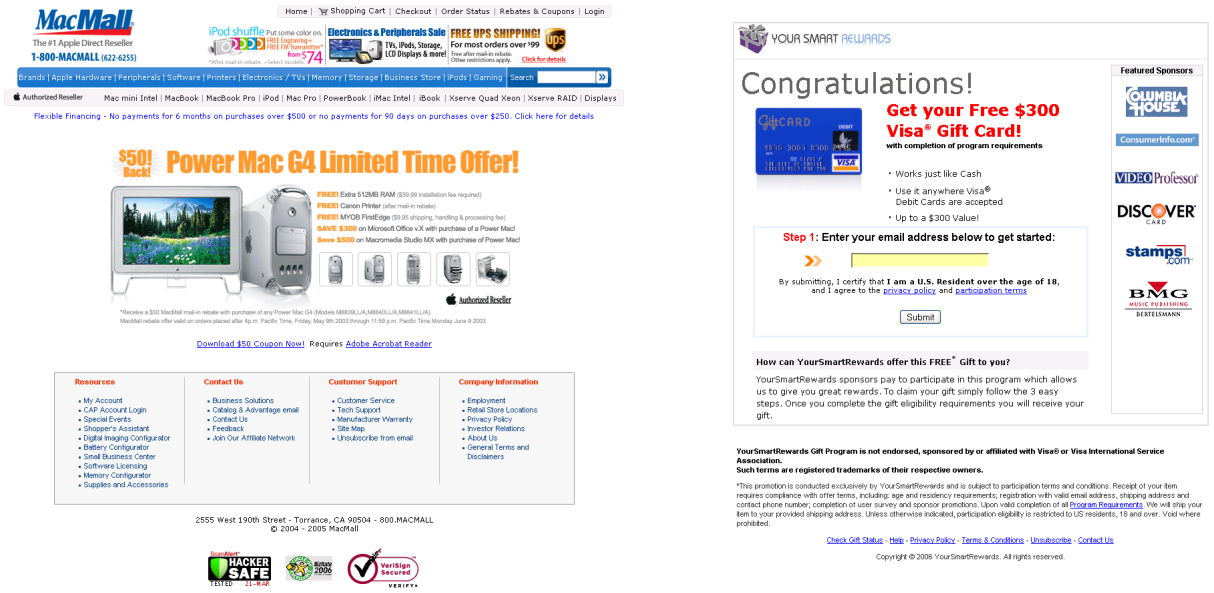
click on one of the categories in these fake directories, they are typically redirected to an ad listing. Figure 2(b) shows an example of an ad listing that is returned when a user clicks on one of the links depicted in Figure 2(a). The links that are displayed in these ad listings are typically obtained from an ad syndicator, but the HTML structures used by ad farms are created by web spammers.

Ad farms are extremely common in our corpus; 21 of the 50 largest equivalence classes are composed exclusively of ad farms. The 1st cluster contains pages that use JavaScript location objects and meta refresh tags to redirect users to an ad farm. Out of the cluster's 13,806 pages, only 1,821 unique hostnames are represented, and those hostnames consist of various subdomains off of 58 unique domain names (e.g., `valuevalet.seeq.com`, `happy-thoughts.seeq.com`, `inraw.seeq.com`, etc.). Spammers create numerous subdomains for three reasons. First, every subdomain represents a new address on the web. As a result, spammers can use each of these addresses as another unique entry point into an ad farm. Second, creating multiple subdomains is far less expensive than creating an equivalent number of unique domain names. Third, the name of a given subdomain can help influence the actual advertising links that are displayed in the ad farm. Thus, spammers can maximize the coverage of their ad farms by using numerous, non-overlapping subdomain names.

The pages in the 5th, 8th, 41st, and 46th clusters use a frameset (consisting of two frames) to redirect users to an ad farm. The first frame loads fake directory content (i.e., various categories and subcategories, which lead to corresponding ad listings), and the second frame loads a search field that allows users to search for specific ad farm content. Each of the four clusters contains numerous subdomains off of a specific domain name. The 5th cluster uses `mx07.com`; the 8th cluster uses `emailcourrier.com`; the 41st cluster uses `yearendsaver.com`, and the 46th cluster uses `brightermail.com`. We grouped these clusters together in Table 1 because all of their pages have similar HTML structures and were hosted at the same IP address (64.69.68.141). The only difference between the clusters is the actual text that is used in the ad links on their pages. For example, the ad farms in the 8th cluster are primarily concerned with email-related ads, whereas the ad farms in the 41st cluster are focused on accounting-related ads.

The 9th, 10th, 11th, and 21st clusters also contain pages that display ad farms. However, unlike the previous examples, these pages do not use redirection techniques. They display the ad farms directly. Similar to the last group of clusters, we grouped these clusters together in Table 1 because their pages use the same HTML structure, and all of their pages were hosted at one of two IP addresses (204.251.15.193 and 204.251.15.194). The clusters also use a number of subdomains off of a specific domain name. The 9th, 11th, and 21st clusters use `ew01.com`, and the 10th cluster uses `www.msstd.com`.

The 23rd, 28th, 30th, 34th, 43rd, and 47th clusters are also particularly interesting because they contain an additional level of redirection that is missing from the files in the previous clusters. First, the pages in these clusters redirect users to `lb1.youbettersearch.com`. To accomplish this initial redirection, some of the pages use the replace method for JavaScript location objects, and others use meta refresh tags. Then, that hostname uses another level of redirection to obtain the content for its ad farms.



(a) (b)

Figure 3: Advertisement Examples.

3.4 Parked Domains

Domain parking services allow individuals to display a web page that acts as a place holder for newly registered domains. A popular choice for this place holder is an ad listing because it allows an individual to monetize a domain with minimal effort. Unfortunately, web spammers quickly exploited this opportunity and began parking hundreds of thousands of domains with ad listings.

Parked domains are functionally equivalent to ad farms. They both use ad syndicators as their primary sources of content, and they both provide little to no value to their visitors. However, parked domains possess two unique characteristics that distinguish them from ad farms. First, parked domains rely on domain parking services (e.g., `apps5.oingo.com`, `searchportal.information.com`, `landing.domainsponsor.com`, etc.) to provide their entire advertising infrastructure (the HTML structure of the entry pages as well as the content for the ad listings). Second, domain parkers are typically much more motivated than ad farmers to sell the domains they are using to display ad links. In many cases, parked domains even include links with phrases such as "Offer To Buy This Domain" or "Purchase This Domain" to persuade visitors to buy the domain.

Eight of the clusters (#4, #6, #13, #17, #35, #42, #44, and #48) contain pages that use various techniques to redirect users to ad listings, which are provided by various domain parking services. The pages in the 4^{th} cluster use a frameset (consisting of one frame) to redirect users to `apps5.oingo.com`, while the pages in the 6^{th} cluster use the replace method for JavaScript location objects to accomplish the redirection. The 13^{th} and 17^{th} clusters consist of pages that use a frameset to redirect users to a handful of different domain parking services. For both clusters, `searchportal.information.com` is the most commonly used service. The 35^{th} and 42^{nd} clusters both contain pages that redirect users to `apps5.oingo.com`. The pages in the 35^{th} cluster use a frame to accomplish the redirection, and the pages in the 42^{nd} cluster use an iframe. The 44^{th} cluster contains pages that use a frameset (consisting of two frames) to redirect users to `landing.domainsponsor.com` or `apps5.oingo.com`. The 48^{th} cluster also contains pages that rely on domain parking services (`landing.domainsponsor.com` and `searchportal.information.com`). However, unlike the other clusters, which contain pages that redirect users with content-based redirection, these pages are obtained by following HTTP redirects.

Three of the clusters (#18, #26, and #33) contain pages that were generated by DNS registrars. The pages in the 18^{th} and 26^{th} clusters were generated by registrars that provide their own domain parking services (GoDaddy and DomainDiscover, respectively). These pages contain a combination of syndicated ad listings and registrar-specific advertisements. The pages in the 33^{rd} cluster were generated by a DNS registrar (Network Solutions); however, the pages rely on a domain parking service (`apps5.oingo.com`) for their content.

3.5 Advertisements

In addition to ad farms and parked domains, which display ad listings for various web pages, the corpus also contains numerous pages that advertise specific products and services. Ad farms and parked domains are essentially directories for advertisements, and these advertisement pages are examples of the types of pages being advertised in those directories. Two examples of these advertisement pages are shown in Figures 3(a) and 3(b).

The 7^{th} cluster contains pages that display advertisements for software and hardware products that are being sold at `macmall.com`. The pages in the 12^{th} cluster offer free gift

cards in exchange for a user's personal information (e.g., email address), and they all use the same domain name (yoursmartrewards.com). The 15th cluster consists of pages that use meta refresh tags to redirect users to http://www.ebates.com. This site is a well-known advertiser and adware/spyware distributor. The pages in the 22nd and 25th clusters all display advertisements for various software packages, and the 24th cluster contains pages that display advertisements for "Pink Floyd Products." The pages in the 27th cluster display advertisements for various domain names being sold by www.netidentity.com. These pages are different from parked domains because the pages are not concerned with generating ad revenue – their sole purpose is selling domains. The 37th cluster contains pages that display advertisements for various foreign language instructional materials (e.g., books, tapes, videos, etc.) being sold at www.pimsleurapproach.com.

3.6 Pornography

Although only 2 of the 50 largest equivalence classes consist of pornography-related pages, those 2 clusters account for almost 2% of the entire corpus. The 3rd cluster contains 5,420 pages that execute "drive-by advertising." Specifically, these pages generate pop-up advertisements for www.freeezinebucks.com, and then, they use meta refresh tags to redirect users to various pornographic web sites (e.g., www2.grandegirls.com, www.wannawatch.com, etc.). The 45th cluster contains 570 pages that prompt the user to log in to a pornographic site (e.g., www.brunettes4free.com, www.girlgirl4free.com, etc.).

3.7 Redirection

Many web spammers use redirection to hide their spam content [7]. Thus, one of the most ubiquitous characteristics of web spam pages is their use of redirection. Table 1 shows that 27 of the 50 largest equivalence classes contain pages that utilize redirection techniques. In this section, we investigate the most popular techniques, and we present the most popular redirection destinations.

The easiest way to accomplish redirection is at the HTTP-level (i.e., returning a 3xx status code). As explained in our previous work [12], the Webb Spam Corpus contains 223,414 redirect files that represent examples of this type of HTTP redirection. All of these HTTP redirect files contain one of two 3xx status codes: 301 ("Moved Permanently") and 302 ("Found"). Aside from HTTP redirection, a number of content-based redirection techniques also exist. Based on our manual examination of the largest equivalence classes in the corpus, we identified six content-based redirection techniques that are repeatedly employed by web spammers. Three of these techniques are accomplished using HTML, and the other three are accomplished using JavaScript. The HTML techniques make use of meta refresh tags, frame tags, and iframe tags. The JavaScript techniques include assigning a URL to a location object, assigning a URL to a location object's href attribute, and passing a URL to the replace method of a location object.

Identifying examples of the HTML redirection techniques was fairly straightforward due to the syntactic properties of the HTML tags that are used (meta, frame, and iframe). Specifically, we wrote a custom HTML parser (based on Perl's HTML::Parser module) to identify these tags and extract the URLs being used for redirection. For the remainder of this paper, we will refer to these extracted URLs (and their corresponding hostnames) as *targets* of redirection.

Identifying examples of the JavaScript redirection techniques was significantly more challenging for a number of reasons. First, many of the pages in the corpus contain external JavaScript references that use relative addresses. These relative addresses rely on the existence of locally stored JavaScript scripts. However, the files in the corpus only contain HTML content and embedded JavaScript scripts (i.e., none of the external JavaScript scripts are stored locally). To solve this problem, we dynamically rewrote the HTML files, replacing the relative script addresses with absolute addresses. As a result, we were able to download the necessary external script files when they were needed.

Another challenge posed by the JavaScript techniques is the nondeterministic behavior of JavaScript script execution. Unlike the HTML techniques, which we easily identified with an HTML parser, the JavaScript techniques were often hidden by conditional statements or accomplished with additional levels of indirection (e.g., method calls). Additionally, many of the techniques used variables to assign the targets of redirection (as opposed to using direct assignments).

To overcome these obstacles, we dynamically rewrote the HTML files to trap JavaScript method calls (e.g., replace()) and assignments to important JavaScript objects and attributes (e.g., location and location.href) that dealt with redirection. Specifically, we replaced each redirection technique with an alert method. Then, to capture the targets of redirection, we passed the original redirection parameters as arguments to the alert method. As a result, the JavaScript redirection techniques were replaced as follows:

- *location = URL;* became *alert("location: URL");*
- *location.href = URL;* became *alert("location.href: URL");*
- *location.replace(URL);* became *alert("location.replace: URL");*

In each of the above replacements, *URL* could be a static string or a variable construction. Our HTML rewriting techniques were able to handle both of these cases.

After rewriting our corpus files, we used HtmlUnit 1.10[2] to create a custom WebClient that trapped alert method calls and parsed their arguments (i.e., the targets of redirection). Then, we used our WebClient to access the rewritten HTML files, execute their JavaScript scripts (using the Rhino JavaScript engine[3]), and capture the alert method calls that were generated by JavaScript redirection techniques. Finally, we extracted the redirection targets and converted any relative target addresses to absolute target addresses.

Based on our analysis, we discovered that the corpus contains 144,801 unique redirect chains, each containing an average of 1.54 HTTP redirects. Thus, 41.5% of the web spam pages were obtained by following a redirect chain. Additionally, of the 348,878 web spam pages, 153,265 (43.9%) use some form of HTML or JavaScript redirection. Only 1,304 of the 223,414 redirect files (0.6%) use HTML or JavaScript redirection techniques, but that is not surprising since most of those files only contain session information for HTTP redirects.

[2] http://htmlunit.sourceforge.net/
[3] http://www.mozilla.org/rhino/

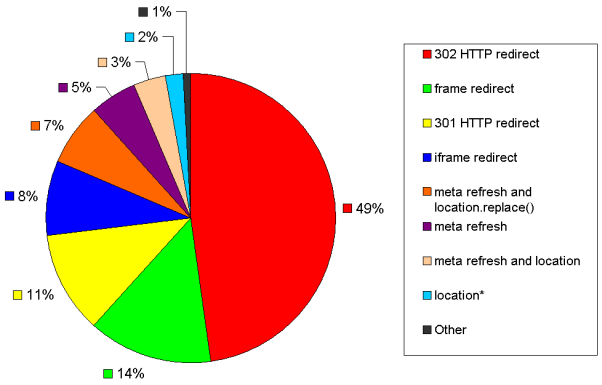


Figure 4: Relative frequency of redirection techniques.

Figure 4 shows a pie chart that breaks down the relative frequency of the redirection techniques across all of the corpus files (i.e., web spam pages and redirect files). HTTP redirection (without the use of any content-based techniques) is clearly the most popular form of redirection in the corpus, accounting for 60% of the redirections (49% for "Found" redirects and 11% for "Moved Permanently" redirects). The next most popular techniques involve only using HTML frame tags or HTML iframe tags. These techniques account for 14% and 8% of the redirections, respectively. Redirection using meta refresh tags appears in a variety of flavors. The most popular form of meta refresh redirection is accomplished in conjunction with the replace method of a JavaScript location object. This technique accounts for 7% of the redirections in the corpus. Files that exclusively use one of the three JavaScript techniques, which we grouped together as "location*" in the figure, account for 2% of the redirections. All of the other combinations of redirection techniques collectively account for 1% of the redirections.

Table 2 shows the hostnames that are most frequently the targets of redirection in our corpus. The first set of counts represent the combined view of all of the HTTP, HTML, and JavaScript redirection techniques we identified. This list consists of 2 ad farms (`lb1.youbettersearch.com` and `bluerocketonline.TechBuyer.com`), 2 advertisers (`ads2.drivelinemedia.com` and `login.tracking101.com`), and 1 domain parking service. The top 5 HTTP redirect targets are all advertisers. The top 5 frame redirect targets consist of 3 domain parking services (`apps5.oingo.com`, `searchportal.information.com`, and `landing.domainsponsor.com`), 1 ad farm (`click.recessionspecials.com`), and 1 parked domain. The top 5 iframe redirect targets consist of 1 ad farm (`lb3.netster.com`), 3 advertisers (`ads2.drivelinemedia.com`, `simg.zedo.com`, and `www.creativecow.net`), and 1 domain parking service (`apps5.oingo.com`). The top 5 meta refresh redirect targets consist of 2 ad farms (`lb1.youbettersearch.com` and `bluerocketonline.TechBuyer.com`), 2 advertisers (`www.ebates.com` and `biz.tigerdirect.com`), and 1 pornographer. The top 5 location* redirection targets consist of 2 ad farms (`lb1.youbettersearch.com` and `bluerocketonline.TechBuyer.com`) and 3 advertisers (`www.classmates.com`, `c.azjmp.com`, and `yoursmartrewards.com`).

Table 2: Most common targets of redirection.

Top 5 targets of redirection	
Hostname	Count
lb1.youbettersearch.com	44,334
ads2.drivelinemedia.com	15,798
bluerocketonline.TechBuyer.com	12,204
login.tracking101.com	11,639
apps5.oingo.com	10,153
Top 5 targets of HTTP redirection	
Hostname	Count
login.tracking101.com	11,639
www.macmall.com	8,895
cpaempire.com	5,350
mailer.ebates.com	3,656
click.be3a.com	2,488
Top 5 targets of frame redirection	
Hostname	Count
apps5.oingo.com	6,952
searchportal.information.com	5,796
landing.domainsponsor.com	5,256
click.recessionspecials.com	1,836
migada.com	1,553
Top 5 targets of iframe redirection	
Hostname	Count
ads2.drivelinemedia.com	15,798
simg.zedo.com	6,002
lb3.netster.com	4,961
apps5.oingo.com	3,201
www.creativecow.net	2,098
Top 5 targets of meta refresh redirection	
Hostname	Count
lb1.youbettersearch.com	22,278
bluerocketonline.TechBuyer.com	6,108
www.ebates.com	1,828
biz.tigerdirect.com	803
programs.weginc.com	722
Top 5 targets of location* redirection	
Hostname	Count
lb1.youbettersearch.com	22,056
bluerocketonline.TechBuyer.com	6,096
www.classmates.com	659
yoursmartrewards.com	427
c.azjmp.com	417

4. HTTP SESSION ANALYSIS

In addition to the HTML content of web spam pages, our corpus also contains the HTTP session information that was obtained from the servers that were hosting those pages. In this section, we characterize this session information, focusing on the most common server IP addresses and session header values.

4.1 Hosting IP Addresses

One of the most important pieces of HTTP session information is the IP address that hosted a given web spam page – the *hosting IP address*. Figure 5 shows the distribution of all of the hosting IP addresses in our corpus. This figure clearly shows that most of the hosting IP addresses were concentrated around a few IP address ranges. Specifically, the 63.* – 69.* and 204.* – 216.* IP address ranges account for 45.4% and 38.6% of the hosting IP addresses in the cor-

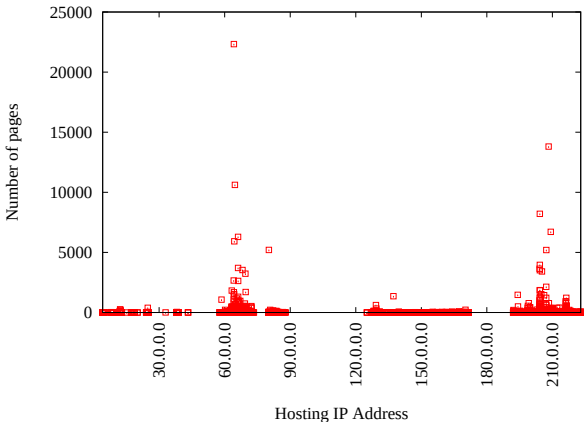


Figure 5: Number of pages being hosted by a single IP address.

Table 3: Top 10 hosting IP addresses.

Hosting IP Address	Count
64.225.154.135	22,332
208.254.3.166	13,806
64.69.68.141	10,615
204.251.15.194	8,211
209.233.130.40	6,713
66.116.109.62	6,294
64.40.102.44	5,923
80.245.197.244	5,206
207.219.111.23	5,201
204.251.15.193	3,963

pus, respectively (84%, collectively). Table 3 shows the 10 most popular hosting IP addresses, and all but one of them are in these IP address ranges.

Interestingly, these IP address ranges also include the two most popular web spam IP address ranges discussed in previous work by Wang et al. [11]. Specifically, they found that most of their spam examples were being hosted on two IP address ranges (64.111.* and 66.230.*). Our corpus includes 120 and 916 examples from those address ranges, respectively. The presence of these hosting IP addresses in our corpus reaffirms their results, and it also emphasizes the value of our corpus and the method used to obtain it.

4.2 HTTP Session Headers

In addition to the hosting IP addresses of web spam pages, our corpus also contains all of the HTTP session headers that were associated with the page request transaction. In this section, we identify the most commonly used headers as well as the most popular header values.

Many of the corpus files contained more than one value for a given header. In each of those cases, we concatenated all of the separate values into a comma delimited list. We did this because we wanted a one-to-one mapping for a file and each of its headers to simplify header comparisons from one file to another.

Table 4 shows the 10 most popular HTTP session headers in terms of the number of web spam pages that contain them. The table shows the number of pages each header appears in, the number of unique values each header has, and the most popular value for each of the headers. The "Content-Type" header is the most popular header, appearing in all 348,878 of the web spam pages in our corpus. As we explained in our previous work [12], the corpus only contains files with textual "Content-Type" values. Thus, the values for the "Content-Type" header are primarily "text/html" combined with permutations of various charset encodings (e.g., "iso-8859-1," "utf-8," etc.).

The "Server" header is the second most popular header, appearing in 343,168 of the web spam pages. This header is extremely important because it describes the web server that actually served a given web spam page. Microsoft IIS 6.0 is the most commonly used web server in our corpus (18.9% of the pages were hosted by it), but generally, Apache (63.9%) was used more frequently than Microsoft IIS (30.3%). Overall, these two web servers were clearly the most popular option among web spammers, accounting for 94.2% of the pages that contain a "Server" header.

5. RELATED WORK

Fetterly et al. [5] statistically analyzed two data sets of web pages (DS1 and DS2) using properties such as linkage structure, page content, and page evolution. They found that many of the outliers in the statistical distributions of these properties were web spam, and they manually identified 98 out of 1,286 web pages as spam. Ntoulas et al. [8] extended the work by investigating additional content-based features of a collection of 2,364 web spam pages (e.g., fraction of visible content, compressibility, independent n-gram likelihoods, etc.). Castillo et al. [2] also identified a few spam features (e.g., synthetic text, parked domains, etc.) using a collection of 1,447 manually labeled web spam pages. Our work differs from these previous evaluations in two very important ways. First, our characterization was performed on a collection of web spam pages that is two orders of magnitude larger than the collections used in previous studies. Second, we are the first to analyze the HTTP session information associated with web spam pages.

Wu and Davison [13] performed a preliminary evaluation of the redirection techniques used on the web. Specifically, they looked at HTTP redirection, meta refresh redirection, and two types of JavaScript redirection in a collection of unlabelled web pages. Wang et al. [11] took this work a step further by analyzing network redirection traffic from known spam domains to identify redirection URLs. In this paper, we provide an evaluation of the redirection techniques used by web spammers that enhances these previous studies in three ways. First, our analysis encompasses hundreds of thousands of web spam pages, whereas previous studies only used a few thousand pages. Second, we investigate a more comprehensive list of redirection techniques: HTTP redirection, 3 HTML-based techniques, and 3 JavaScript-based techniques. Third, previous research [7, 13] detailed the difficulties associated with identifying and processing JavaScript redirection techniques. We were able to overcome these difficulties by using sophisticated HTML rewriting techniques, and as a result, we are the first to present a large-scale evaluation of JavaScript-based redirection techniques.

Wang et al. [10] developed an automated approach for identifying typo-squatting domains, which are a specific type of parked domains. Using this approach, they found that a

Table 4: Top 10 HTTP session headers.

Header	Total Count	Unique Count	Most Popular Value (Count)
Content-Type	348,878	688	text/html (155,401)
Server	343,168	6,513	microsoft-iis/6.0 (64,787)
Connection	327,478	6	close (304,557)
X-Powered-By	209,215	261	asp.net (80,294)
Content-Length	162,532	31,232	1470 (6,115)
Cache-Control	148,715	548	private (69,571)
Set-Cookie	145,315	140,431	gx_jst=9fa7274e662d6164; path=/apps/system, gx_jst=9fa7274e662d6164 (626)
Link	142,785	15,573	</style/kentech.css>; rel="stylesheet"; type="text/css" (25,620)
Expires	93,477	25,056	mon, 26 jul 1997 05:00:00 gmt (18,933)
Pragma	75,435	32	no-cache (64,344)

handful of domain parking services are responsible for parking about 30% of the typo-squatting domains they identified. In our study, we also identified numerous examples of parked domains; however, our analysis was not confined to typo-squatting domains, and we investigated a significantly larger collection of web spam pages.

6. CONCLUSIONS AND FUTURE WORK

We have conducted the first large-scale experimental study of web spam through content and HTTP session analysis on the Webb Spam Corpus – a collection of almost 350,000 web spam pages. Our results are consistent with the hypothesis that web spam pages are fundamentally different from normal web pages. Specifically, we found that the rate of duplication among web spam pages is twice the duplication rate for normal web pages. Our content analysis also found five important categories of web spam: **Ad Farms**, **Parked Domains**, **Advertisements**, **Pornography**, and **Redirection**.

In addition to content analysis, we also performed HTTP session analysis on the session data that was collected during the construction of the Webb Spam Corpus. This session analysis showed two trends. First, the hosting IP addresses are concentrated in two narrow ranges (63.* – 69.* and 204.* – 216.*). Second, significant overlaps exist among the session header values. Both of these trends are consistent with the hypothesis that web spam pages are detectably different from normal web pages, in a way similar to the results of our content analysis.

Although previous web spam research has focused primarily on link analysis, our results suggest that content and HTTP session analysis techniques can contribute greatly towards distinguishing web spam pages from normal web pages. This is a promising result because these techniques can become new weapons in our ongoing battle against web spam.

7. ACKNOWLEDGEMENTS

This work was partially supported by NSF under the ITR (grants CCR-0121643 and CCR-0219902) and CyberTrust / DAS (grant IIS-0242397) programs, an AFOSR grant, an IBM SUR grant, and Hewlett-Packard.

8. REFERENCES

[1] A. Broder et al. Syntactic clustering of the web. In *Proceedings of the 6th International World Wide Web Conference (WWW '97)*, pages 391–404, 1997.

[2] C. Castillo et al. A reference collection for web spam. *SIGIR Forum*, 40(2):11–24, 2006.

[3] D. Fetterly et al. A large-scale study of the evolution of web pages. In *Proceedings of the 12th International World Wide Web Conference (WWW '03)*, pages 669–678, 2003.

[4] D. Fetterly, M. Manasse, and M. Najork. On the evolution of clusters of near-duplicate web pages. In *Proceedings of the 1st Latin American Web Congress*, pages 37–45, 2003.

[5] D. Fetterly, M. Manasse, and M. Najork. Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages. In *Proceedings of the 7th International Workshop on the Web and Databases (WebDB '04)*, pages 1–6, 2004.

[6] D. Fetterly, M. Manasse, and M. Najork. Detecting phrase-level duplication on the world wide web. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 170–177, 2005.

[7] Z. Gyöngyi and H. Garcia-Molina. Web spam taxonomy. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIRWeb '05)*, 2005.

[8] A. Ntoulas et al. Detecting spam web pages through content analysis. In *Proceedings of the 15th International World Wide Web Conference (WWW '06)*, pages 83–92, 2006.

[9] M. Rabin. Fingerprinting by random polynomials. Technical Report TR-15-81, Center for Research in Computing Technology, Harvard University, 1981.

[10] Y. Wang et al. Strider typo-patrol: Discovery and analysis of systematic typo-squatting. In *Proceedings of the 2nd Workshop on Steps to Reduce Unwanted Traffic on the Internet (SRUTI '06)*, pages 31–36, 2006.

[11] Y. Wang et al. Spam double funnel: Connecting web spammers with advertisers. In *Proceedings of the 16th International World Wide Web Conference (WWW '07)*, 2007.

[12] S. Webb, J. Caverlee, and C. Pu. Introducing the webb spam corpus: Using email spam to identify web spam automatically. In *Proceedings of the 3rd Conference on Email and Anti-Spam (CEAS '06)*, 2006.

[13] B. Wu and B. D. Davison. Cloaking and redirection: A preliminary study. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIRWeb '05)*, 2005.