UNIVERSITY OF CALIFORNIA, SAN DIEGO

A Comprehensive Approach to Undermining Search Result Poisoning

A dissertation submitted in partial satisfaction of the
requirements for the degree of Doctor of Philosophy

in

Computer Science

by

David Yi-Chen Wang

Committee in charge:

Professor Geoffrey M. Voelker, Chair
Professor Stefan Savage, Co-Chair
Professor Gert Lanckriet
Professor Lawrence Saul
Professor Alex Snoeren

2014

The Dissertation of David Yi-Chen Wang is approved and is acceptable in quality and form for publication on microfilm and electronically:

Co-Chair

Chair

University of California, San Diego

2014

DEDICATION

To my friends and family.

EPIGRAPH

Computers are useless. They can only give you answers.

Pablo Picasso

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my advisors Professors Geoffrey Voelker and Stefan Savage for their continuous inspiration, support, and guidance in this long journey through Graduate School. I would certainly not be here without their belief in me and I am forever grateful for this opportunity.

I would also like to thank my co-authors who provided invaluable assistance throughout: Matthew Der, Mohammad Karami, Damon McCoy, and Lawrence Saul. In addition there are many collaborators from CCIED and CESR that I would like to thank: Kirill Levchenko, Andreas Pitsillidis, Neha Chachra, Chris Kanich, Justin Ma, Brandon Enright, Marti Motoyama, Tristan Halvorson, Bhanu Vattikonda, Grant Jordan, Sarah Meiklejohn, Qing Zhang, Lonnie Liu, Louis Dekoven, Brown Farinholt, Edward Sullivan, Mayank Dhiman, Vern Paxon, Nick Weaver, Christian Kreibich, Chris Grier, Kurt Thomas, Paul Pearce, Frank Li, Linda Lee, and Alex Kaprovelos.

Additionally, I would like to acknowledge my office mates of 3144, who have made the many long working days enjoyable: Ryan Huang, Robert Liu, Feng Lu, and Michael Vrable.

I would also like to thank my thesis committee members for their invaluable feedback and support: Gert Lanckriet and Alex Snoeren.

And lastly, I would like to thank all my friends and family who have always supported me.

Chapter 2, 3 in part, are a reprint of the material as it appears in Proceedings of the ACM Conference on Computer and Communications Security 2011. Wang, David; Savage, Stefan; Voelker, Geoffrey M. The dissertation author was the primary investigator and author of this paper.

Chapter 2, 4 in part, are a reprint of the material as it appears in Processings of

the Network and Distributed System Security Symposium 2013. Wang, David; Savage, Stefan; Voelker, Geoffrey M. The dissertation author was the primary investigator and author of this paper.

Chapter 2, 5, in part, have been submitted for publication of the material as it may appear in Proceedings of the ACM Internet Measurement Conference 2012. Wang, David; Der, Matthew; Karami, Mohammad; Saul, Lawrence; McCoy, Damon; Savage, Stefan; Voelker, Geoffrey M. The dissertation author was the primary investigator and author of this paper.

VITA

2005	Bachelor of Arts in Computer Science, University of California, Berkeley
2010	Master of Science in Computer Science, University of California, San Diego
2014	Doctor of Philosophy in Computer Science, University of California, San Diego

PUBLICATIONS

David Y. Wang, Matthew Der, Mohammad Karami, Lawrence Saul, Damon McCoy, Stefan Savage, and Geoffrey M. Voelker. "Search + Seizure: The Effectiveness of Interventions on SEO Campaigns." In *Proceedings of the ACM Internet Measurement Conference, Vancouver, Canada, November 2014.*

Qing Zhang, David Y. Wang, and Geoffrey M. Voelker. "DSpin: Detecting Automatically Spun Content on the Web.". In *Proceedings of the Network and Distributed System Security Symposium, San Diego, CA, February 2014.*

David Y. Wang, Stefan Savage, and Geoffrey M. Voelker. "Juice: A Longitudinal Study of an SEO Campaign." In *Proceedings of the Network and Distributed System Security Symposium, San Diego, CA, February 2013.*

David Y. Wang, Stefan Savage, and Geoffrey M. Voelker. "Cloak and Dagger: Dynamics of Web Search Cloaking." In *Proceedings of the ACM Conference on Computer and Communications Security, Chicago, IL, October 2011.*

Chris Kanich, Neha Chachra, Damon McCoy, Chris Grier, David Y. Wang, Marti Motoyama, Kirill Levchenko, Stefan Savage, and Geoffrey M. Voelker. "No Plan Survives Contact: Experience with Cybercrime Measurement." In *Proceedings of Workshop on Cyber Security Experimentation and Test (CSET), August 2011.*

ABSTRACT OF THE DISSERTATION

A Comprehensive Approach to Undermining Search Result Poisoning

by

David Yi-Chen Wang

Doctor of Philosophy in Computer Science

University of California, San Diego, 2014

Professor Geoffrey M. Voelker, Chair
Professor Stefan Savage, Co-Chair

Black hat search engine optimization (SEO), the practice of manipulating search results, has long been used by attackers to abuse search engines. In one such instance, *search result poisoning*, attackers siphon off large volumes of user traffic from organic search through organized efforts called *SEO campaigns*, and monetize the resulting traffic through scams ranging from sales of illicit goods to malware distributions. Entire *ecosystems* exist, each consisting of multiple campaigns poisoning on behalf of the same type of funding scam (e.g., counterfeit luxury goods).

These campaigns are supported by two low-level mechanisms: *poisoned search*

results (PSRs) and an *SEO botnet*. Disguised as a typical search result, PSRs in reality entrap unsuspecting users and direct them to scams. To prolifically generate PSRs, campaigns use an SEO botnet of compromised sites.

Although interventions designed to disrupt search poisoning exist (e.g., demoting PSRs, seizing domain names), they tend to treat individual symptoms rather than address root causes. Thus, these reactive approaches are expensive and offer marginal benefit, leading to impractical and limited defenses.

In this dissertation, I present a framework to understand and address the root causes of search result poisoning. In support, I analyze search poisoning from three perspectives: PSRs, SEO botnets, and an ecosystem. Additionally, I synthesize insights acquired while examining lower-level mechanisms (PSRs, SEO botnets) into a comprehensive understanding capable of impacting the attacker's high level operation — their SEO campaign.

From the point-of-view of PSRs, I explore modern cloaking to characterize the role of this black hat SEO technique in supporting PSRs. Then, by infiltrating an SEO botnet, I characterize the composition of an SEO botnet and how attackers generate PSRs at large scale. Lastly, I evaluate the effectiveness of current interventions in disrupting SEO campaigns found in the counterfeit luxury goods ecosystem.

In the end, I present a "bottom-up" approach to understanding and addressing the root causes of search result poisoning. Using a framework constructed from my analyses of lower-level mechanisms, I provide a methodology for identifying campaigns and their infrastructure that provides the improved targeting required for more robust, comprehensive, and systematic intervention.

Chapter 1

Introduction

Search engines serve a critical role on the Web: they connect users and content. In short, search engines return a list of the most relevant Web pages (referred to as *organic search results*) in response to a user input keyword query. Although we commonly think of search engines as solely servicing users, in reality they provide an essential service to two separate groups: users and content owners. They empower users to efficiently find information spread across the Web. Meanwhile, from the content owners' perspective, search engines provide the vital stream of potential customers for their businesses in the form of user traffic. In fact, recent reports from multiple large, consumer-facing sites claim over 60% of their visits originate from search engines [39, 49].

Whereas users pay nothing for this service, businesses bear a cost for their stream of customers. Either they pay a fee to search engines to advertise alongside organic search results in a practice called *sponsored search*, or they invest their efforts in cultivating their organic search rankings. Even though paid search requires less effort, businesses typically take advantage of both options due to the enticing volume of traffic from organic search (reportedly $5\times$ the size of paid search [49]).

However, the abstract task of improving a business' search rankings is not straightforward. In response to a user query, search engines use ranking algorithms (e.g., PageRank [45]) to determine the set of most relevant and important pages to return. And since

organic search largely serves ordinary users, search engines take great care to ensure their ranking algorithms are unbiased and free of business interests. Thus, businesses cannot directly inflate their rankings. At most, they can only indirectly affect their rankings by following the search engine's best practices (e.g., creating high quality content) [17].

Despite this complication, search marketing experts developed search engine optimization (SEO) techniques to improve a client's search rankings in response to the high demand for ranking prominently in search results. These techniques range from "white hat" (e.g., keyword-friendly URLs, descriptive meta tags, sitemaps) to "black hat" (e.g., cloaking, keyword stuffing, hidden text) and are classified based upon their intent. White hat SEO generally reflects best practices for facilitating a search engine's crawling and indexing of pages. Black hat SEO, on the other hand, achieves influence through deceiving and manipulating the search engine's ranking algorithm. For example, in the black hat SEO technique known as *cloaking*, attackers return different versions of the same page to different types of visitors (e.g., scams to ordinary users and benign content to search engine crawlers).

Although seemingly innocuous, in actuality, cloaking is an essential ingredient behind a form of large scale search engine abuse commonly referred to as *search result poisoning*. Through organized efforts called an *SEO campaign*, an attacker "poisons" search results to surreptitiously acquire large volumes of user traffic. This traffic is subsequently monetized through an assortment of scams ranging from sales of illicit goods (e.g., pharmacy, software, luxury goods) to malware distribution (e.g., fake anti-virus). Thus, SEO campaigns deliver user traffic to scams, and in return, the scams fund the campaigns in an exchange of services commonly referred to as *abusive advertising*. Entire *ecosystems* exist, where each ecosystem is composed of several SEO campaigns poisoning on behalf of the same class of funding scam (e.g., a counterfeit luxury goods ecosystem consists of SEO campaigns funded through counterfeit sales).

In general, an SEO campaign refers to all high-level activities working towards acquiring user traffic through search poisoning (e.g., targeting counterfeit Louis Vuitton terms), while the specific task of "poisoning" search results is supported by two low-level mechanisms: *poisoned search results* (PSRs) and an *SEO botnet*. Disguised as a typical search result returned in response to a user query, in reality, a PSR is a mechanism that entraps unsuspecting users and directs them to scams through "baiting-and-switching". In essense, a PSR uses cloaking to return enticing, relevant content to search crawlers in hopes of promoting itself into the search results returned to users. Then, when users visit, the PSR again uses cloaking, but this time to send the users to scams. And to generate these PSRs at large scale, an SEO campaign harnesses a collection of thousands of compromised Web sites known as an SEO botnet.

Because SEO campaigns threaten multiple interested parties (e.g., search engines, users, brand holders), each group has developed intervention strategies. While the specific technical details are murkey, these approaches attempt to disrupt search poisoning by preventing attackers from reaching users. For example, search engines respond to PSRs by either demoting their rankings in search results or removing them entirely to prevent exposing them to users. Similarly, luxury brand holders frequently seize the domain names of counterfeit luxury stores that monetize traffic from SEO campaigns to prevent users from shopping at such stores. Although these approaches are well-intentioned, their efficacy remains an open question. Furthermore, by targeting the attacker's low-level resources (PSRs, domain names) that are inexpensive and abundant, the attacker can quickly adapt with remarkably little cost. For instance, in response to the aforementioned defenses, the attacker can simply create more PSRs and register new domains. Essentially, current defenses are reactively treating the individual symptoms, rather than proactively undermining the root causes — SEO campaigns.

Therefore, in this dissertation, I demonstrate that explicitly targeting an attackers

SEO campaign rather than their low-level resources can lead to the more comprehensive and robust interventions necessary to disrupt search result poisoning. In support of this, I analyze search poisoning from three distinct perspectives: poisoned search results, SEO botnets, and an ecosystem comprised of multiple SEO campaigns. Additionally, I demonstrate how to synthesize insights and infrastructure acquired from examining lower-level mechanisms (crawling and detecting PSRs, infiltrating SEO botnets) into a more comprehensive understanding capable of impacting the attacker's high level operation — their SEO campaign.

1.1 Contributions

To understand the phenomenon of search result poisoning, I take a data-driven approach centered around investigation and empirical measurement. Specifically, I study search poisoning from three distinct perspectives ranging from lower-level mechanisms (poisoned search results, SEO botnets) to high-level operations (ecosystem). From the viewpoint of PSRs, I perform a contemporary study of modern cloaking to characterize the role of this black hat SEO technique in supporting PSRs. Then, by infiltrating an SEO botnet, I characterize both the composition of an influential SEO botnet and how attackers use SEO botnets to generate PSRs at large scale. Finally, I perform an ecosystem-level analysis on the counterfeit luxury goods underground market and evaluate the effectiveness of the current interventions in disrupting SEO campaigns.

The contributions of this dissertation are as follows. First, I conduct a contemporary study of modern cloaking found in search results and characterize the role of cloaking in supporting search poisoning. In addition, I describe *Dagger*, a crawler-based system designed to harvest search result data and detect semantic cloaking at near real time. Running Dagger for two classes of search terms (undifferentiated trending keywords and targeted pharmaceutical keywords), I compare and contrast how attackers target

and monetize two different types of user traffic. And in an early look into the temporal dynamics of cloaking (e.g., lifetime of cloaked pages, responsiveness of search engines), I find that the majority of PSRs persist for more than 24 hours despite the presence of defenses, thereby giving attackers a large window to monetize traffic.

Then, I infiltrate an influential SEO botnet called *GR* that was at one time the largest perpetrator of search poisoning for trending search results. By performing an in-depth investigation into the attacker's objectives and operation over nine months, I characterize both the composition of GR (number of compromised sites and their churn) and how attackers use an SEO botnet to generate PSRs at scale. By correlating contemporaneous data, I also quantify GR's effectiveness in poisoning search results and the subsequent response from search engines it engenders. Additionally, I highlight the symbiotic dependency between SEO campaigns and the scams monetizing traffic, which suggests there is potential for impactful intervention at this level.

Lastly, I perform an ecosystem-level analysis on the counterfeit luxury goods underground market, focusing primarily on the principal SEO campaigns profiting from counterfeit sales and the cat-and-mouse relationship between campaigns and intervention efforts. Incorporating insights and infrastructure from my past work, I create a general methodology to evaluate the effectiveness of interventions in disrupting SEO campaigns by examining the campaign's prominence in search results. In support of this, I demonstrate the correlation between a campaign's search results volume with a business' typical success metrics (e.g., user traffic, order volume). When applying this methodology to contemporary anti-counterfeiting efforts (e.g., search result ranking demotion by search engines, domain name seizure by luxury brand holders), I find both defenses lack the necessary comprehensiveness and responsiveness to affect more than a couple of campaigns, leading to an ecosystem that remains rife with abuse.

In the end, I present a "bottom-up" approach to understanding and addressing

the root causes of search result poisoning. In particular, I perform analyses from the perspective of lower-level mechanisms supporting search poisoning (e.g., PSRs and SEO botnets), and I demonstrate how to consolidate the knowledge acquired into a framework for examining and affecting the attacker's high-level operation (e.g., evaluating the relationship between SEO campaigns and intervention efforts). Ultimately this framework serves as the foundation for identifying campaigns and their critical infrastructure that provides improved targeting required for more robust, comprehensive, and systematic intervention.

1.2 Organization

The remainder of this dissertation is organized in the following manner.

Chapter 2 provides the necessary background and related work on the search poisoning phenomenon.

Chapter 3 examines the prevalence of modern cloaking found in search results from three contemporary search engines (Google, Yahoo, Bing). I describe the implementation of Dagger, a crawler-based system that detects semantic cloaking in search results using a series of heuristics to identify differences between pages returned to ordinary users and those returned to search engine crawlers. Running Dagger on two classes of search terms (trending, pharmaceutical), I characterize the role of cloaking in supporting PSRs.

Chapter 4 describes infiltrating and investigating the GR SEO botnet over nine months. I present a characterization of day-to-day operations focusing both on the composition of GR (size and churn) and on how the attacker uses GR to generate PSRs on behalf of their SEO campaign. By correlating contemporaneous data, I quantify GR's effectiveness and response to interventions. Lastly, I provide insight into the attacker's long-term objectives, including their successes and failures.

Chapter 5 describes a methodology to evaluate the effectiveness of interventions against SEO campaigns by using crawled search result and order volume data. Applying this methodology to current interventions within the counterfeit luxury goods ecosystem, I find defenses lack the necessary comprehensiveness and responsiveness to disrupt the SEO campaigns profiting from search poisoning.

Finally, Chapter 6 concludes this dissertation with a discussion of the most compelling results and identifies opportunities for future research in this space.

Chapter 2

Background

This chapter provides essential background information for understanding the intricacies of search result poisoning. We describe the rationale behind and general approach to cloaking, along with a survey of different techniques used for visitor differentiation. Then we walk through a real life example of a search poisoning attack. Lastly, we survey related work related to the general problem of search engine abuse, much of which inspires our own.

2.1 Cloaking

The term "cloaking", as applied to search engines, has an uncertain history, but dates to at least 1999 when it entered the vernacular of the emerging search engine optimization (SEO) market.[1] The growing role of search engines in directing Web traffic created strong incentives to reverse engineer search ranking algorithms and use this knowledge to "optimize" the content of pages being promoted and thus increase their rank in search result listings. However, since the most effective way to influence search rankings frequently required content vastly different from the page being promoted, this encouraged SEO marketers to serve different sets of page content to search engine

[1] For example, in a September 2000 article in *Search Engine Watch*, Danny Sullivan comments that "every major performance-based SEO firm I know of does [use cloaking]" [58].

crawlers than to normal users; hence, cloaking.

For cloaking to work, the attacker must be able to distinguish between user segments based on some identifier visible to a Web server. The choice of identifier used is what distinguishes between cloaking techniques, which include *Repeat Cloaking*, *User Agent Cloaking*, *Referrer Cloaking* (sometimes also called "Click-through Cloaking"), and *IP Cloaking*.

In the case of *Repeat Cloaking*, the Web site stores state on either the client side (using a cookie) or the server side (e.g., tracking client IPs). This mechanism allows the site to determine whether the visitor has previously visited the site, and to use this knowledge in selecting which version of the page to return. Thus first-time visitors are given a glimpse of a scam, in the hopes of making a sale, but subsequent visits are presented with a benign page stymieing reporting and crawlers (who routinely revisit pages).

In contrast, *User Agent Cloaking* uses the `User-Agent` field from the HTTP request header to classify HTTP clients as user browsers or search engine crawlers. User agent cloaking can be used for benign content presentation purposes (e.g., to provide unique content to Safari on an iPad vs. Firefox on Windows), but is routinely exploited by attackers to identify crawlers via the well-known `User-Agent` strings they advertise (e.g., Googlebot).

Referrer Cloaking takes the idea of examining HTTP headers even further by using the `Referer` field to determine which URL visitors clicked through to reach their site. Thus, attackers commonly only deliver a scam page to users that visit their site by first clicking through the search engine that has been targeted (e.g., by verifying that the `Referer` field is http://www.google.com). This technique has also been used, in combination with repeat cloaking and chains of Web site redirections, to create one-time-use URLs advertised in e-mail spam (to stymie security researchers). However, we

restrict our focus to search engine cloaking in this paper.

Finally, one of the simplest mechanisms in use today is *IP Cloaking*, in which an attacker uses the IP address of the requester in determining the identity of the visitor. With an accurate mapping between IP addresses and organizations, an attacker can then easily distinguish all search engine requests and serve them benign content in a manner that is difficult to side-step. Indeed, the only clear way for search engine operators to mechanistically detect such cloaking is through acquiring fresh IP resources—but the signal of "delisting" a search result performing IP cloaking seems to provide a clear path for efficiently mapping such address space even if it is initially unknown [3]. Although challenging to detect in principle since it would nominally require crawling from a Google IP address, in practice a crawler like Dagger, described in Section 3.2, can still detect the use of IP cloaking because attackers still need to expose different versions of a page to different visitors.

2.2 Example of Search Poisoning

Figure 2.1 shows the steps in a typical search poisoning attack, where users are baited into clicking through a poisoned search result and redirected to a scam. In this example, we presuppose that due to exogenous factors there is sudden interest in terms related to volcanoes (e.g., an eruption somewhere). The scam proceeds as follows: (1) The attacker exploits a vulnerability on a Web site and installs an SEO kit (Section 4.2.1), malware that runs on the compromised site that changes it from a legitimate site into a *doorway* under the attacker's control. (2) Next, when a search engine Web crawler requests the page http://⟨doorway⟩/index.html from the doorway, the SEO kit detects the visitor as a crawler by performing IP cloaking, as described previously in Section 2.1, and returns a page related to volcanoes (the area of trending interest) together with cross links to other compromised sites under the attacker's control. (3) The search engine

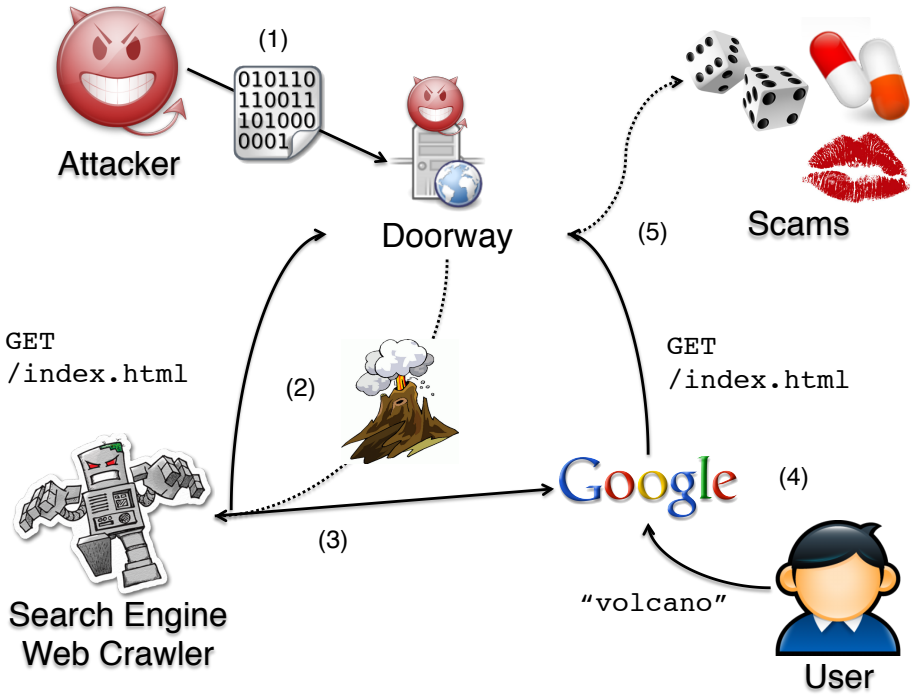


Figure 2.1. A typical search poisoning attack.

indexes this page, and captures its heavy concentration of volcano-related terms and its linkage with other volcano-related sites. (4) Later a user searches for "volcano" and clicks through a now highly ranked search result that links to http://⟨doorway⟩/index.html. (5) Upon receiving this request, the SEO kit on the doorway detects that it is from a user arriving after clicking through a search result, by using a combination of User Agent and Referrer cloaking, and attempts to monetize the click by redirecting the user to a scam such as fake anti-virus [9, 47], which shares little with "volcano".

The example presented above describes a single instance of a quintessential poisoned search result. However, attackers typically orchestrate an SEO botnet, composed of thousands of Web sites operating in unison, to create well over hundreds of thousands of PSRs. Figure 2.2, shows the composition of a typical SEO botnet with sites serving as either doorways or part of a link farm. Depending on the visitor, each doorway returns

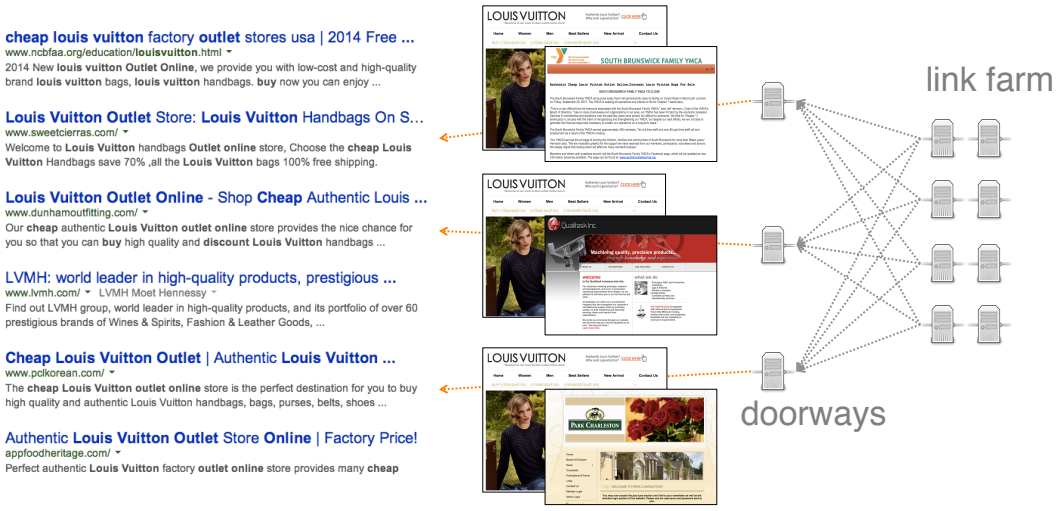


Figure 2.2. A typical SEO botnet composed of doorways and a link farm.

either a scam (e.g., a counterfeit Louis Vuitton storefront) or a benign page. Meanwhile, a link farm creates several backlinks to these doorways in hopes of falsely increasing the doorways' PageRank.

These lower-level mechanisms (PSRs, SEO botnets) support the attacker's high-level organized efforts, otherwise known as their SEO campaign. In Chapter 4 we describe both SEO botnets and SEO campaigns in greater depth.

2.3 Related Work

Related work on search engine abuse tends to be split between well over a decade's worth of work focusing on cloaking (Section 2.3.1), and more recent interest centered around search result poisoning (Section 2.3.2).

2.3.1 Traditional Cloaking Mechanisms

The earliest study of cloaking we are aware of is due to Wu and Davidson [64]. They first developed the now standard technique of crawling pages multiple times (us-

ing both user and crawler identifiers) and comparing the returned content. Using this approach, they refer to situations in which the content differs as “syntactic cloaking”, whereas the subset of these differences that are deemed to be driven by fraudulent intent (using some other classifier [65]) are termed “semantic cloaking”.

Chellapilla and Chickering used a similar detection framework to compare syntactic cloaking on the most popular and monetizable search terms [8]. In this study, monetizability corresponds to the amount of revenue generated from users clicking on sponsored ads returned with search results for a search term. Using logs from a search provider, they found that monetized search terms had a higher prevalence of cloaking (10%) than just popular terms (6%) across the top 5000 search terms.

Up to this point, all studies had focused exclusively on user agent cloaking. In 2006, Wang et al. extended this analysis to include referrer cloaking (called click-through cloaking by them), where pages only return cloaked content if accessed via the URL returned in search results [60]. Targeting a handful of suspicious IP address ranges, they found widespread referrer cloaking among the domains hosted there. In a related, much more extensive study, Wang et al. used this analysis in a focused study of redirection spam, a technique by which “doorway” Web pages redirect traffic to pages controlled by spammers [61]. Finally, Niu et al. performed a similar analysis incorporating referrer cloaking but focused exclusively on forum spamming [43].

These prior studies have used a range of different inputs in deciding whether multiple crawls of the same page are sufficiently different that cloaking has occurred. These include differences in the word distribution in the content of the two pages [8, 64], differences in the links in the page [64], differences in HTML tags [32] or differences in the chain of domain names in the redirection chain [60, 61]. A nice summary of these techniques as well as the different algorithms used to compare them is found in [32].

2.3.2 Phenomenon of Search Poisoning

Recently, cloaking is predominately driven by the phenomenon known as search result poisoning in which the perpetrators seek to acquire traffic by manipulating organic search result rankings. As a result, recent work focuses on various approaches for identifying poisoned search results, including the work from Lu et al. who developed a machine learning approach that proposes important features for statistical modeling and showing their effectiveness on search results for trending terms [34]. During the same time period, Leontiadis et al. [30] and Moore et al. [41] also measured the exposure of poisoned search results to users, and used their measurements to construct an economic model for the financial profitability of this kind of attack.

The work of John et al. is the most similar to portions of the study we have undertaken [24]. Also using an SEO malware kit, they extrapolated key design heuristics for a system, *deSEO*, to identify SEO campaigns using a search engine provider's Web graph. They found that analyzing the historical links between Web sites is important to detecting, and ultimately preventing, SEO campaigns.

Chapter 2, in part, is a reprint of the material as it appears in Proceedings of the ACM Conference on Computer and Communications Security 2011. Wang, David Y.; Savage, Stefan; Voelker, Geoffrey M. The dissertation author was the primary investigator and author of this paper.

Chapter 2, in part, is a reprint of the material as it appears in Proceedings of the Network and Distributed System Security Symposium 2013. Wang, David Y.; Savage, Stefan; Voelker, Geoffrey M. The dissertation author was the primary investigator and author of this paper.

Chapter 2, in part, is a reprint of the material as it appears in Proceedings of the ACM Internet Measurement Conference 2014. Wang, David Y.; Der, Matthew; Karami,

Mohammad; Saul, Lawrence; McCoy, Damon; Savage, Stefan; Voelker, Geoffrey M. The dissertation author was the primary investigator and author of this paper.

Chapter 3

Cloak and Dagger: Dynamics of Web Search Cloaking

In this chapter, we examine search result poisoning from the perspective of poisoned search results, the low-level mechanism responsible for entrapping unsuspecting users and directing them to scams. In particular, we characterize the black hat SEO technique known as cloaking, concentrating on how cloaking allows PSRs to thrive.

3.1 Introduction

The growth of e-commerce in the late 20th century in turn created value around the attention of individual Internet users — described crassly by Caldwell as "The Great Eyeball Race" [5]. Since then, virtually every medium of Internet interaction has been monetized via some form of advertising, including e-mail, Web sites, social networks and on-line games, but perhaps none as successfully as *search*. Today, the top Internet search engines are a primary means for connecting customers and sellers in a broad range of markets, either via "organic" search results or sponsored search placements—together comprising a $14B marketing sector [51].

Not surprisingly, the underlying value opportunities have created strong incentives to influence search results—a field called "search engine optimization" or SEO.

Some of these techniques are benign and even encouraged by search engine operators (e.g., simplifying page content, optimizing load times, etc.) while others are designed specifically to manipulate page ranking algorithms without regard to customer interests (e.g., link farms, keyword stuffing, blog spamming, etc.) Thus, a cat and mouse game has emerged between search engine operators and scammers where search operators try to identify and root out pages deemed to use "black hat" optimization techniques or that host harmful content (e.g., phishing pages, malware sites, etc.) while scammers seek to elude such detection and create new pages faster than they can be removed.

In this conflict, one of the most potent tools is *cloaking*, a "bait-and-switch" technique used to hide the true nature of a Web site by delivering blatantly different content to different user segments. Typically a cloaker will serve "benign" content to search engine crawlers and scam content to normal visitors who are referred via a particular search request. By structuring the benign version of this content to correspond with popular search terms—a practice known as keyword stuffing—Web spammers aggressively acquire unwitting user traffic to their scam pages. Similarly, cloaking may be used to prevent compromised Web servers hosting such scam pages from being identified (i.e., by providing normal content to visitors who are not referred via the targeted search terms). In response to such activity, search engine providers attempt to detect cloaking activity and delist search results that point to such pages.

In this chapter, we study the dynamics of this cloaking phenomenon and the response it engenders. We describe a system called *Dagger*, designed to harvest search result data and identify cloaking in near real-time. Using this infrastructure for over five months we make three primary contributions. First, we provide a contemporary picture of cloaking activity as seen through three popular search engines (Google, Bing and Yahoo) and document differences in how each is targeted. Second, we characterize the differences in cloaking behavior between sites found using undifferentiated "trending"

keywords and those that appear in response to queries for targeted keywords (in particular for pharmaceutical products). Finally, we characterize the *dynamic behavior* of cloaking activity including the lifetime of cloaked pages and the responsiveness of search engines in removing results that point to such sites.

3.2 Methodology

Dagger consists of five functional components: collecting search terms, fetching search results from search engines, crawling the pages linked from the search results, analyzing the pages crawled, and repeating measurements over time. In this section, we describe the design and implementation of each functional component, focusing on the goals and potential limitations.

3.2.1 Collecting Search Terms

The first challenge in data collection is building a meaningful test set for measuring cloaking. Since our goal is to understand the dynamics of scammers utilizing cloaking in search results, we want to target our data collection to the search terms that scammers also target rather than a random subset of the overall search space. In particular, we target two different kinds of cloaked search terms: those reflecting popular terms intended to gather high volumes of undifferentiated traffic, and terms reflecting highly targeted traffic where the cloaked content matches the cloaked search terms.

For our first set of search terms, as with previous work we seed our data collection with popular *trending* search terms. We also enhance this set by adding additional sources from social networks and the SEO community. Specifically, we collect popular search terms from Google Hot Searches, Alexa, and Twitter, which are publicly available and provide real-time updates to search trends at the granularity of an hour.[1] We extract the

[1]We originally considered using an even broader range of search term sources, in particular

top 20 popular search trends via Google Hot Searches and Alexa, which reflect search engine and client-based data collection methods, respectively, while the 10 most popular search terms from Twitter add insight from social networking trends. These sources generally compliment each other and extend our coverage of terms. We found that terms from Google Hot Searches only overlapped 3–8% with trending terms from both Twitter and Alexa. Note that, for trending terms, the page being cloaked is entirely unrelated to the search terms used to SEO the page. A user may search for a celebrity news item and encounter a search result that is a cloaked page selling fake anti-virus.

For our second set of search terms, we use a set of terms catering to a specific domain: pharmaceuticals. We gathered a generic set of pharmaceutical terms common to many spam-advertised online pharmacy sites, together with best-selling product terms from the most popular site [31]. Unlike trending search terms, the content of the cloaked pages actually matches the search terms. A user may search for "viagra" and encounter a cloaked page that leads to an online pharmacy site selling Viagra.

We construct another source of search terms using keyword suggestions from Google Suggest. Google Suggest is a search term autocomplete service that not only speeds up user input, but also allows users to explore similar long-tail queries. For example, when users enter "viagra 50mg", they are prompted with suggestions such as "viagra 50mg cost" and "viagra 50mg canada". Specifically, we submit search terms from Google Hot Searches and the online pharmacy site to Google Suggest and use the result to create dynamic feeds of search terms for trending and pharmaceutical searches, respectively. While investigating SEO community forums, we found various software packages and services using popular search term services as seeds for extending the terms

Yahoo Buzz, Ask IQ, AOL Hot Searches, eBay Pulse, Amazon Most Popular Tags, Flickr Popular Tags, and WordPress Hot Topics. Since we detected no more than a few cloaked results in multiple attempts over time, we concluded that scammers are not targeting these search terms and no longer considered those sources.

they target using suggestion services. Combined with a suggestion service, each search term forms the basis of a cluster of related search terms from lists of suggestions [63]. The main attraction of a suggestion service is that it targets further down the tail of the search term distribution, resulting in less competition for the suggestion and a potentially more advantageous search result position. Moreover, long-tail queries typically retain the same semantic meaning as the original search term seed. Furthermore, recent studies have shown superior conversion rates of long-tail queries [59].

3.2.2 Querying Search Results

Dagger then submits the terms, every four hours for trending queries and every day for pharmaceutical queries, to the three most actively used search engines in the US: Google, Yahoo, and Bing. With results from multiple search engines, we can compare the prevalence of cloaking across engines and examine their response to cloaking. From the search results we extract the URLs for crawling as well as additional metadata such as the search result position and search result snippet.

At each measurement point, we start with the base set of search terms and use them to collect the top three search term suggestions from Google Suggest.[2] For trending searches, Google Hot Searches and Alexa each supply 80 search terms every four-hour period, while Twitter supplies 40. Together with the 240 additional suggestions based on Google Hot Searches, our search term workload is 440 terms. Note that while overlap does occur within each four-hour period and even between periods, this overlap is simply an artifact of the search term sources and represents popularity as intended. For example, a term that spans multiple sources or multiple periods reflects the popularity of the term. For pharmaceutical queries, we always use a set of 230 terms composed of the original

[2]We only use three suggestions to reduce the overall load on the system while still maintaining accuracy, as we found no significant difference in our results when using five or even ten suggestions.

74 manually-collected terms and 156 from Google Suggest.

Next, we submit the search terms and suggestions to the search engines and extract the top 100 search results and accompanying metadata. We assume that 100 search results, representing 10 search result pages, covers the maximum number of results the vast majority of users will encounter for a given query [53]. Using these results Dagger constructs an index of URLs and metadata, which serves as the foundation for the search space that it will crawl and analyze. At this point, we use a whitelist to remove entries in the index based on regular expressions that match URLs from "known good" domains, such as http://www.google.com. This step reduces the number of false positives during data processing. In addition, whenever we update this list, we re-process previous results to futher reduce the number of false positives. Afterwards, we group similar entries together. For example, two entries that share the same URL, source, and search term are combined into a single entry with the same information, except with a count of two instead of one to signify the quantity. As a result, for each search engine, instead of crawling 44,000 URLs for trending search results (440 search terms $\times$ 100 search results), on average Dagger crawls roughly 15,000 unique URLs in each measurement period.

3.2.3 Crawling Search Results

For each search engine, we crawl the URLs from the search results and process the fetched pages to detect cloaking in parallel to minimize any possible time of day effects.

Web crawler. The crawler incorporates a multithreaded Java Web crawler using the `HttpClient 3.x` package from Apache. While this package does not handle JavaScript redirects or other forms of client-side scripting, it does provide many useful features, such as handling HTTP 3xx redirects, enabling HTTP header modification,

timeouts, and error handling with retries. As a result, the crawler can robustly fetch pages using various identities.

Multiple crawls. For each URL we crawl the site three times, although only the first two are required for analysis. We begin disguised as a normal search user visiting the site, clicking through the search result using Internet Explorer on Windows. Then we visit the site disguised as the Googlebot Web crawler. These crawls download the views of the page content typically returned to users and crawlers, respectively. Finally, we visit the site for a third time as a user who does not click through the search result to download the view of the page to the site owner. As with previous approaches, we disguise ourselves by setting the `User-Agent` and the `Referer` fields in the HTTP request header. This approach ensures that we can detect any combination of user-agent cloaking and referrer cloaking. Moreover, our third crawl allows us to detect pure user-agent cloaking without any checks on the referrer. We found that roughly 35% of cloaked search results for a single measurement perform pure user-agent cloaking. For the most part, these sites are not malicious but many are involved in black-hat SEO operations. In contrast, pages that employ both user-agent and referrer cloaking are nearly always malicious (Section 3.3.5).

IP cloaking. Past studies on cloaking have not dealt with IP address cloaking, and the methodology we use is no different. However, because the emphasis of our study is in detecting the situation where cloaking is used as an SEO technique in scams, we do not expect to encounter problems caused by IP cloaking. In our scenario, the cloaker must return the scam page to the user to potentially monetize the visit. And the cloaker must return the SEO-ed page to the search crawler to both index and rank well. Even if the cloaker could detect that we are not a real crawler, they have few choices for the page to return to our imitation crawler. If they return the scam page, they are potentially leaving themselves open to security crawlers or the site owner. If they return the SEO-ed page, then there is no point in identifying the real crawler. And if they return a benign

page, such as the root of the site, then Dagger will still detect the cloaking because the user visit received the scam page, which is noticeably different from the crawler visit. In other words, although Dagger may not obtain the version of the page that the Google crawler sees, Dagger is still able to detect that the page is being cloaked.

To confirm this hypothesis, we took a sample of 20K cloaked URLs returned from querying trending search terms. We then crawled those URLs using the above methodology (three crawls, each with different `User-Agent` and `Referer` fields). In addition, we performed a fourth crawl using Google Translate, which visits a URL using a Google IP address and will fool reverse DNS lookups into believing the visit is originating from Google's IP address space. From this one experiment, we found more than half of current cloaked search results do in fact employ IP cloaking via reverse DNS lookups, yet in every case they were detected by Dagger because of the scenario described above.

3.2.4 Detecting Cloaking

We process the crawled data using multiple iterative passes where we apply various transformations and analyses to compile the information needed to detect cloaking. Each pass uses a comparison-based approach: we apply the same transformations onto the views of the same URL, as seen from the user and the crawler, and directly compare the result of the transformation using a scoring function to quantify the delta between the two views. In the end, we perform thresholding on the result to detect pages that are actively cloaking and annotate them for later analysis.

While some of the comparison-based mechanisms we use to detect cloaking are inspired from previous work, a key constraint is our real-time requirement for repeatedly searching and crawling to uncover the time dynamics of cloaking. As a result, we cannot use a single snapshot of data, and we avoided intensive offline training for machine

learning classifiers [8, 32, 64, 65]. We also avoided running client-side scripts, which would add potentially unbounded latency to our data collection process. Consequently, we do not directly measure all forms of redirection, although we do capture the same end result: a difference in the semantic content of the same URL [60, 61]. Since we continuously remeasure over time, manual inspection is not scalable outside of a couple of snapshots [43]. Moreover, even an insightful mechanism that compares the structure of two views using HTML tags, to limit the effects of dynamic content [32], must be applied cautiously as doing so requires significant processing overhead.

The algorithm begins by removing any entries where either the user or crawler page encountered an error during crawling (a non-200 HTTP status code, connection error, TCP error, etc.); on average, 4% of crawled URLs fall into this category.

At this point, the remaining entries represent the candidate set of pages that the algorithm will analyze for detecting cloaking. To start, the detection algorithm filters out nearly identical pages using text shingling [4], which hashes substrings in each page to construct signatures of the content. The fraction of signatures in the two views is an excellent measure of similarity as we find nearly 90% of crawled URLs are near duplicates between the multiple crawls as a user and as a crawler. From experimentation, we found that a difference of 10% or less in sets of signatures signifies nearly identical content. We remove such pages from the candidate set.

From this reduced set, we make another pass that measures the similarity between the snippet of the search result and the user view of the page. The snippet is an excerpt from the page content obtained by search engines, composed from sections of the page relevant to the original query, that search engines display to the user as part of the search result. In effect, the snippet represents ground truth about the content seen by the crawler. Often users examine the snippet to help determine whether to follow the link in the search result. Therefore, we argue that the user has an implicit expectation that the page content

should resemble the snippet in content.

We evaluate snippet inclusion by first removing noise (character case, punctuation, HTML tags, gratuitous whitespace, etc.) from both the snippet and the body of the user view. Then, we search for each substring from the snippet in the content of the user view page, which can be identified by the character sequence '...' (provided in the snippet to identify non-contiguous text in the crawled page). We then compute a score of the ratio of the number of words from unmatched substrings divided by the total number of words from all substrings. The substring match identifies similar content, while the use of the number of words in the substring quantifies this result. An upper bound score of 1.0 means that no part of the snippet matched, and hence the user view differs from the page content originally seen by the search engine; a lower bound score of 0.0 means that the entire snippet matched, and hence the user view fulfills the expectation of the user. We use a threshold of 0.33, meaning that we filter out entries from the candidate set whose user view does not differ by more than two-thirds from the snippet. We chose this threshold due to the abundance of snippets seen with three distinct substrings, and 0.33 signifies that the majority of the content differs between the snippet and user view. In practice, this step filters 56% of the remaining candidate URLs.

At this point, we know (1) that the views are different in terms of unstructured text, and (2) that the user view does not resemble the snippet content. The possibility still exists, however, that the page is not cloaked. The page could have sufficiently frequent updates (or may rotate between content choices) that the snippet mismatch is misleading. Therefore, as a final test we examine the page structures of the views via their DOMs as in [32]. The key insight for the effectiveness of this approach comes from the fact that, while page content may change frequently, as in blogs, it is far less likely for the page structure to change dramatically.

We compare the page structure by first removing any content that is not part of

a whitelist of HTML structural tags, while also attempting to fix any errors, such as missing closing tags, along the way. We compute another score as the sum of an overall comparison and a hierarchical comparison. In the overall comparison, we calculate the ratio of unmatched tags from the entire page divided by the total number of tags. In the hierarchical comparison, we calculate the ratio of the sum of the unmatched tags from each level of the DOM hierarchy divided by the total number of tags. We use these two metrics to allow the possibility of a slight hierarchy change, while leaving the content fairly similar. An upper bound score of 2.0 means that the DOMs failed to match any tags, whereas a lower bound score of 0.0 means that both the tags and hierarchy matched. We use a threshold of 0.66 in this step, which means that cloaking only occurs when the combination of tags and hierarchy differ by a third between the structure of the user and crawler views. We chose this threshold from experimentation that showed the large majority of cloaked pages scored over 1.0. Once we detect an entry as actively cloaking, we annotate the entry in the index for later processing.

When using any detection algorithm, we must consider the rate of false positives and false negatives as a sign of accuracy and success. Because it is infeasible to manually inspect all results, we provide estimates based on sampling and manual inspection. For Google search results, we found 9.1% (29 of 317) of cloaked pages were false positives, meaning that we labeled the search result as cloaking, but it is benign; for Yahoo, we found 12% (9 of 75) of cloaked pages were false positives. It it worth noting that although we labeled benign content as cloaking, they are technically delivering different data to users and crawlers. If we consider false positives to mean that we labeled the search result as cloaking when it is not, then there are no false positives in either case. In terms of false negatives, when manually browsing collections of search results Dagger detected cloaked redirection pages for the majority of the time. The one case where we fail is when the site employs advanced browser detection to prevent us from fetching the browser view,

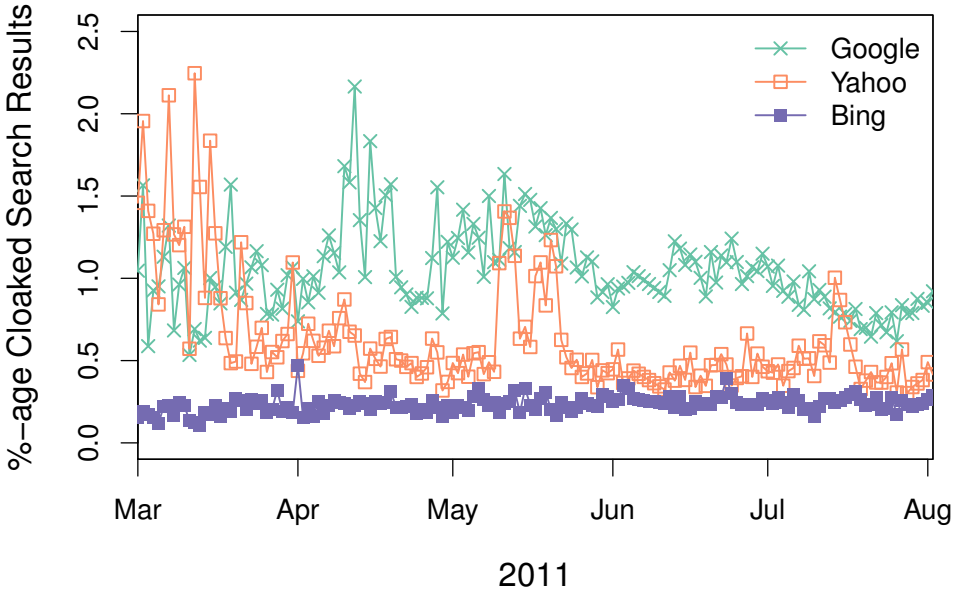
but we have only encountered this case a handful of times.

3.2.5 Temporal Remeasurement

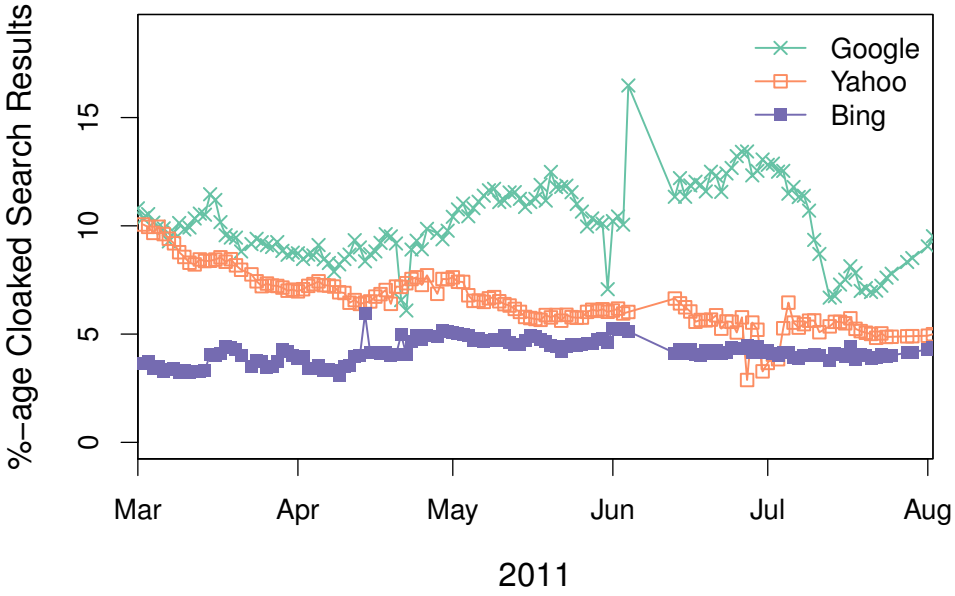
The basic measurement component captures data related to cloaking at one point in time. However, because we want to study temporal questions such as the lifetime of cloaked pages in search results, Dagger implements a temporal component to fetch search results from search engines and crawl and process URLs at later points in time. In the experiments in this chapter, Dagger remeasures every four hours up to 24 hours, and then every day for up to seven days after the original measurement.

The temporal component performs the same basic data collection and processing steps as discussed in the previous components. To measure the rate at which search engines respond to cloaking, we fetch results using the original search term set and construct a new index from the results that will capture any churn in search results since the original measurement. Then we analyze the churn by searching for any entry with a URL that matches the set of cloaked pages originally identified and labeled. Note that there still exists the possibility that for every cloaked page removed from the new index, another cloaked page, which originally was not a part of the index, could have taken its place. Therefore, this process does not measure how clean the search results are at a given time, just whether the original cloaked pages still appear in the search results.

To measure the duration pages are cloaked, the temporal component selects the cloaked URLs from the original index. It then performs the measurement process again, visiting the pages as both a user and a crawler, and applying the detection algorithm to the results. There still exists the possibility that pages perform cloaking at random times rather than continuously, which we might not detect. However, we consider these situations unlikely as spammers need sufficient volume to turn a profit and hence cloaking continuously will result in far greater traffic than cloaking at random.



(a) Trends



(b) Pharmacy

Figure 3.1. Prevalence of cloaked search results in Google, Yahoo, and Bing over time for trending and pharmaceutical searches.

3.3 Results

This section presents our findings from using Dagger to study cloaking in trending and pharmaceutical search results in Google, Yahoo, and Bing. We use Dagger to collect search results every four hours for five months, from March 1, 2011 through August 1, 2011, crawling over 47 million search results. We examine the prevalence of cloaking in search results over time, how cloaking correlates to the various search terms we use, how search engines respond to cloaking, and how quickly cloaked pages are taken down. We also broadly characterize the content of cloaked pages, the DNS domains with cloaked pages, and how well cloaked pages perform from an SEO perspective. Where informative, we note how trending and pharmaceutical cloaking characteristics contrast, and also comment on the results of cloaking that we observe compared with results from previous studies.

3.3.1 Cloaking Over Time

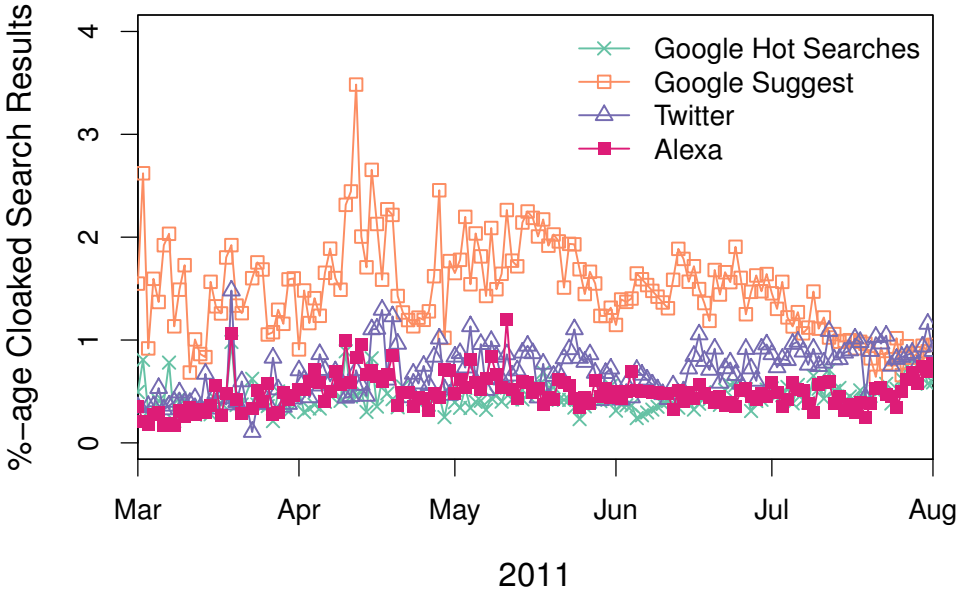
Figure 3.1a shows the prevalence of cloaking over time for trending search results returned from each search engine. We show the percentage of the cloaked search results averaged across all searches made each day. Recall from Section 3.2.3 that we crawl the top 100 search results every four hours for 183 unique trending search terms (on average) collected from Google Hot Searches, Google Suggest, Twitter, and Alexa, resulting on average in 13,947 unique URLs to crawl after de-duping and whitelisting. Although we crawl every four hours, we report the prevalence of cloaking at the granularity of a day for clarity (we did not see any interesting time-of-day effects in the results). For example, when cloaking in Google search results peaked at 2.2% on April 12, 2011, we found 2,094 out of 95,974 cloaked search results that day.

Initially, through May 4th, we see the same trend for the prevalence of cloaked

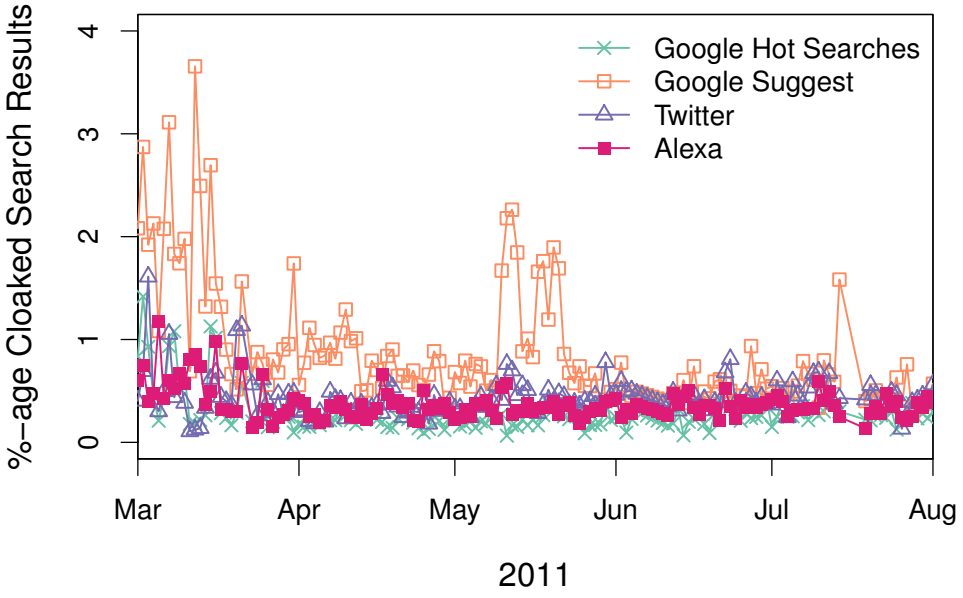
search results among search engines: Google and Yahoo have nearly the same amount of cloaked results (on average 1.11% on Google and 0.84% on Yahoo), whereas Bing has 3–4× fewer (just 0.25%). One explanation is that Bing is much better at detecting and thwarting cloaking, but we find evidence that cloakers simply do not appear to target Bing nearly as heavily as Google and Yahoo. For instance, cloaked results often point to link farms, a form of SEO involving collections of interlinked pages used to boost page rank for sets of search terms. For large-scale link farms that we have tracked over time, we consistently find them in Google and Yahoo results, but not in Bing results.

Similarly, Figure 3.1b shows the prevalence of cloaking over time when searching for pharmaceutical terms. We crawl the top 100 search results daily for 230 unique pharmaceutical search terms collected from a popular spam-advertised affiliate program, further extended with Google Suggest, resuling in 13,646 unique URLs to crawl after de-duping and whitelisting. (Note that the gap in results in the first week of June corresponds to a period when our crawler was offline.) Across all days, we see the same relative ranking of search engines in terms of cloaking prevalence, but with overall larger quantities of cloaked results for the same respective time ranges: on average 9.4% of results were cloaked on Google, 7.7% results on Yahoo, and 4.0% on Bing.

The difference in quantities of cloaked results for trending and pharmaceutical terms reflects the differences between these two types of searches. In trending searches the terms constantly change, with popularity being the one constant. This dynamic allows cloakers to target many more search terms and a broad demographic of potential victims—anyone by definition searching using a popular search term—at the cost of limited time to perform the SEO needed to rank cloaked pages highly in the search results. In contrast, pharmaceutical search terms are static and represent product searches in a very specific domain. Cloakers as a result have much more time to perform SEO to raise the rank of their cloaked pages, resulting in more cloaked pages in the top results. Note,



(a) Google



(b) Yahoo

Figure 3.2. Prevalence of cloaked search results over time associated with each source of trending search terms.

though, that these targeted search terms limit the range of potential victims to just users searching in this narrow product domain. Section 3.3.7 further explores the effects of SEO on cloaked results.

Looking at trends over time, cloakers were initially slightly more successful on Yahoo than Google for trending search terms, for instance. However, from April 1 through May 4th, we found a clear shift in the prevalence of cloaked search results between search engines with an increase in Google (1.2% on average) and a decrease in Yahoo (0.57%). We suspect this is due to cloakers further concentrating their efforts at Google (e.g., we uncovered new link farms performing reverse DNS cloaking for the Google IP range). In addition, we saw substantial fluctuation in cloaking from day to day. We attribute the variation to the adversarial relationship between cloakers and search engines. Cloakers perform blackhat SEO to artifically boost the rankings of their cloaked pages. Search engines refine their defensive techniques to detect cloaking either directly (analyzing pages) or indirectly (updating the ranking algorithm). We interpret our measurements at these time scales as simply observing the struggle between the two sides. Finally, we note that the absolute amount of cloaking we find is less than some previous studies, but such comparisons are difficult to interpet since cloaking results fundamentally depend upon the search terms used.

3.3.2 Sources of Search Terms

Cloakers trying to broadly attract as many visitors as possible target trending popular searches. Since we used a variety of sources for search terms, we can look at how the prevalence of cloaking correlates with search term selection.

Figures 3.2a and 3.2b show the average prevalence of cloaking for each source on search results returned from Google and Yahoo, respectively, for trending searches; we do not present results from Bing due to the overall lack of cloaking. Similar to Figure 3.1,

Table 3.1. Top 10 pharmaceutical search terms with the highest percentage of cloaked search results, sorted in decreasing order.

Search Term	% Cloaked Pages
viagra 50mg canada	61.2%
viagra 25mg online	48.5%
viagra 50mg online	41.8%
cialis 100mg	40.4%
generic cialis 100mg	37.7%
cialis 100mg pills	37.4%
cialis 100mg dosage	36.4%
cialis 10mg price	36.2%
viagra 100mg prices	34.3%
viagra 100mg price walmart	32.7%

each point shows the percentage of cloaked links in the top 100 search results. Here, though, each point shows the average percentage of cloaked results for a particular source, which normalizes the results independent of the number of search terms we crawled from each source. (Because different sources provided different numbers of search terms, the percentages do not sum to the overall percentages in Figure 3.1.)

From the graphs, we see that, through May 4th, using search terms from Google Suggest, seeded initially from Google Hot Searches, uncovers the most cloaking. For Google search results, averaged across the days, Google Suggest returns 3.5× as many cloaked search results as Google Hot Searches alone, 2.6× as Twitter, and 3.1× as Alexa. Similarly, even when using Yahoo, Google Suggest returns 3.1× as many cloaked search results as Google Hot Searches alone, 2.6× as Twitter, and 2.7× as Alexa. As discussed in Section 3.2.1, cloakers targeting popular search terms face stiff SEO competition from others (both legitimate and illegitimate) also targeting those same terms. By augmenting popular search terms with suggestions, cloakers are able to target the same semantic topic as popular search terms. Yet, because the suggestion is essentially an autocomplete, it possesses the long-tail search benefits of reduced competition while

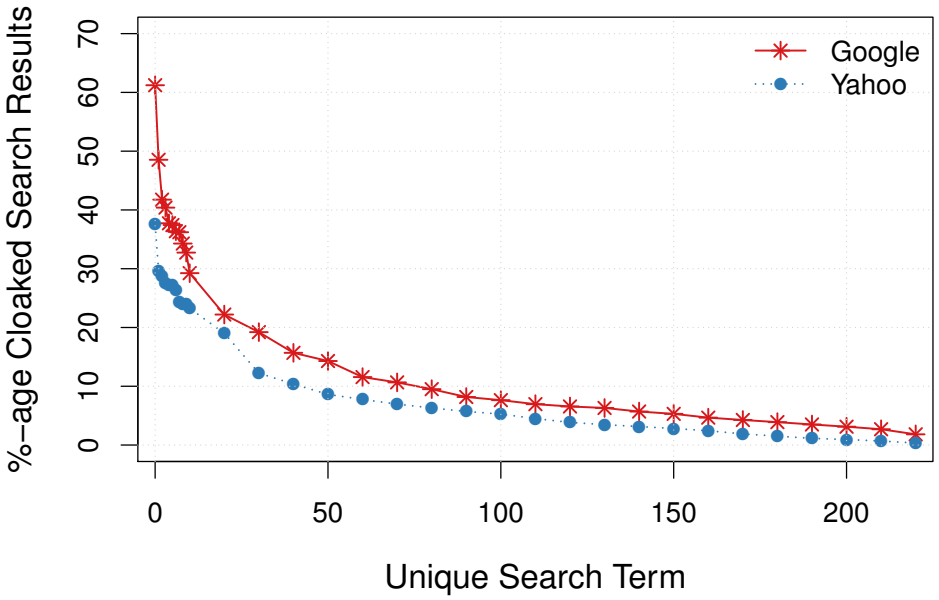
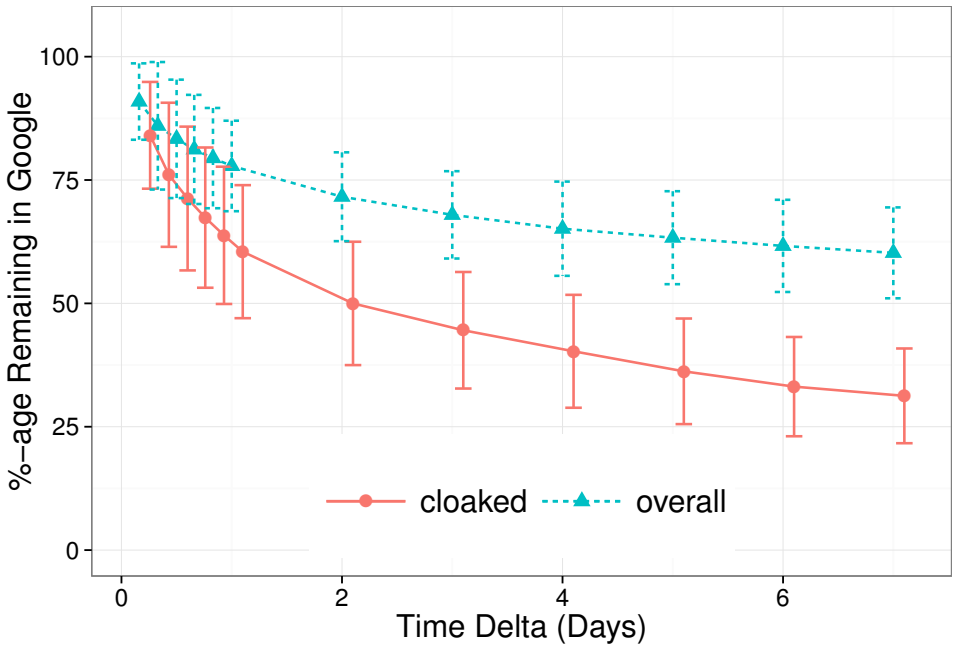


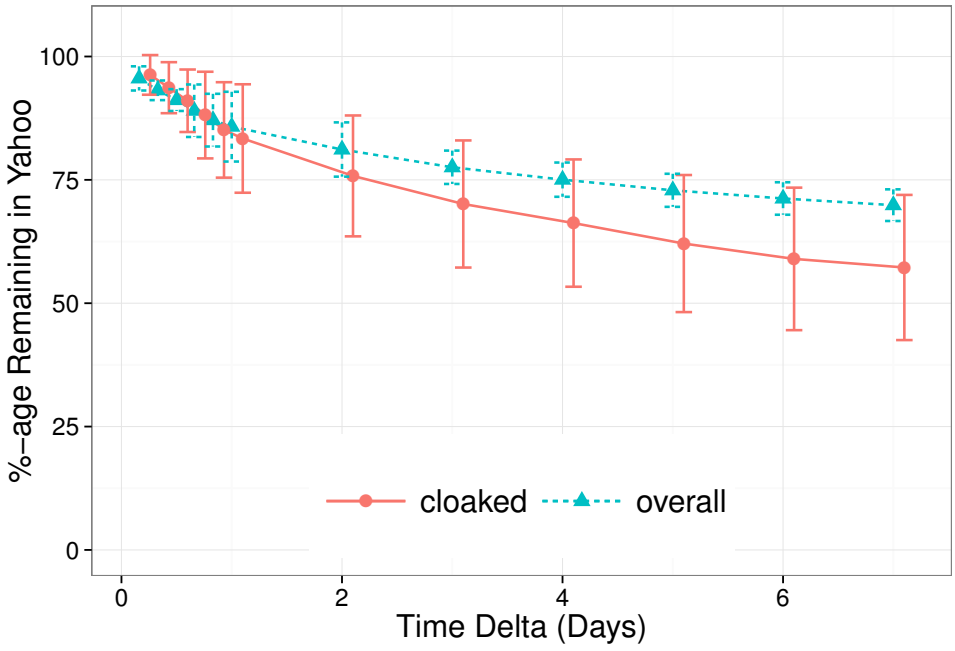
Figure 3.3. Distribution of percentage of cloaked search results for pharmaceutical search terms, sorted in decreasing order.

remaining semantically relevant.

The above results demonstrate that the prevalence of cloaking in search results is highly influenced by the search terms. As another perspective, for each measurement period that crawls the search terms at a given point in time, we can count the number of cloaked results returned for each search term. Averaging across all measurement periods, 23% and 14% of the search terms accounted for 80% of the cloaked results from Google and Yahoo, respectively. For reference, Table 3.1 lists the results for the top 10 search terms on Google and Figure 3.3 shows the distribution of the percentage of cloaked search results for pharmaceutical search terms. The query "viagra 50mg canada" is the pharmaceutical term with the largest percentage of cloaked search results on Google with 61%. Yet, the median query "tramadol 50mg" contains only 7% of cloaked search results. Note that the percentages sum to much more than 100% since different search

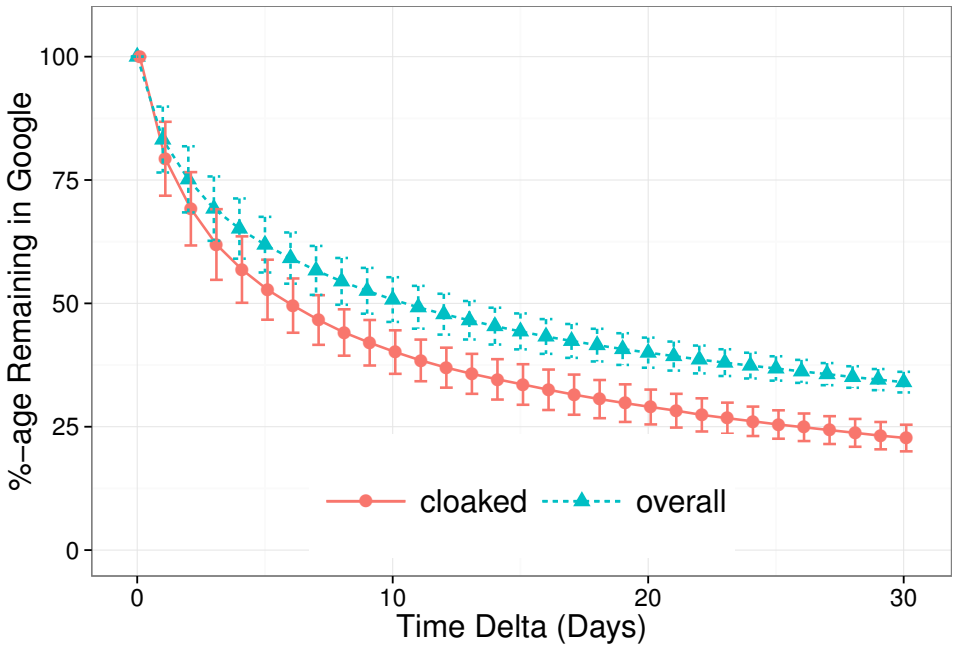


(a) Google

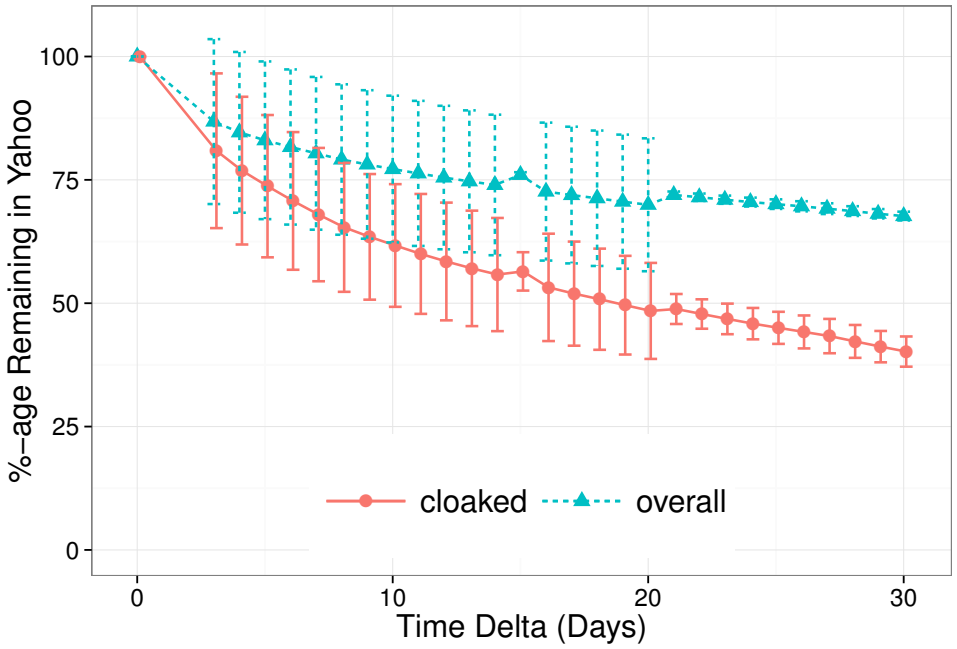


(b) Yahoo

Figure 3.4. Churn in the top 100 cloaked search results and overall search results from Google and Yahoo for trending search terms.



(a) Google



(b) Yahoo

Figure 3.5. Churn in the top 100 cloaked search results and overall search results from Google and Yahoo for pharmaceutical search terms.

terms can return links to the same cloaked pages. As an example in Figure 3.3, the sixth point shows the average percentage of cloaked search results, across all measurements, for the search term with the sixth highest percentage of cloaked search results. We plot the first 10 points with the most significant percentage of cloaked search results, then plot every 10th search term, for clarity. From these results we see high variance in the percentage of cloaked search results.

3.3.3 Search Engine Response

Next we examine how long cloaked pages remain in search results after they first appear. For a variety of reasons, search engines try to identify and thwart cloaking. Although we have little insight into the techniques used by search engines to identify cloaking,[3] we can still observe the external effects of such techniques in practice.

We consider cloaked search results to have been effectively "cleaned" by search engines when the cloaked search result no longer appears in the top 100 results. Of course, this indicator may not be directly due to the page having been cloaked. The search engine ranking algorithms could have adjusted the positions of cloaked pages over time due to other factors, e.g., the SEO techniques used by cloakers may turn out to be useful only in the short term. Either way, in this case we consider the cloaked pages as no longer being effective at meeting the goals of the cloakers.

To measure the lifetime of cloaked search results, we perform repeated search queries for every search term over time (Section 3.2.5). We then examine each new set of search results to look for the cloaked results we originally detected. The later search results will contain any updates, including removals and demotions, that search engines have made since the time of the initial measurement. To establish a baseline we also measure the lifetime of all our original search results, cloaked or not. This baseline allows

[3] Google has a patent in the area [42], but we have not seen evidence of such a client-assisted approach used in practice.

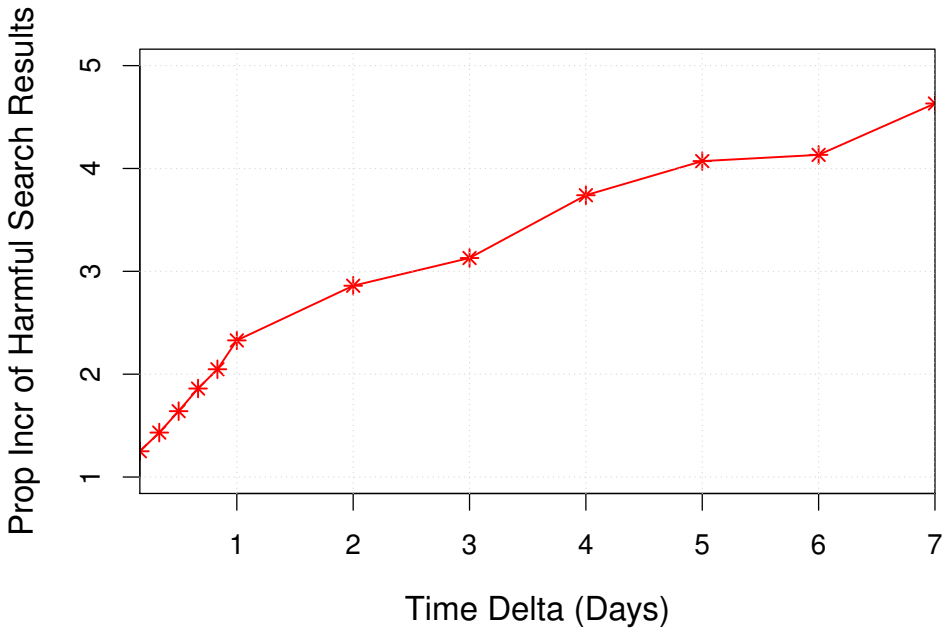


Figure 3.6. Proportional increase in harmful trending search results over time on Google as labeled by Google Safe Browsing.

us to differentiate any churn that occurs naturally with those attributable to "cleaning". We perform these repeated searches on each term every four hours up to 24 hours and then every day up to seven days.

Figures 3.4a and 3.4b show the lifetime of cloaked and overall search results for Google and Yahoo for trending searches. Each point shows the average percentage of search results that remain in the top 100 for the same search terms over time. The error bars denote the standard devation across all searches, and we plot the points for "cloaked" slightly off-center to better distinguish error bars on different curves. The results, for both search engines, show that cloaked search results rapidly begin to fall out of the top 100 within the first day, with a more gradual drop thereafter. In contrast, search results in general have similar trends, but decline more gradually. For Google, nearly 40% of cloaked search results have a lifetime of a day or less, and over the next six days

only an additional 25% drop from the top 100 results. In contrast, for the baseline only 20% of overall search results have a lifetime of a day of less, and an additional 15% are cleaned after the next six days. Yahoo exhibits a similar trend, although with less rapid churn and with a smaller separation between cloaked and the baseline (perhaps reflecting differences in how the two search engines deal with cloaking). Overall, though, while cloaked pages do regularly appear in search results, many are removed or demoted by the search engines within hours to a day.

Figures 3.5a and 3.5b show similar results for pharmaceutical searches. Note that the maximum time delta is 30 days because, unlike trending terms, the pharmacy search terms do not change throughout the duration of our experiment and we have a larger window of observation. While we still see similar trends, where cloaked search results drop more rapidly than the churn rate and Google churns more than Yahoo, the response for both Google and Yahoo is slower for pharmaceutical terms than for trending terms. For example, whereas 45% and 25% of cloaked trending search results were "cleaned" for Google and Yahoo, respectively, within two days, only 30% and 10% of cloaked pharmacy search results were "cleaned" for Google and Yahoo, respectively.

As another perspective on "cleaning", Google Safe Browsing [16] is a mechanism for shielding users by labeling search results that lead to phishing and malware pages as "harmful". These harmful search results sometimes employ cloaking, which Google Safe Browsing is able to detect and bypass. This insight suggests that the rate that Google is able to discover and label "harmful" search results correlates with the rate at which they can detect cloaking. We can measure this Safe Browsing detection by repeatedly querying for the same terms as described in Section 3.2.5 and counting the number of "harmful" search results.

As observed in Section 3.3.2, the prevalence of cloaking is volatile and depends heavily on the specific search terms. The prevalence of detected harmful pages is similarly

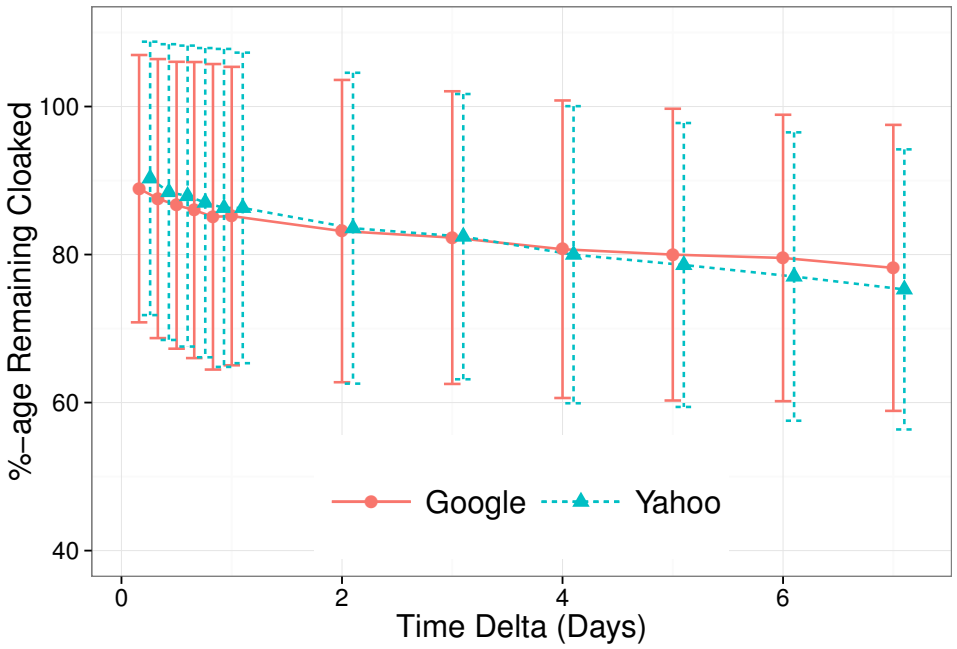


Figure 3.7. Duration pages are cloaked.

volatile; although 37% of the results on average on Google are marked as harmful for the terms we search for, there is substantial variance across terms. Therefore, we normalize the change over time in the number of harmful search results labeled by Google Safe Browsing relative to the first measurement. Figure 3.6 shows the average normalized change in the number of harmful labels over time, across all queries on trending search terms. The number of harmful labels increases rapidly for the first day, with nearly 2.5× more labels than the original measurement, and then increases steadily over the remaining six days, where there are nearly 5× more labels than the original query. This behavior mirrors the results on cleaning above.

3.3.4 Cloaking Duration

Cloakers will often subvert existing pages as an SEO technique to capitalize on the already established good reputation of those pages with search engines. We have

seen that the search engines respond relatively quickly to having cloaked pages in search results. Next we examine how long until cloaked pages are no longer cloaked, either because cloakers decided to stop cloaking or because a subverted cloaked page was discovered and fixed by the original owner.

To measure the duration that pages are cloaked, we repeatedly crawl every cloaked page that we find over time, independent of whether the page continues to appear in the top 100 search results. We then apply the cloaking detection algorithm on the page (Section 3.2.4), and record when it is no longer cloaked. As in Section 3.3.3, we crawl each page every four hours up to 24 hours and then every day up to seven days.

Figure 3.7 shows the time durations that pages are cloaked in results returned by Google and Yahoo. Each point shows the percentage of all cloaked pages for each measurement period that remain cloaked over time, and the error bars show the standard deviations across measurement periods. We see that cloaked pages have similar durations for both search engines: cloakers manage their pages similarly independent of the search engine. Further, pages are cloaked for long durations: over 80% remain cloaked past seven days. This result is not very surprising given that cloakers have little incentive to stop cloaking a page. Cloakers will want to maximize the time that they might benefit from having a page cloaked by attracting customers to scam sites, or victims to malware sites. Further, it is difficult for them to recycle a cloaked page to reuse at a later time. Being blacklisted by Google Safe Browsing, for instance, requires manual intervention to regain a positive reputation. And for those cloaked pages that were subverted, by definition it is difficult for the original page owners to detect that their page has been subverted. Only if the original page owners access their page as a search result link will they realize that their page has been subverted; accessing it any other way will return the original contents that they expect.

Table 3.2. Breakdown of cloaked content for manually-inspected cloaked search results from Google for trending search terms. Note that "Traffic Sale" pages are the start of redirection chains that typically lead to Fake-AV, CPALead, and PPC landing pages.

Category	% Cloaked Pages
Traffic Sale	81.5%
Error	7.3%
Legitimate	3.5%
Software	2.2%
SEO-ed Business	2.0%
PPC	1.3%
Fake-AV	1.2%
CPALead	0.6%
Insurance	0.3%
Link farm	0.1%

3.3.5 Cloaked Content

Since the main goal of cloaking as an SEO technique is to obtain user traffic, it is natural to wonder where the traffic is heading. By looking at the kind of content delivered to the user from cloaked search results, not only does it suggest why cloaking is necessary for hiding such content, but it also reveals the motives cloakers have in attracting users.

We have no fully automated means for identifying the content behind cloaked search results. Instead, we cluster cloaked search results with the exact same DOM structure of the pages as seen by the user when clicking on a search result. We perform the clustering on all cloaked search results from Google across all measurement points for trending searches. To form a representative set of cloaked pages for each cluster, we select a handful of search results from various measurement times (weekday, weekend, dayime, morning, etc.) and with various URL characteristics. We then manually label pages in this representative set to classify the content of the pages being cloaked.

We manually label the content of each cluster into one of ten categories: traffic

sales, pay-per-click (PPC), software, insurance, Fake-AV, CPALead,[4] link farm, SEO-ed business, error, and legitimate. Traffic sales are cloaked search results with the sole purpose of redirecting users through a chain of advertising networks, mainly using JavaScript, before arriving at a final landing page. Although we are unable to follow them systematically, from manually examining thousands of traffic sales, we observed these search results directing users primarily to Fake-AV, CPALead, and PPC pages. Occasionally cloaked search results do not funnel users through a redirection chain, which is how we are able to classify the PPC, software, insurance, Fake-AV, and CPALead sets. The link farm set contains benign pages that provide many backlinks to boost the rankings of beneficiary pages. The SEO-ed business refers to businesses that employ black-hat SEO techniques, such as utilizing free hosting to spam a large set of search results for a single term. The errors are pages that have client side requirements we were unable to meet, i.e., having an Adobe Flash plugin. Finally, the legitimate set refers to pages that display no malicious behavior but were labeled as cloaking due to delivering differing content, as is the case when sites require users to login before accessing the content.

Table 3.2 shows the breakdown of cloaked search results after manually inspecting the top 62 clusters, out of 7671 total, which were sorted in decreasing order of cluster size. These 62 clusters account for 61% of all cloaked search results found in Google for trending searches across all measurement points. From this, we see that about half of the time a cloaked search result leads to some form of abuse. Further, over 49% of the time, cloaked search results sell user traffic through advertising networks, which will eventually lead to Fake-AV, CPALeads, or PPC.

Interestingly, the DOM structure of the largest cluster, which alone acounted for

[4]Cost-per-action pages that ask a user to take some action, such as filling out a form, that will generate advertising revenue.

34% of cloaked search results, was a single JavaScript snippet that performs redirection as part of traffic sales. Other large clusters that accounted for 1–3% of cloaked search results also consist primarily of JavaScript that performs redirection as part of traffic sales.

Despite the fact that the clusters we have examined account for 61% of all cloaked search results, there still exists 39% that have not been categorized and likely do not share the same distribution. While incrementally clustering, we noted that the top clusters grew larger and larger as more and more data was added. This suggests the presence of long-term SEO campaigns, as represented by the top clusters, that constantly change the search terms they are targeting and the hosts they are using. Therefore, since the uncategorized search results fall within the long tail, they are unlikely to be actively involved in direct traffic sales. Instead, we speculate that they fall in the link farm or legitimate sets given that those groups have the most difficult time in forming large clusters because they are not being SEO-ed as heavily across search terms.

The kinds of pages being cloaked is also dynamic over time. Figure 3.8 shows the classification of cloaked page content for search results from Google using trending terms, from March 1, 2011 through July 15, 2011. We classify the HTML of cloaked pages, using the file size and substrings as features, into one of the following categories: Linkfarm, Redirect, Error, Weak, or Misc. The Linkfarm category represents content returned to our "Googlebot" crawler that contains many outbound links hidden using CSS properties. The Redirect category represents content returned to a user that is smaller than 4 KB and contains JavaScript code for redirection, or an HTML meta refresh. The Error category represents user content that is smaller than 4 KB and contains a blank page or typical error messages. The Weak category contains user content below 4 KB in file size not already classified; similarly, the Misc category contains user content larger than 4 KB not already classified. As an example, on March 25th approximately 2% of

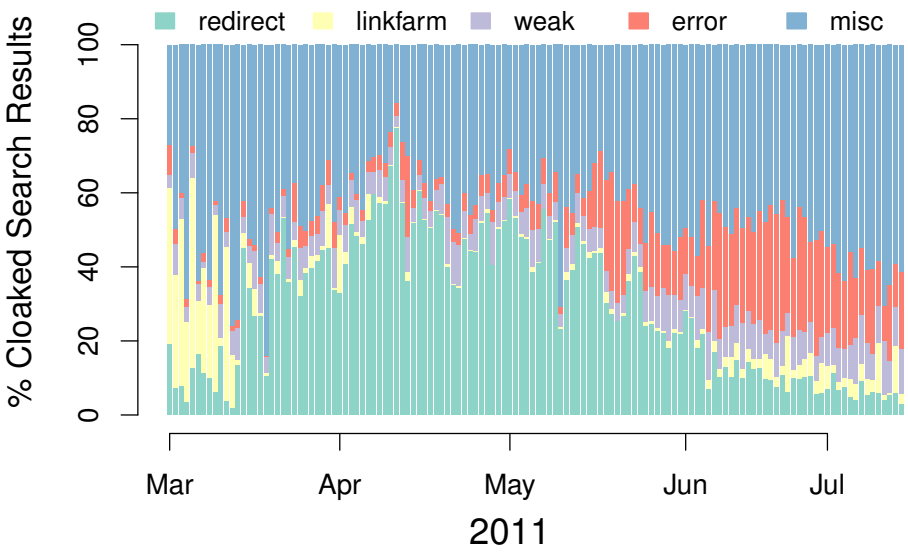


Figure 3.8. Proportional distribution of cloaked search results in Google over time for trending searches.

the cloaked content detected were linkfarms, 46% were redirects, 10% were errors, 4% were weak and 38% were misc.

Looking at the trends over time reveals the dynamic nature of the content being hidden by cloaking. In particular, we saw a surge in redirects from March 15th to June 5th. During this period, the average distribution of redirects per day increased from 11.4% to 41.3% and later dropped off to 8.7%. Interestingly, as redirects begin to fall off, we see a corresponding increase in errors. During the high period of redirects, errors represented 8.0% of the average distribution, but afterwards represented 24.3%. One explanation of this correlation is that the infrastructure supporting redirects begins to collapse at this point. Anecdotally, this behavior matches the general time frame of the undermining of key Fake-AV affiliate programs [28], frequently observed at the end of the redirect chains.

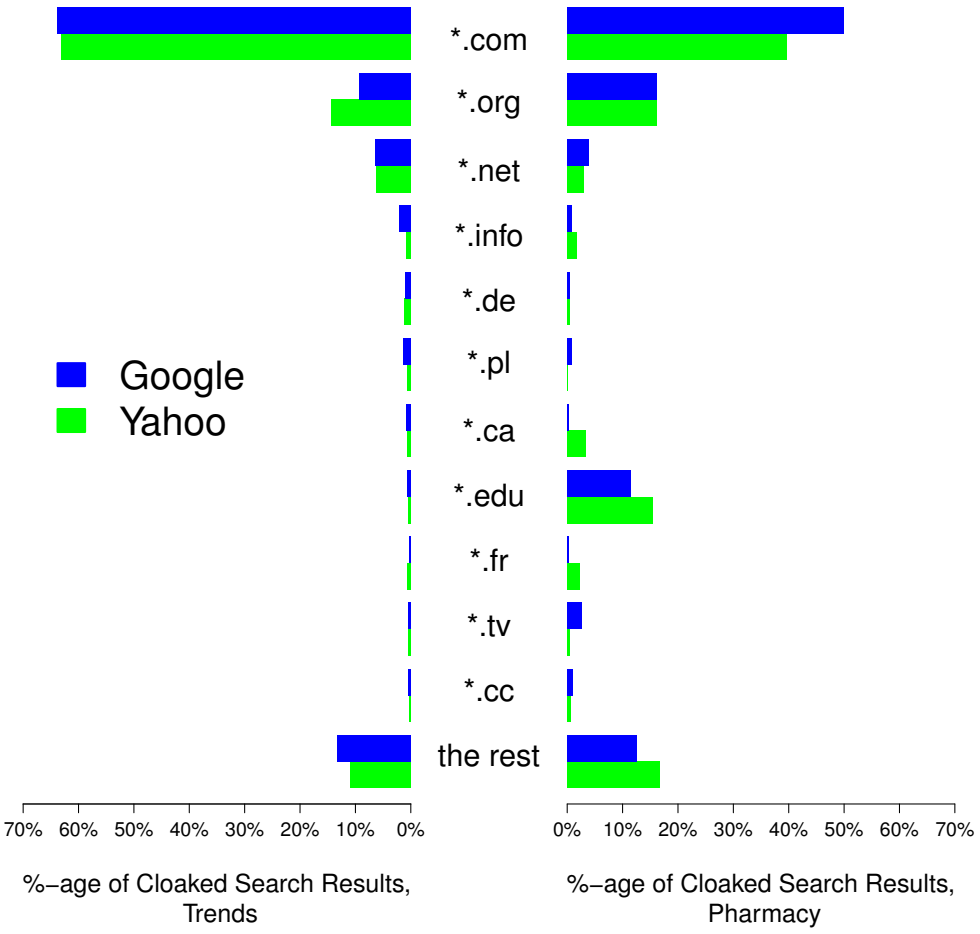


Figure 3.9. Histogram of the most frequently occuring TLDs among cloaked search results.

3.3.6 Domain Infrastructure

Analyzing the underlying intent of cloaked pages confirmed again that cloakers are attempting to attract traffic by illegitimately occupying top search result positions for trending and pharmacy search terms and their suggestions. The implication is that the key resource that spammers must possess, to effectively utilize cloaking in their scams, is access to Web sites and their domains. Ideally, these sites should be established sites already indexed in search engines. Otherwise, solely using traditional SEO tactics,

such as link farming, will have limited success in obtaining high search result positions. Recent reports confirm that many pages have been targeted and infected by well known exploits to their software platforms, and subsequently used to cloak hidden content from search engines [55].

In this section, we examine the top level domains (TLDs) of cloaked search results. Figure 3.9 shows histograms of the most frequently occuring TLDs among all cloaked search results, for both Google and Yahoo. We see that the majority of cloaked search results are in `.com`. Interestingly, cloaked search results from pharmaceutical queries utilize domains in `.edu` and `.org` much more frequently, where together they represent 27.6% of all cloaked search results seen in Google and 31.7% in Yahoo. For comparison, `.edu` and `.org` together represent just 10% in Google and 14.8% in Yahoo for trending searches. Cloakers spamming pharmaceutical search terms are using the "reputation" of pages in these domains to boost their ranking in search results similar to the accusations against `overstock.com` [11].

3.3.7 SEO

Finally, we explore cloaking from an SEO perspective by quantifying how successful cloaking is in high-level spam campaigns. Since a major motivation for cloaking is to attract user traffic, we can extrapolate SEO performance based on the search result positions the cloaked pages occupy. For example, a campaign that is able to occupy search result positions between 1–20 is presumably much more successful than one that is only able to occupy search result positions between 41–60.

To visualize this information, we calculate the percentage of cloaked search results found between ranges of search result positions for each measurement. Then we take the average across all measurements. Again, we only focus on Google and Yahoo due to the lack of cloaked search results in Bing. For clarity, we bin the histogram by

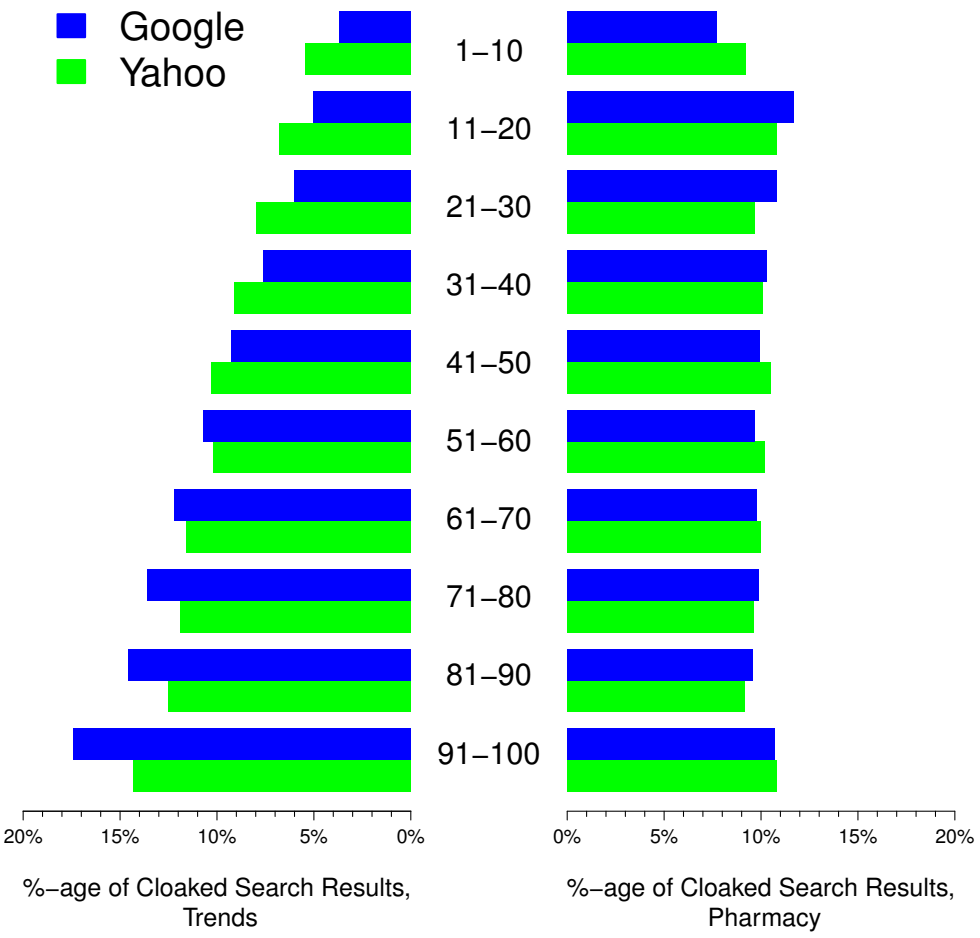


Figure 3.10. Distribution of cloaked search result positions.

grouping every ten positions together, the default number of search results per page.

Figure 3.10 shows the resulting histograms for trending search terms and pharmaceutical terms, side by side. For trending searches we see a skewed distribution where cloaked search results mainly hold the bottom positions; for both Google and Yahoo, positions further away from the top contain more cloaking. Compared to the positions 1–10 on the first page of results, the number of cloaked search results are 2.1× more likley to hold a search result positon between 31–40 in Google, and 4.7× more likely to be in position 91–100; results for Yahoo are similar. In some ways, this distribution

indicates that cloaking is not very effective for trending search terms. It does not lead to a higher concentration in the most desirable search result positions (top 20), likely due to the limited amount of time available to SEO. Although cloakers will have fewer opportunities for their scams as a result, presumably it still remains profitable for cloakers to continue the practice.

Interestingly, we see a very different trend in pharmaceutical searches where there is an even distribution across positions. The number of cloaked pages are just as likely to rank in the first group of search results (positions 1–10) as any other group within the top 100. Wu and Davison [64] had similar findings from 2005. One possible explanation is that the differences again reflect the differences in the nature of cloaked search terms. Cloaking the trending terms by definition target popular terms that are very dynamic, with limited time and heavy competition for performing SEO on those search terms. Cloaking pharmacy terms, however, is a highly focused task on a static set of terms, providing much longer time frames for performing SEO on cloaked pages for those terms. As a result, cloakers have more time to SEO pages that subsequently span the full range of search result positions.

3.4 Summary

Cloaking has become a standard tool in the scammer's toolbox and one that adds significant complexity for differentiating legitimate Web content from fraudulent pages. Our work has examined the current state of search engine cloaking as used to support Web spam, identified new techniques for identifying it (via the search engine snippets that identify keyword-related content found at the time of crawling) and, most importantly, we have explored the dynamics of cloaked search results and sites over time. We demonstrate that the majority of cloaked search results remain high in rankings for 12 hours and that the pages themselves can persist far longer. Thus, cloaking is likely to be

an effective mechanism so long as the overhead of site placement via SEO techniques is less than the revenue obtained from 12 hours of traffic for popular keywords. We believe it is likely that this holds, and search engine providers will need to further reduce the lifetime of cloaked results to demonetize the underlying scam activity.

Chapter 3, in part, is a reprint of the material as it appears in Proceedings of the ACM Conference on Computer and Communications Security 2011. Wang, David Y.; Savage, Stefan; Voelker, Geoffrey M. The dissertation author was the primary investigator and author of this paper.

Chapter 4

Juice: A Longitudinal Study of an SEO Campaign

In this chapter, we change the perspective of our search result poisoning analysis to SEO botnets, the low-level mechanism responsible for spawning poisoned search results at large-scale. In particular, we infiltrate an influential SEO botnet called GR and study it from within, focusing on characterizing both the composition of the botnet (size, churn, etc.) and the machinery of how attackers use botnets to generate PSRs.

4.1 Introduction

Traffic is the lifeblood of online commerce: eyeballs equal money in the crass parlance of today's marketers. While there is a broad array of vectors for attracting user visits, Web search is perhaps the most popular of these and is responsible for between 10 and 15 billion dollars in annual advertising revenue [12, 15].

However, in addition to the traffic garnered by such *sponsored* search advertising, even more is driven by so-called "organic" search results. Moreover, it is widely held that the more highly ranked pages—those appearing at the beginning of search results—attract disproportionately greater volumes of visitors (and hence potential revenue). Thus, a large ecosystem has emerged to support *search engine optimization* or SEO—the practice

of influencing a site's ranking when searching under specific query terms. Many of these practices are explicitly encouraged by search engines with the goal of improving the overall search experience (e.g., shorter load times, descriptive titles and metadata, effective use of CSS to separate content from presentation, etc.) and such approaches are commonly called "white hat" SEO techniques. However, on the other side of the spectrum are "black hat" techniques that explicitly seek to manipulate the search engine's algorithms with little interest in improving some objective notion of search quality (e.g., link farms, keyword stuffing, cloaking and so on).

Unsurprisingly, such black hat techniques have quickly been pressed into the service of *abusive advertising*—advertising focused on attracting traffic for compromise (e.g., drive-by downloads [20]), for fraud (e.g., fake anti-virus [56]), or for selling counterfeit goods (e.g., pharmaceuticals or software).[1] While a few such incidents would not generate alarm, there is increasingly clear evidence of large-scale SEO campaigns being carried out: large numbers of compromised Web sites harnessed in unison to poison search results for attractive search queries (e.g., trending search terms). Indeed, one recent industry report claims that 40% of all malware infestations originate in poisoned search results [29]. However, the details of how such *search poisoning attacks* are mounted, their efficacy, their dynamics over time and their ability to manage search engine countermeasures are still somewhat opaque.

In service to these questions, this chapter examines *in depth* the behavior of one influential search poisoning botnet, "GR".[2] In particular, we believe our work offers three primary contributions in this vein.

Botnet characterization. By obtaining and reverse engineering a copy of the

[1] Indeed, in one recent study of counterfeit online pharmaceuticals the most successful advertiser was not an e-mail spammer, but rather was an SEO specialist [38].

[2] Each of the functions and global variables in this botnet are prefixes with a capital GR. We believe it is an acronym, but at the time of this writing we do not know what the authors intended it to stand for.

"SEO kit" malware installed on compromised Web sites, we were able to identify other botnet members and infiltrate the command and control channel. Using this approach we characterize the activities of this botnet and its compromised hosts for nine months. We show that unlike e-mail spamming botnets, this SEO botnet is modest in size (under a thousand compromised Web sites) and has a low rate of churn (with individual sites remaining in the botnet for months). Moreover, we document how the botnet code is updated over time to reflect new market opportunities.

Poisoning dynamics. By correlating captured information about the keywords being promoted with contemporaneous Internet searches, we are able to establish the effectiveness of such search poisoning campaigns. Surprisingly, we find that even this modest sized botnet is able to effectively "juice" the ranking of thousands of specific search terms within 24 hours and, in fact, it appears to have been the dominant contributor to poisoned trending search results at Google during its peak between April and June 2011.

Targeting. By systematically following and visiting the "doorway" pages being promoted, both through redirections and under a variety of advertised browser environments, we are able to determine the ultimate scams being used to monetize the poisoning activity. We find evidence of a "killer scam" for search poisoning and document high levels of activity while the fake antivirus ecosystem is stable (presumably due to the unusually high revenue generation of such scams [56]). However, after this market experienced a large setback, the botnet operator explores a range of lower-revenue alternatives (e.g., pay-per-click, drive-by downloads) but never with the same level of activity.

Finally, in addition to these empirical contributions, this chapter also documents a methodology and measurement approach for performing such studies in the future. Unlike e-mail spam which delivers its content on a broad basis, search poisoning involves many more moving parts including the choice of search terms and the behavior of the

search engine itself. Indeed, our analyses required data from three different crawlers to gather the necessary information: (1) a host crawler for identifying and monitoring compromised Web sites, (2) a search crawler to identify poisoned search results and hence measure the effectiveness of the poisoning, and (3) a redirection crawler that follows redirection chains from doorway pages linked from poisoned search results to identify the final landing pages being advertised.

4.2 The GR Botnet

In this section we present the architecture of the GR botnet responsible for poisoning search results and funneling users, as traffic, to various scams. We start by introducing its SEO malware kit, and then present a high-level overview of its architecture, highlighting specific functionality found in the SEO kit and the evolution of the source code.

4.2.1 SEO Kit

An SEO kit is software that runs on each compromised Web site that gives the botmaster backdoor access to the site and implements the mechanisms for black hat search engine optimization. We obtained an SEO kit after contacting numerous owners of compromised sites. After roughly 40 separate attempts, one site owner was willing and able to send us the injected code found on their site. Although we cannot pinpoint the original exploit vector on the compromised Web site, there have been many recent reports of attackers compromising Web sites by exploiting Wordpress and other similar open source content management systems [36].

The SEO kit is implemented in PHP and consists of two components, the *loader* and the *driver*. The loader is initially installed by prepending PHP files with an `eval` statement that decrypts base64 encoded code. When the first visitor requests the modified

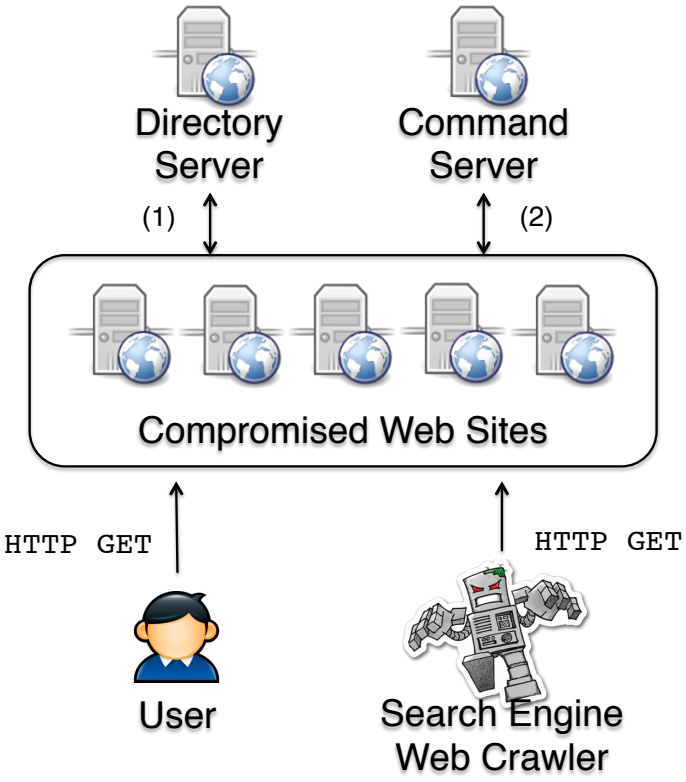


Figure 4.1. A user and a search engine Web crawler issue a request to a compromised Web site in the botnet. The site will (1) contact the directory server for the address of the C&C, and then (2) contact the C&C for either the URL for redirecting the user, or the SEO content for the Web crawler.

page, causing execution of the PHP file, the loader sets up a cache on the site's local disk. This cache reduces network requests, which could lead to detection or exceeding the Web site host's bandwidth limits. Then the loader will contact a directory server using an HTTP GET request to find the location of a command-and-control server (C&C) as either a domain name or IP address. Upon contacting the C&C server, the loader downloads the driver code which provides the main mechanisms used for performing black hat SEO.

4.2.2 Botnet Architecture

Figure 4.1 shows the high-level architecture of the botnet. The botnet has a command and control architecture built from pull mechanisms and three kinds of hosts: compromised Web sites, a directory server, and a command and control server (C&C).

Compromised Web Sites

Compromised Web sites act as doorways for visitors and are controlled via the SEO kit installed on the site. As detailed in Chapter 3, the SEO kit uses *cloaking* to mislead search engines, users, and site owners, as well as to provide a control mechanism for the botmaster. Cloaking is a mechanism that returns different content to different types of users based upon information gleaned from the HTTP request (Figure 4.1).

The SEO kit first checks to see if the user is a search engine crawler. If it is, the SEO kit returns content to the crawler to perform black hat search engine optimization. When the SEO kit is invoked via an HTTP GET request, the driver looks up the hostname of the visitor's IP address using `gethostbyaddr`. It then searches for the substring googlebot.com within the hostname to determine if Google's search crawler is accessing the page.[3] If the match is successful, the driver pulls SEO content from the C&C server and returns it to the crawler with the specific goal of improving the ranking of the page in search results independent of the original content of the page. Specifically, the driver builds a page with text and images related to the trending search results that link to the site. The SEO kit retrieves this content on demand by issuing auxiliary requests to search engines and *spinning* content constructed from the resulting search results snippets and images.[4]Additionally, the SEO kit inserts links to other nodes of the botnet, as directed

[3]It appears that the botmaster is only interested in poisoning Google's search results, as they solely target the Googlebot crawler—a trend also observed in Chapter 3.

[4]Spinning is another black hat SEO technique that rephrases and rearranges text to avoid duplicate content detection.

by the C&C, into the spun content to manipulate the search engine's ranking algorithms. As search engines typically use the number of backlinks to a page as one signal of high desirability [57], the botmaster aims to develop a linking strategy to improve the ranking of compromised sites in the SEO botnet.

If the SEO kit does not identify the visitor as a crawler, the driver next checks if the visit reflects user search traffic. The SEO kit identifies users by reading the `Referrer` field in the HTTP request headers, and verifying that the user clicked through a Google search results page before making the request to the compromised site. For these users, the SEO kit contacts the C&C server on demand for a target URL that will lead users to various scams, such as fake anti-virus, malware, etc., all of which can earn money for the botmaster. The SEO kit then returns this target URL together with redirect JavaScript code as the HTTP response to trigger the user's browser to automatically visit the target.

The SEO kit also uses its cloaking mechanism to provide backdoor access to the compromised site for the botmaster. To identify the botmaster, the SEO kit inspects the `User-Agent` field in the HTTP request headers, looking for a specific, unique phrase as the sole means of authentication. With this authentication token, the botmaster has the ability to read files from the local hard disk of the site, fetch URLs while using the compromised site as a proxy, run scripts pulled from the C&C, etc., all controlled through parameters to HTTP GET requests.

Finally, if the visitor does not match either the Googlebot crawler, a user clicking on a search result, or the backdoor, then the SEO kit returns the original page from the site before it was compromised. Thus, site owners who visit their pages directly will be unaware of the compromise.

Table 4.1. Timeline of SEO kit versions along with the capabilities (e.g., SEO techniques, redirect mechanisms and policies, cloaking techniques, Google Image Search techniques) introduced in each version.

Date	Version	Capability
Aug 6 2010	page v1	Build SEO page using Bing search results. User-Agent cloaking against Google, Yahoo, and Bing while ignoring “site:” queries. Redirect traffic from Google, Yahoo, Bing search using JS through gogojs.net.
Sep 22 2010	index v1.1	Reverse DNS cloaking against Googlebot.
Oct 6 2010	page v2.1	Use statistical model (# links, # images) to build SEO page. Also redirect traffic from Google Image Search. Redirect traffic with HTTP 30X and use cookie to redirect only once a day per visitor.
Mar 29 2011	page v4	Modify .htaccess to rewrite URLs and use Google Suggest terms for cross linking. Reverse DNS cloaking only against Googlebot.
Jul 15 2011	index v6 page v5	Hotlink images from Bing Image Search to help build SEO page. Proxy images instead of hotlinking.
Aug 18 2011	v7	index + page code branches merged. Morph proxied images. Redirect traffic using JS.
Sep 14 2011	v7.2	Clean other malware.
Sep 27 2011	vOEM	OEM terms targeted.
Oct 28 2011	vMAC	Mac OS X OEM terms targeted for low frequency traffic. Redirect traffic from any Google service due to referer policy change.
Mar 06 2012	v8	Only redirect Google Image Search traffic.

Directory Server

The directory server's only role is to return the location of the C&C server, either as a domain or IP address. Although relatively simple in functionality, it is the first point of contact from the compromised Web sites in the botnet and performs the important function of rendezvousing a compromised site with the C&C server. As a result, the directory server must be reachable and available and the SEO kit uses a typical multi-step process to locate it. The SEO kit will first attempt to reach the directory server through a hard-coded domain from the SEO kit, then a hard-coded IP address, before finally resorting to a backup domain generation algorithm (DGA) calculated using a time-based function. The directory server appears to have received little takedown pressure, though. We probed the potential backup domains up to a year into the future and found that no backup domains were registered, suggesting that this final fallback has not been necessary.

Command Server

The C&C server acts as a centralized content server where the botmaster stores data that the compromised sites will eventually pull down. The content is mostly transient in nature, and includes the trending search terms to target with SEO, the redirect URLs returned to users leading them to scams, and even the driver component of the SEO kit. This architecture allows the botmaster to make a single update that eventually propagates to all active nodes of the botnet.

4.2.3 SEO Kit Evolution

Examining the SEO kit's source revealed a variety of comments in the code. These comments were primarily written in Russian, suggesting the SEO campaign is implemented and operated by Russian speakers. From the translated comments we saw

hints of the existence of previous versions of the SEO kit in the wild, such as:

```
/**
 * v7.2 (14.09.11)
 * - Automatic cleaning of other malware
 *
 * v7.1 (05.09.11)
 * - Re-written for object oriented model
```

These indications of previous versions of the SEO kit motivated us to search for them using identifying substrings unique to the SEO kit code, such as "GR_HOST_ID". We discovered that previous versions were posted on the Web by site owners who were seeking assistance in deciphering the injected code on their site. After verifying older versions existed, we were able to download additional previous versions of the SEO kit from the C&C server by reverse engineering the protocol for downloading the driver and fuzzing likely inputs. In the end, we were able to download nearly all major SEO kit revisions since August 2010.

As seen in the sample above, the comments from each version of the SEO kit have a date and a short log message about the update similar to a version control system. From these comments, we reconstructed the developments in the SEO kit and thus the evolution of the SEO botnet and the botmaster's SEO strategies over two years. Table 4.1 summarizes our findings by presenting changes in capabilities with the corresponding version and date. Below are some highlights, many of which confirmed our early theories.

Structure. The compromised sites were at one time divided into *indexers*, which SEO-ed search engine visitors, and *doorways*, which redirected users, each with different cloaking mechanisms and policies. Starting August 2011, however, the code was merged into a single SEO kit with a unified cloaking mechanism and policy.

Cloaking. Initially, the doorways and indexers used User-Agent cloaking, where the server examines the `User-Agent` field in the HTTP request headers to identify user traffic and avoid detection. Specifically, the doorways used the cloaking mechanism to identify visitors who clicked through one of the three largest search engines: Google, Yahoo, Bing. By late September 2010, however, the indexers implemented the reverse DNS cloaking mechanism as described above. Similarly, by late March 2011 the doorways used the same cloaking mechanism and began targeting user traffic from Google exclusively.

Redirection. The redirection mechanism, used to funnel user traffic to scams, also changes significantly over time. Originally, the doorways redirected user traffic using JavaScript through an intermediary site, gogojs.net, which we suspect served as a traffic aggregation hop to collect statistics. By October 2010, the doorway redirected traffic via the HTTP 30* status with a cookie to limit visitors to one visit per day. Then in August 2011, the SEO kit returns to using JavaScript to redirect user traffic.

SEO. The SEO models and policies, used by the SEO kit to manipulate search result ranking, also change heavily over time. In the earliest version we have, the SEO page returned to search engine crawlers was generated from Bing search results. Then the SEO kit began using a statistical model when building an SEO page, requiring that the SEO page contents be composed of various percentages of text, images, and links. In late March 2011, the SEO kit used Google Suggest to target long-tail search terms. Then in late September 2011 it began to poison search results for OEM queries. And by late October 2011, the SEO kit started poisoning Mac OEM queries, also long-tail search terms.

Image Search. One of the surprising findings from the SEO kit code is the amount of effort placed in poisoning Google Image Search. The doorways first started redirecting user traffic from Google Image Search in October 2010. In July 2011, the

indexers hotlinked images from Bing to help build the SEO page and shortly thereafter the doorways began proxying images instead of hotlinking. By August 2011, the SEO kit began morphing the images, such as inverting them, to avoid duplicate detection. And currently, since March 2012, the SEO kit only redirects traffic from Google Image Search.

4.3 Methodology

We use data from three crawlers to track the SEO botnet and monitor its impact: (1) a botnet crawler for tracking compromised Web sites in the botnet and downloading SEO data from the C&C server, (2) a search crawler that identifies poisoned search results in Google, enabling us to evaluate the effectiveness of the botnet's black hat SEO, and (3) a redirection crawler that follows redirection chains from the doorway pages linked from poisoned search results to the final landing pages of the scams the botmaster uses to monetize user traffic. Table 4.2 summarizes these data sets, and the rest of this section describes each of these crawlers and the information that they provide.

4.3.1 Odwalla Botnet Crawler

We implemented a botnet crawler called Odwalla to track and monitor SEO botnets for this study. It consists of a host crawler that tracks compromised Web sites and a URL manager for tracking URL to site mappings.

Host Crawler. The host crawler tracks the compromised Web sites that form the SEO botnet. Recall from Section 4.2.2 that the SEO kit provides a backdoor on compromised sites for the botmaster through the HTTP request's `User-Agent` field. While this backdoor provides access to many possible actions, the default response is a simple diagnostic page with information about the compromised Web site such as:

```
Version: v MAC  1 (28.10.2011)
```

Table 4.2. The three data sets we use to track the SEO botnet and monitor its impact.

	Odwalla	Dagger	Trajectory
Time Range	October 2011 – June 2012	April 2011 – August 2011	April 2011 – August 2011
Data Collected	Diagnostic pages and cross links from nodes of SEO campaign.	Cloaked search results in trending searches over time.	Redirect chains from cloaked search results in trending searches.
Data Perspective	SEO Campaign botmaster.	Users of search engines.	Users of search engines.
Contribution	Characterize support infrastructure of SEO campaign.	Assess efficacy of SEO campaign.	Analyze landing scams.

```
Cache ID: v7mac_cache
Host ID: example.com
```

These fields show the basic configuration of the SEO kit: the version running on the compromised site, the version of the cache it is running, and the compromised site's hostname. The diagnostic page also reports a variety of additional information, such as the relative age of the SEO kit (for caching purposes), various capabilities of the Web host (e.g., whether certain graphics libraries are installed), and information about the requestor and request URL (e.g., whether the visitor arrived via Google Search). While the majority of this information allows the botmaster to debug and manage the botnet, we use the diagnostic page to both confirm a site's membership in the botnet and monitor the status of the compromised site.

The host crawler maintains a set of potentially compromised sites together with site metadata, such as the representative probe URL for a site and the last time it confirmed the site as compromised. The probe URL is the URL that the host crawler visits for each potentially compromised site. Since a given site may have many URLs that link to different pages, all managed by the same SEO kit, the host crawler maintains one active probe URL per site to limit crawl traffic. As URLs expire, a URL manager (described below) provides alternate probe URLs for a site. The host crawler visits each probe URL twice, once to fetch the diagnostic page and once to fetch the SEO page—the page returned to search engines—containing the cross links.

The last time the site was detected as compromised influences the crawling rate. The host crawler visits all sites that were either previously confirmed as compromised, using the diagnostic page mechanism described above, or newly discovered from the cross links. It crawls these sites at a four-hour interval. For the sites that were not confirmed as compromised, for example because it could not fetch the diagnostic page, the host crawler visits them using a two-day interval as a second chance mechanism.

If it does not detect a site as compromised after eight days, it removes the site from the crawling set. This policy ensures that we have near real time monitoring of known compromised sites, while limiting our crawling rate of sites where we are uncertain.

We used three methods to bootstrap the set of hosts for Odwalla to track. First, in October 2011 and then again in January 2012, we identified candidate sites using manual queries in Google for literal combinations of search terms targeted by the SEO botnet. Since the terms formed unusual combinations, such as "herman cain" and "cantaloupe", typically only SEO pages on compromised sites contained them. Second, since these pages contained cross links to other compromised sites for manipulating search ranking algorithms, we added the cross links as well. Interestingly, these cross links were insufficient for complete bootstrapping. We found multiple strongly connected components in the botnet topology, and starting at the wrong set of nodes could potentially only visit a portion of the network. Finally, we modified the SEO kit to run our own custom bots that infiltrated the botnet. These custom bots issued requests to the C&C server to download targeted search terms and links to other hosts in the botnet, providing the vast majority of initial set of bots to track. Once bootstrapped, the host crawler used the cross links embedded in the SEO pages returned by compromised sites to identify new bots to track.

URL Manager. The host crawler tracks compromised sites using one probe URL to that site at a time. Often a site can have multiple pages infected with the SEO kit, though, such as a site with multiple blogs, all of the comment pages attached to blogs and articles, etc. Over time, a site owner may remove or clean an infected page while other URLs to other pages on the site remain compromised and active with the same SEO kit. In these cases, the host crawler switches to a new URL to continue to track and monitor this compromised site.

The URL manager addresses this need. It maintains a list of all URLs, as

discovered from cross links, for a given site in the crawling set. It periodically checks whether each URL could potentially serve as the probe URL for a particular site by attempting to fetch a diagnostic page from that URL. Then, whenever the host crawler cannot fetch a diagnostic page for a site, it consults the URL manager to find another representative probe URL, if one exists. If not, the host crawler will continue to use the same probe URL, eventually timing out after eight days if all URLs to the site are not operational. In this case, it declares the site as "sanitized" since the SEO kit is no longer operational. Because there are far more URLs than sites, the URL manager crawls just once a day to check newly discovered URLs.

4.3.2 Dagger Search Crawler

Before we began crawling the SEO botnet, we previously explored the general dynamics of cloaking on the Web as described in Chapter 3. We knew from examining the code of previous versions of the SEO kit that the botnet poisoned trending search terms from April 2011 through September 2011, so we suspected that poisoned search results from the SEO botnet would also appear in our previous data set.

We had collected cloaking data using a crawler called Dagger, which ran every four hours to: (1) download trending search terms, (2) query for each trending search term on various search engines, (3) visit the page linked from each search result, and (4) run a cloaking detection algorithm to identify poisoned search results. The Dagger cloaking data allows us to analyze the impact of the SEO botnet on trending search results in near real time for a seven-month period (Section 4.4.3). Unfortunately, although we had continued to crawl cloaking search results, the SEO botnet changed its SEO policy to target first OEM software and then random search terms, so we would expect only accidental overlap with the Dagger data after September 2011.

4.3.3 Trajectory Redirection Crawler

While the host crawler downloads the contents of the doorway pages directly linked by poisoned search results, we also want to identify which sites these doorways ultimately lead to (e.g., a fake antivirus scam page) and hence infer how the botmaster monetizes user traffic. Since following the doorway pages to final landing pages typically involves following a complicated redirection chain, often involving JavaScript redirection code, we used the high-fidelity Trajectory crawler from yet another project [31]. This crawler uses an instrumented version of Mozilla Firefox to visit URLs, follows all application-level redirects (including JavaScript and Flash), logs the HTTP headers of the intermediate and final pages of a redirect chain, and captures the HTML and a screenshot of the final page. For all of the poisoned search results crawled by Dagger, we also crawled them using this Trajectory crawler to track scams.

4.4 Results

With the data sets we have gathered, we now characterize the activities of the SEO botnet and its compromised hosts.

4.4.1 Infrastructure

Using the nine months of data collected by Odwalla, we start by analyzing the botnet infrastructure used in the SEO campaigns: the scale of the botnet, the lifetime of compromised sites in the botnet, and the extent to which the botmaster monitors and manages the botnet.

Scale

Compared to other kinds of well-known botnets, such as spamming botnets with tens to hundreds of thousands of hosts, the SEO botnet is only modest in size. Figure 4.2

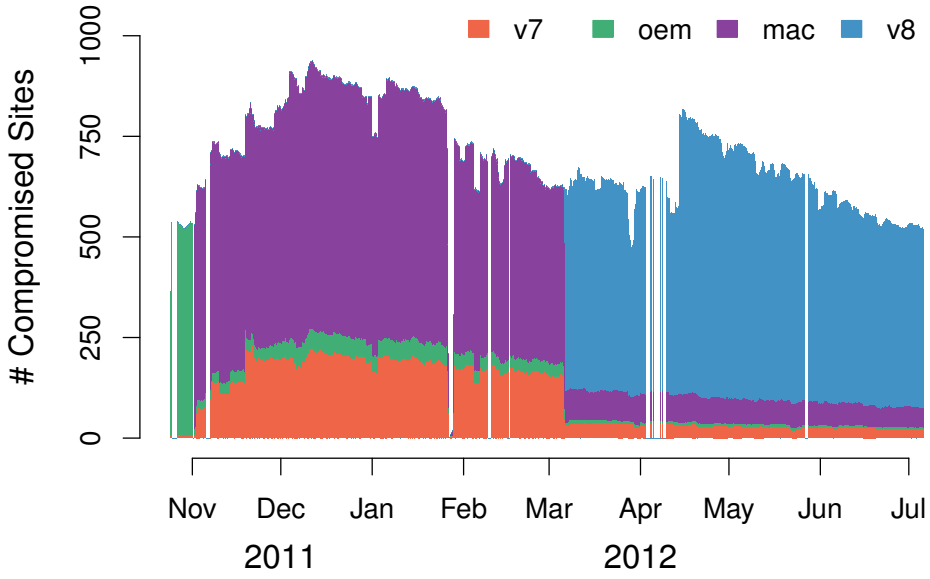


Figure 4.2. Stacked area plot of the number of active nodes in the botnet over time. Each colored area shows the number of nodes operating different versions of the SEO kit.

presents the measured size of the botnet over time as a stacked area plot. Each colored area shows the number of nodes operating a specific version of the SEO kit, and the sum of the areas shows the total number of all nodes across all versions. For example, on December 1, 2011, we found 821 compromised sites in total, of which 585 sites were running the MAC version of the SEO kit, 42 were running OEM, and 194 were running v7.

Also unlike other kinds of botnets, the SEO botnet does not exhibit frequent churn. Over nine months, the botnet consisted of 695 active nodes on average, with a maximum size of 939 nodes on December 11, 2011. Yet, we observed the botnet running on a total of just 1,497 unique compromised sites across the entire measurement period. In contrast, spamming botnets like Storm would experience churn of thousands of hosts a day [25].

Instead, we see a few key points in time where the botnet membership fluctuates in response to SEO kit updates by the botmaster, rather than from external intervention. At the time of the upgrades, the botmaster also changes the cross linking policy among nodes, potentially revealing new nodes. In between these upgrades, the botnet size primarily fluctuates due to variations in host availability, with a degree of slow attrition. For example, on November 1, 2011, the botmaster updated the SEO kit from OEM→MAC. Even though the OEM nodes appear to have entirely switched over to MAC, the size of the botnet increases by over 200 nodes, all due to nodes running the older version V7. It appears that during the update the botmaster changed the cross linking policy to include additional nodes running V7, incidentally widening our vantage point but only for stagnant sites running an older version. March 6, 2012, marks a similar version switch over from MAC→V8 in response to another software upgrade. In this upgrade, the 298 newly discovered compromised sites were running the latest version (V8), and were discovered a month later, due to what we suspect is the deployment time for a new cross linking mechanism that utilizes blogspot.com as a level of indirection. Note that the large drop in botnet size on January 28, 2012, corresponds to an outage on the directory server that triggered errors on the nodes, making the nodes unresponsive to our crawler.

As a final data point, recall from Section 4.2.2 that the GR botnet uses a pull mechanism to ensure that compromsied sites always run an updated version of the SEO kit. As a first step, a site makes up to three attempts to contact the directory server using first a hardcoded domain, then a hardcoded IP address, and finally the output of a time-based domain generation algorithm (DGA).

While crawling the botnet we found that both the directory server's hard coded domain and IP address were unreachable starting on September 9, 2012. We took advantage of this occurrence by registering the DGA domains that compromised sites will contact when attempting to reach the directory server. Thus, we pose as the directory server and

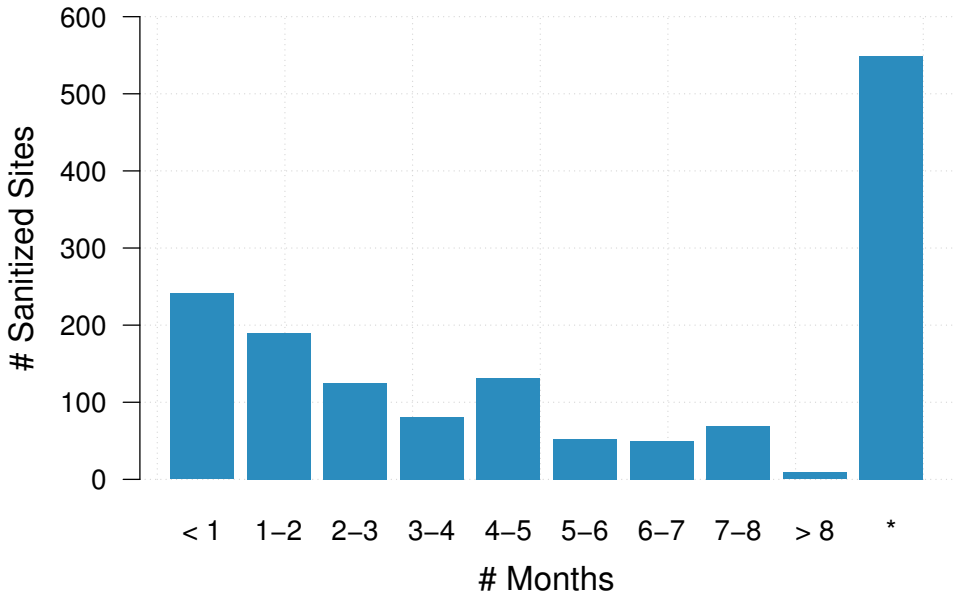
intercept all requests from the botnet's compromised sites for nearly a month between October 4 through October 30, 2012. From this vantage, we found that 1,813 unique IPs contacted our directory proxy. Since we found that, on average, 1.3 compromised sites are hosted behind a unique IP from the host crawler data, extrapolation places the botnet at 2,365 compromised sites—in agreement with our findings above that the GR botnet is on the scale of thousands of nodes.

Lifetime

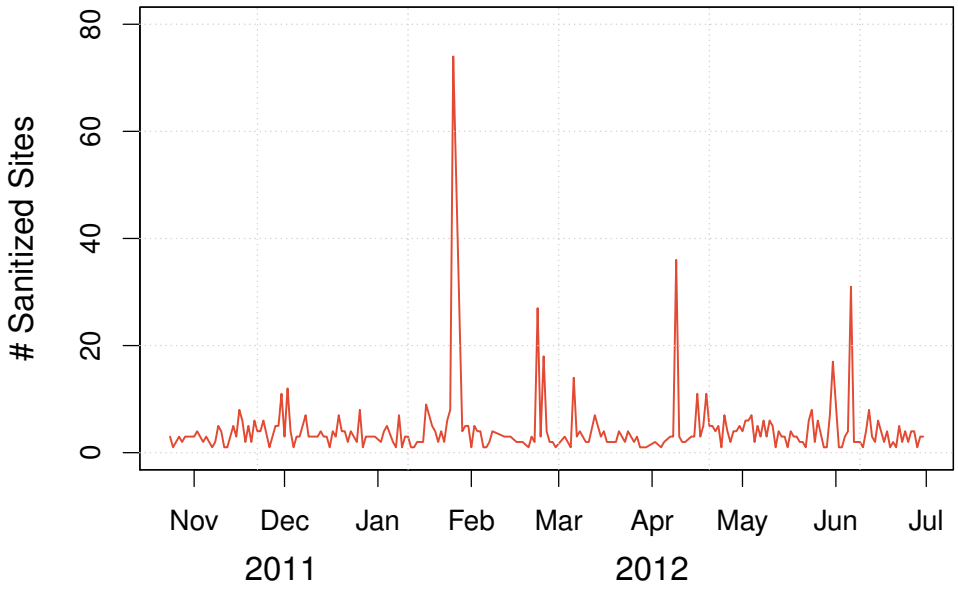
The relatively stable botnet size and membership suggests that compromised sites are long-lived in the botnet. Indeed, we find that many of these sites remain compromised for long periods of time and the botmaster is able to use them continuously without needing to constantly refresh the botnet with fresh sites to maintain viability.

We define the *lifetime* of a compromised site as the time between the first and last time the crawler observed the SEO kit running on the site. This estimate is conservative since a site may have been compromised before we first crawled it. However, we note that our measurement period of compromised sites is nine months and we began monitoring 74% of all 1497 compromised sites within the first 40 days of our study. Thus, even without the exact time of compromise, we are still able to observe the sites for long periods of time. (As further evidence, for the 537 sites that also appear in the earlier Dagger search results (Section 4.3.2) the majority were compromised back to April 2011.)

We decide that a site is cleaned when the site does not respond to the SEO C&C protocol for eight consecutive days, suggesting that the site no longer runs the SEO kit. Typically a site stops running the SEO kit because the site owner removed the SEO malware, sanitizing the site, or the Web host or DNS registrar made the site unavailable by preventing visitors from loading the site or resolving the domain.



(a) Lifetime



(b) Attrition

Figure 4.3. On top, the distribution of time that sites were compromised (sanitized sites only); the '*' bin shows the number of compromised sites still actively running the SEO kit at the end of the measurement period. For sites that were sanitized, the bottom graph shows the number of sites sanitized each day.

Consequently, the botmaster is able to use compromised sites for SEO campaigns for long periods of time. Figure 4.3a presents a histogram of the lifetime of the compromised sites. We distinguish between sites that have been sanitized, avoiding right-censoring of their lifetimes, and sites that have not yet been sanitized. For compromised sites that are eventually sanitized, we bin them according to their respective lifetimes using monthly intervals (30 days). Over 74% of sanitized sites have a lifetime greater than a month, and over 54% have a lifetime greater than two months. There is also a long tail, with the lifetime of some sanitized sites extending beyond even eight months. For compromised sites that have not yet been sanitized, we show them in the '*' bin. These remaining 549 sites are still compromised at the time of writing, and the majority of those have been compromised for at least seven months. This distribution indicates that the majority of compromised sites are indeed long-lived and able to support the SEO campaign for months with high availability.

Figure 4.3b shows the number of sites sanitized each day, indicating a low daily attrition rate of sites leaving the botnet over time (9.9 sites on average). The few spikes in the graph are specific points in time when many compromised sites were sanitized. In some cases, the spikes are partially attributable to a single entity, owning or hosting multiple sites, who cleans multiple sites at the same time. By manually comparing the resolved IP address for domain names as well as parsing WHOIS records, we were able to confirm shared hosting and shared owners, respectively. Note that the largest spike on January 26, 2012, corresponds to the outage of the botnet directory server.

One reason that sites remain compromised for long periods of time is that the SEO kit camouflages its presence to site owners. As discussed in Section 4.2.2, the SEO kit returns the original contents of the page to a visitor unless the SEO kit can determine if the visitor is a search engine crawler or has clicked on a result returned from a search engine. Hence, site owners accessing their own pages typically will not

notice an installed SEO kit. That said, even when they discover the presence of the SEO kit, oftentimes they are unable or unwilling to remove it. In December and January, for instance, we contacted nearly 70 site owners to inform them that their site was infected with the SEO kit, yet just seven sites subsequently removed it.

Control

We use two different approaches to assess the botmaster's ability to monitor and manage the botnet. In the first approach, we observe what fraction of the compromised sites update their SEO kit when the botmaster deploys a new version. We can detect both version changes and site updates by parsing the version information from the diagnostic pages periodically fetched by the host crawler.

As discussed in Section 4.4.1, the data collected by the host crawler overlaps with two version updates. On November 1, 2011, version OEM updated to MAC and then, on March 6, 2012, MAC updated to V8. In both cases, we see a near instantaneous update to the respective new versions from the majority of the compromised sites, followed by a sudden addition of newly seen compromised sites.

In the OEM→MAC update, we see many *stragglers*, sites that continue running older versions of the SEO kit after the majority of sites update themselves to the latest version. Within a month after the first update, 324 out of 970 sites (33%) that comprise the botnet were stragglers. These stragglers suggest that the botmaster lacks full installation privileges on the compromised sites and is unable to force an update. There is no advantage to running old versions because they poison an outdated set of search terms, are not well optimized in search results, and consequently will not attract much traffic. Therefore, the 324 stragglers represents a substantial inefficiency in the botnet. The straggler phenomenon also occurs during the second update, but the numbers are less pronounced.

Our second approach for assessing control looks at how the botmaster adjusts the cross linking policy once a compromised site is sanitized and no longer part of the botnet. Recall that each compromised site is cross linked to other compromised sites to increase search result ranking (Section 4.4.2). Therefore, when a site is no longer compromised, there is no value for the site to receive backlinks. Assuming the botmaster is actively monitoring the sites in the botnet, he should be able to adjust the cross linking policy to only link to sites that are still part of the botnet.

Using the set of sanitized sites described in Section 4.4.1, we track the number of backlinks received by each site over time from other compromised sites, noting whether a sanitized site still receives backlinks and for how long. In addition, we measure the average number of backlinks received before a site is sanitized, and after, to see whether the botmaster updates the cross linking policy to decrease the number of backlinks given to sanitized sites. Surprisingly, sanitized sites still overwhelmingly receive backlinks, and do so for long periods of time. Out of 508 sanitized sites, nearly all sites still receive backlinks even after being sanitized: all but two sanitized sites receive backlinks through February 26, and 488 (96%) through March 2.

In summary, it appears that the botmaster exerts only limited control over many compromised sites, letting many degrade over time. Further, this is but one of the inefficiencies in how the botnet is operated. While we do not have insight into the reasons for these lapses—whether negligence, lack of insight, or lack of need—the large numbers of stragglers and useless cross linking to sanitized sites makes it clear that in its existing regime the botnet does not reach its full potential impact.

4.4.2 Cross Linking

Next we examine the characteristics of the cross linking approach used by the SEO campaign to poison search results. Link “juice” is the SEO vernacular [52] for the

Table 4.3. The number of compromised Web sites grouped by the average amount of juice received, for the three distinct time ranges.

Group	<11/01	11/01 – 01/28	01/28 – 03/06
<10	532	949	834
10 – 100	71	28	31
100 – 1000	12	9	7

number of back links received from other unique Web sites, a well-known feature used by search algorithms when ranking Web pages [57]. Consequentially, one of the primary requirements for the SEO campaign to effectively poison search results is to artificially accumulate juice. Thus, by understanding the campaign's cross linking strategy, we are in a better position to counter search result poisoning.

The SEO botnet performs link farming where, using the terminology of [27], a small subset of compromised Web sites emulate *authorities* and receives substantially more juice than the other sites emulating *hubs*. This relationship lasts for an extended time period and ends when the botmaster rotates authority sites, with a different subset of compromised sites becoming authorities and receiving the dominant fraction of juice, and previous authorities becoming hubs. Link farming benefits the botmaster in a couple of ways. First, because there is a non-linear relationship between search result position and the amount of traffic clicking through the search result, the botmaster can attract more traffic by focusing on having the authority sites occupy a handful of top search result positions, rather than many low search positions using all compromised sites. Second, by selectively "juicing" a relatively small subset of authorities, the botmaster can limit the number of sites lost due to interventions by the site owner or defense mechanisms like Google Safe Browsing.

In our study, we identified two major authority rotations by monitoring when the amount of juice received by sites from the botnet changes substantially. Both occur in

Table 4.4. The number of compromised sites grouped by the total amount of juice received from blog posts, after the release of V8.

Group	**>03/06**
<10	665
10 – 100	250
100 – 1000	1
>1000	63

conjunction with major changes in the botnet. The first rotation occurs on November 1, 2011, when the SEO kit was updated from OEM to MAC. The second rotation occurs on January 28, 2012, when the botnet directory server experienced an outage (Section 4.4.1). In both cases, it seems the botmaster initiated the rotations because they appear related to version changes on the control server.

Table 4.3 summarizes the distribution of "juice" among compromised nodes in the botnet. It shows the number of compromised sites, grouped by order of magnitude of the average daily back links received by each site, for each time period. For example, in the first period from the beginning of the study to November 1st, 2011, there are 532 nodes that receive less than ten back links, 71 nodes that receive 10–100 back links, and 12 nodes that receive 100–1000 back links. Each time range has a consistent pattern: a small subset of sites (authorities) receive hundreds of back links, tens of sites receive tens of back links, and the remaining hubs receive less than 10 back links. We confirmed that the actual roles of compromised sites indeed changed from one period to another. For example, in the Nov 2011 rotation over 80 nodes had their juice reduced by at least an order of magnitude, while 20 nodes had their juice increased by at least an order of magnitude.

The top sites receiving the most juice are the most valuable to the botmaster since they have the most exposure in search results and attract the most traffic. At the same time, because they have the most exposure they are also the most vulnerable. To undermine

SEO botnets, targeting these sites first will be most effective, e.g., via blacklists like Google Safe Browsing.

As noted in Section 4.2, the release of V8 introduces a new cross linking mechanism that uses blogspot blogs as an extra layer of redirection. We do not directly compare the amount of juice observed using this new mechanism with the botmaster's previous approach because there are only two posts per blog, the last of which occurred in early April. Since this strategy rotates juice in sporadic large batches, rather than periodic increments, we focus the V8 cross linking analysis to data after March 6, 2012. As with the previous link farming strategy, though, we see a similar distribution of sites that emulate authorities (63) and hubs (665), albeit with a larger number of middle-sized hubs (251) as shown in Table 4.4.

4.4.3 SEO Effectiveness

Using the earlier data of the Dagger cloaking crawler, we next examine the ability of the SEO botnet to poison search results in Google. These poisoned search results represent doorways, which redirect users who click through the search result, leading to a destination of the botmaster's choosing. Thus, the doorways accumulate traffic for the botmaster to monetize. Therefore, we assess the potential threat posed by the SEO botnet by measuring the poisoned search results as seen from the user's perspective.

We find that the botnet can be quite effective in bursts, with thousands of poisoned search results targeting popular, trending search terms at any given time during the burst. At its peak, it is the dominant contributor to poisoned search results in Google from April through June 2011. The botnet is able to juice specific search terms with poisoned search results in Google within 24 hours, whereas it takes Google over 48 hours to start to counteract the poisoning.

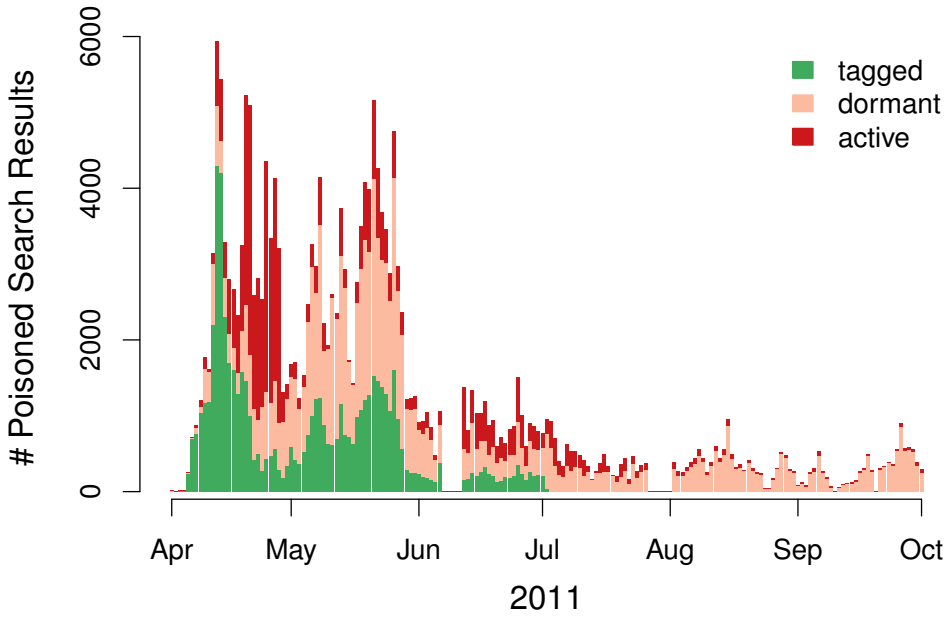


Figure 4.4. Quantity of poisoned search results attributable to the SEO campaign. Each bar shows the number of poisoned results that are redirecting users, dormant, or tagged by Google Safe Browsing.

Quantity

Ultimately, the goal of the SEO botnet is to attract user traffic to its compromised sites by manipulating search results. We first evaluate the effectiveness of the SEO botnet in achieving this goal by analyzing the placement of its sites in search results.

Using the set of compromised sites enumerated by the host crawler, we identify the botnet's poisoned search results using the URLs that link to a compromised site. We then characterize each poisoned search result into one of three states over time: active, tagged, or dormant. *Active* poisoned search results are cloaking and actively redirecting users. Users who click on these search results will be taken to an unexpected site, such as fake AV, to monetize their clicks. *Tagged* results have been labeled as malicious by Google Safe Browsing (GSB) [16], presumably discouraging users from visiting and preventing the botmaster from significantly monetizing traffic to these URLs.

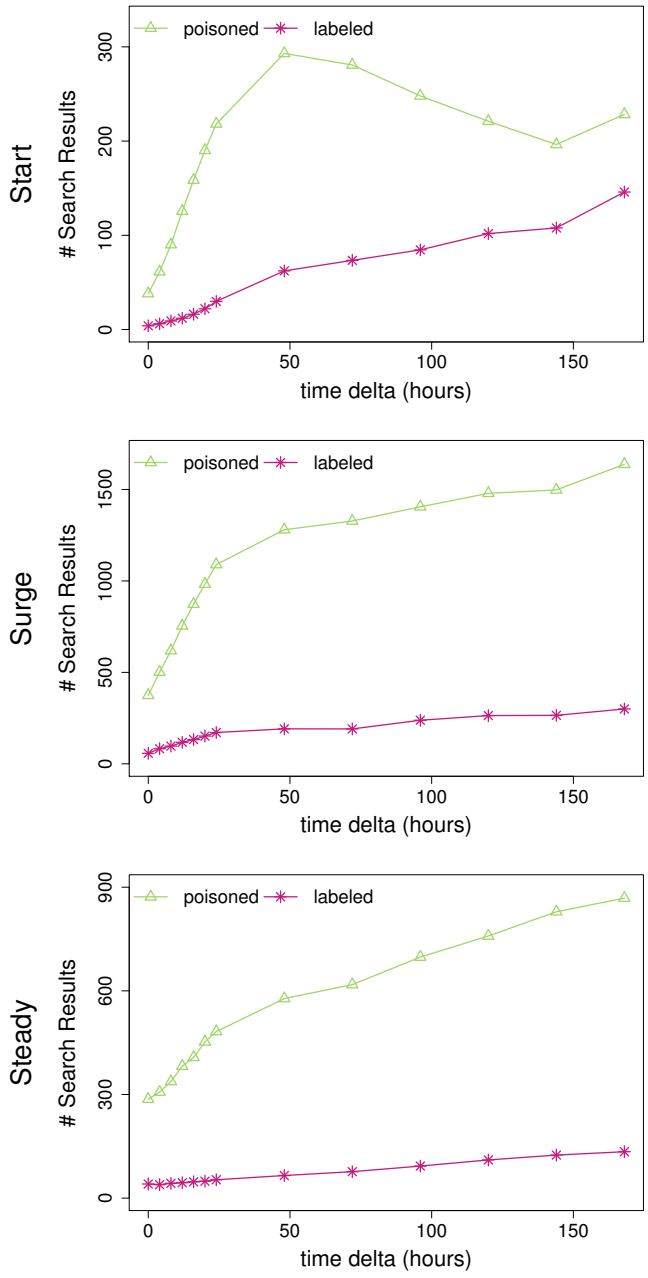


Figure 4.5. The number of poisoned search results attributable to the SEO campaign, when the same query is retried after a time delta. The POISONED line represents poisoned search results that have not been labeled by GSB, whereas the LABELED line represents poisoned search results that have been labeled by GSB.

GSB blacklists URLs that lead to phishing and malware pages. Although not all pages the botnet uses to monetize traffic may fall under GSB's purview, when GSB labels those pages that do it is a useful indicator of defenses undermining the botmaster's SEO campaign. *Dormant* poisoned search results are cloaking but not redirecting. These search results lead to sites that apparently have redirection disabled, and the botmaster no longer derives value from them.

Figure 4.4 shows the evolution of an SEO campaign over time as viewed by the prevalence of the various kinds of poisoned search results. Over six months, we saw four main periods of activity. In a starting period, from April 1st to April 18th, most poisoned search results were tagged yet the volume of active remained high. On April 15th, for example, we observed 2,807 poisoned search results, of which 1,702 search results were tagged, 721 were active, and 384 were dormant. The tagged and dormant search results are the remnants of the previous SEO campaign by this botnet, while the growing number of active results reflects increasing momentum of a new campaign.

The start period transitioned into a *surge* period from April 18th to April 28th, where a surge in active poisoned search results corresponds with a decline in tagged. This surge reflects the campaign promoting a new set of heavily "juiced" hub sites (Section 4.4.2). This 10-day window shows the botnet SEO campaign at its peak, with most poisoned search results actively monetizing traffic before Google Safe Browsing and site owners can react.

A third *steady* period, from April 28th to June 30th, exhibits a substantial decrease in active poisoned results and a corresponding increase in tagged and dormant results. These results are no longer effective for the botmaster since they have either been flagged by GSB to warn away uses or the sites have been sanitized by their owners.

After June is an *idle* period where the total volume of poisoned search results declines substantially, coinciding with the timeframe of an organized intervention into the

fake AV industry by the FBI [28]. From July through October 2011 the SEO campaign had results linked to compromised sites that remain tagged or dormant, but only a negligible number of sites were actively redirecting. It highlights the successful impact of interventions that undermine the vector by which the botmaster could monetize traffic, like the fake AV takedown. Undermining monetization removes the key incentive for the botmaster to keep SEO campaigns active.

Temporal

Section 4.4.3 assesses the SEO campaign's activity level over time. However, quantity alone does not give a complete portrayal of the volume of poisoned search results and their coverage in spamming specific search terms. For example, on April 23, 2011, Dagger queried "the ten commandments list" and found one poisoned search result from this SEO campaign. Then, 16 hours after the initial query, Dagger re-queried "the ten commandments list" and found 20 poisoned search results. We suspect the increase in poisoned search results is due to the increased time available for the campaign to SEO their sites. Regardless, these dynamics demonstrate the importance of the time component in conveying impact. Thus, to assess the botnet's potential threat through volume and coverage of specific search terms, we also measure the quantity of poisoned search results for the same query at subsequent points in time.

Recall that Dagger repeatedly queries for the same search terms over time to enable precisely these kinds of temporal analyses (Section 4.3.2). Figure 4.5 presents the number of poisoned search results attributable to the SEO campaign when the same query is repeated for varying time deltas. For each search term, the zero time delta is when Dagger first sees poisoned search results for that search term. We show results separately for the start, surge, and steady periods to highlight differences among them.

Each graph contains two lines. The POISONED line represents the number of

poisoned search results that have not been labeled by Google Safe Browsing, averaged across the entire data set. We use the same methodology as above, except here we apply it to the additional temporal data and we do not distinguish between active and dormant poisoned search results. Conversely, the LABELED line represents the average number of poisoned search results that have been labeled by GSB. Not surprisingly, we see the same results as before. The start period has a mixture of poisoned search results and labeled search results from a previous SEO campaign. Then there is a burst of poisoned search results during the surge period, and a steady stream of poisoned search results in the steady period.

The number of poisoned search results from their first appearance in a query for a search term is just the tip of the iceberg. In other words, within hours of the initial appearance, users are likely to encounter a flood of poisoned search results. Although applicable for all time periods, it is most prominent during the surge period as the number of poisoned search results seen increases nearly 3× from 374 to 1,089 within 24 hours. In the start period, we see an increase from 38 to 218 within 24 hours, and in the steady period we see an increase from 286 to 482 within 24 hours.

Further, at the start of a new campaign GSB lags the increase in poisoned search results that arrive shortly after the initial appearance: the slope of the POISONED lines is higher than the LABELED lines during the surge and steady period. Only when the campaign enters the start period does GSB begin to react swiftly to the later arriving poisoned search results (the dip after 48 hours in the POISONED line).

Market Share

Intersecting the data from the botnet crawler and the search crawler also allows us to compare the relative size of this SEO campaign against all other similar SEO campaigns in terms of the volume of poisoned search results. Figure 4.6 shows the number of cloaked

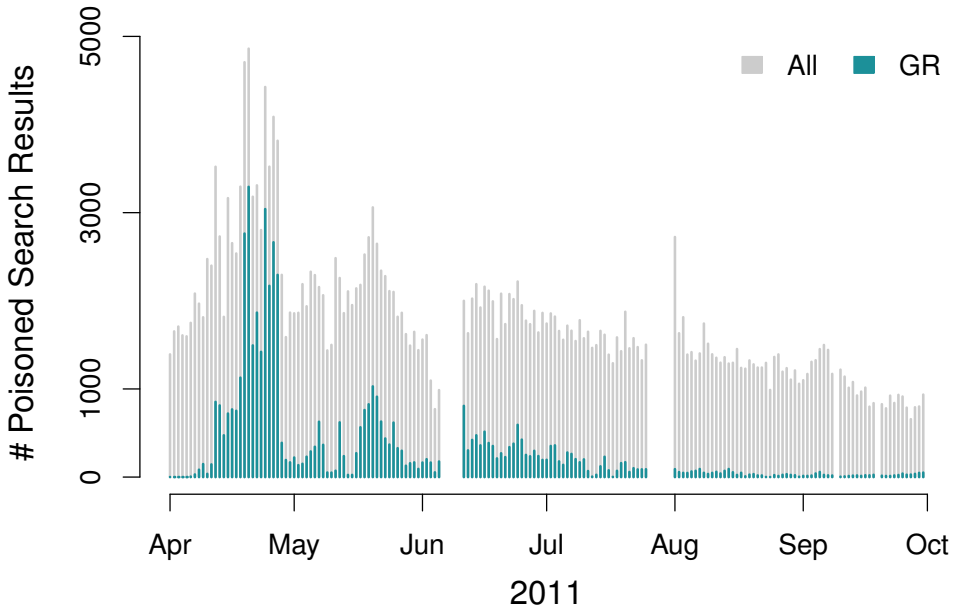


Figure 4.6. Comparison between this SEO campaign against all actively redirecting poisoned search results.

search results over time found by Dagger, and then the subset attributed to just this SEO campaign. During the surge period, this campaign accounted for the majority of cloaked search results at 58%. This campaign was most prominent on April 24th, when 3,041 out of 4,426 (69%) poisoned search results came from this single campaign. Even as the surge decreased in the steady period, the SEO campaign still accounted for 17% of all active poisoned search results observed. As a result, not only is this SEO campaign one of the main contributors of poisoned search results, it has demonstrated the potential to poison more search results than all other competing SEO campaigns combined for an extended duration of 10 days.

Active SEO Duration

Similar to Section 4.4.1, we next quantify how long the botmaster is able to effectively utilize compromised sites as doorways for funneling users to scams. Compared

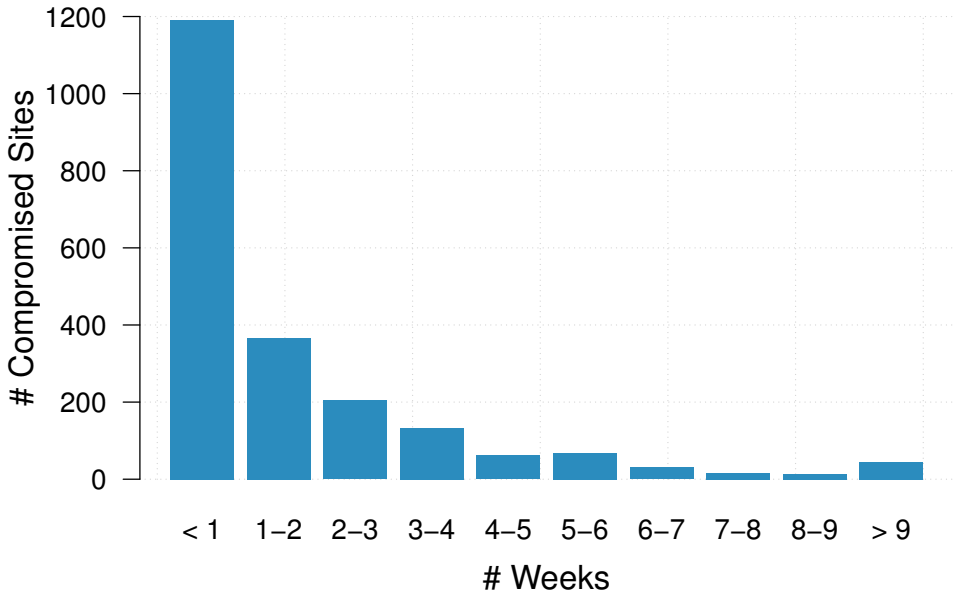


Figure 4.7. Duration of compromised sites in poisoned search results that actively redirect users to scams.

to lifetime, which measures how long a site remains part of the botnet, here we focus on active duration, the total amount of time that the site is both exposed to users through poisoned search results and actively redirects them to scams.

For each compromised site we collect all occurrences of poisoned search results to the site observed from April 1 to September 30, 2011. In addition, we track whether the poisoned search results were redirecting users and whether they were labeled by GSB. We use the first occurrence of a poisoned search result to a site as the start of the site's active duration. We end the active duration when we do not see another occurrence of a poisoned search result, attributable to this site, within the following three weeks of the last result.

In this six-month period, 3,822 compromised sites from this campaign were involved in poisoning search results. Of these, 2,128 sites (56%) actively redirected users.

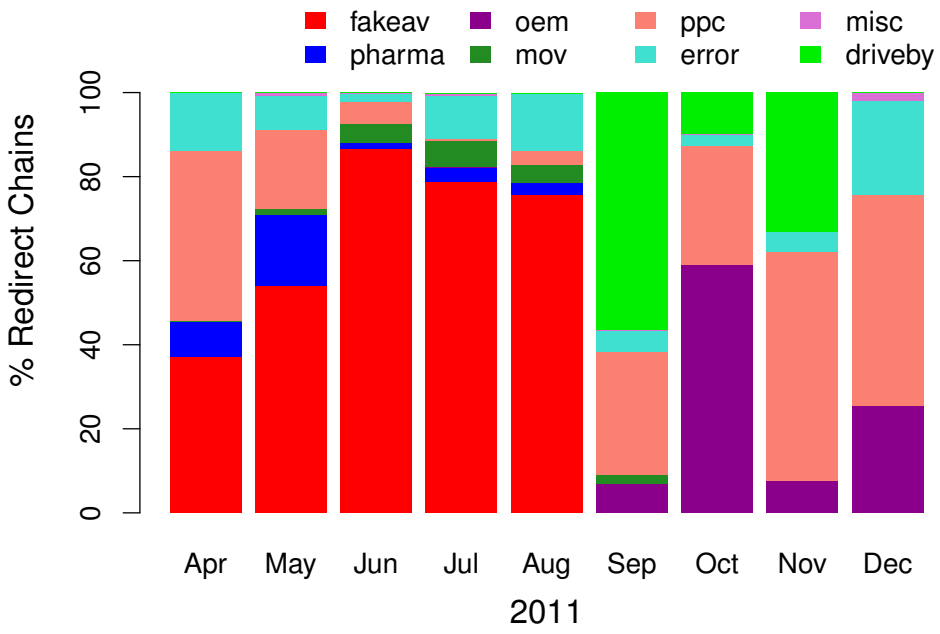


Figure 4.8. Relative breakdown of the categories of scams that poisoned search results ultimately take users.

Figure 4.7 shows a binned histogram of the number of sites that poison search results and actively redirect users at a week granularity. A majority of actively redirecting sites (56%) have durations of less than a week. Of the remaining 937 sites that effectively poison search results and redirect users for longer than a week, most (89%) survive 1–6 weeks.

4.4.4 Monetization

Ultimately, the botmaster operates and maintains the GR botnet to make money. Compromising sites, cloaking and traffic segmentation, cross linking and poisoning search engine results, etc., are all component parts of a black hat SEO machine engineered to make a profit.

Scams Targeted

Using the data from the Trajectory redirection crawler, we categorize the redirection chains that lead users from poisoned search results to the different scams used by the GR botnet to monetize traffic. Specifically, we selected redirection chains that: (1) originated from one of the doorway pages, (2) contained more than one cross site redirection, and (3) occurred while mimicking a Microsoft Windows user running Internet Explorer; as discussed below in Section 4.4.4, the majority of redirect chains observed while mimicking a non-Windows user generally led to the RivaClick pay-per-click affiliate program. We manually clustered the redirection URLs based on similar URL characteristics, such as the same PHP file with the same HTTP GET parameters and arguments. For example, although http://model-seil.ru/afro/index.php and http://softwarename.ru/protect/index.php appear to represent two separate hosts, they in fact resolve to the same IP address. After clustering, we constructed a network graph starting from doorways and ending at different kinds of scams. This graph allows us to trace the scams where the botmaster was an affiliate.

In Chapter 3 we noted that SEO campaigns in general shift between different affiliate programs over time. Therefore, for this analysis we arbitrarily divided the redirect chains by the month when the chain was observed. Figure 4.8 shows the relative breakdown of the kinds of scams that the redirection chains take users. The "misc" category refers to crawls that we could not classify, such as redirections that ultimately led to the Google search page, and "error" are crawls that returned an HTTP error code or screenshot.

We see two distinct periods of scam targeting, with the transition between the two coinciding with the 2011 fake AV takedown [28]. Early on, from April through August, the botnet redirects the majority of poisoned search results to fake AV programs,

presumably because of their profitability [56]. We also see a varying amount of redirection chains leading to counterfeit pharmaceutical programs, including the GlavMed, Mailien, and RX-Partners programs, although not nearly as prevalent as fake AV. From June through August, we also see an increase in the proportion of search results directed to movdl.com, a pirated media affiliate program. Redirection chains to movdl.com stop in September, though.

After the fake AV takedown, the botmaster markedly changes the scams targeted. In September, we see the intermediary node that sent traffic to the one remaining fake AV program now sending traffic to a drive-by download affiliate program. This target is also temporary, as by October the botnet updates the SEO kit to version OEM and redirects the majority of the traffic to OEM affiliate programs (TheSoftWareSellers and OEMPays), which continues until December when we found that the GR botnet stops redirecting. Finally, pay-per-click is notably a steady safety net throughout, and we explore it in more detail next.

RivaClick Traffic Affiliate Program

Recall from Section 4.2.3 that we downloaded past versions of the SEO kit. While crawling these past versions, we found that the SEO campaign was actively redirecting users to the URL:

http://www.rivasearchpage.com/?aid=2277&said=0&n=10&q=[query]

This URL leads to a feed of URLs for the RivaClick Traffic Affiliate Program. RivaClick operates similarly to other Internet advertising platforms. There are advertisers who want to buy traffic for a specific topic of interest, usually determined by the user's query string, and there are publishers who sell traffic. RivaClick groups the advertisers' links into a feed, which is provided to publishers who will receive commissions on click traffic to links from the feed. An important difference between RivaClick and other

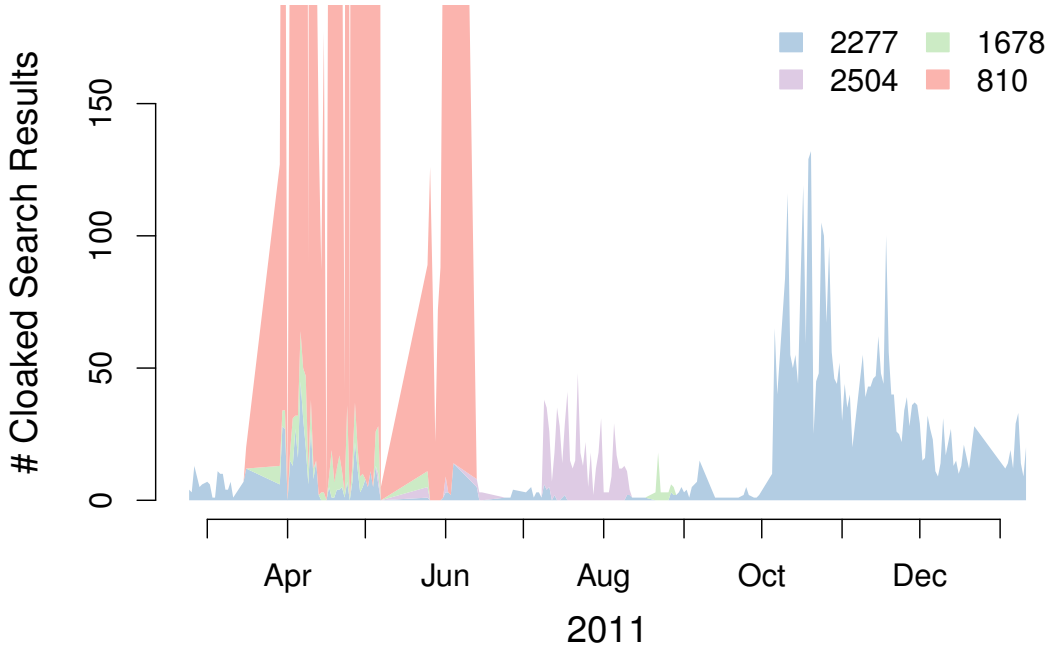


Figure 4.9. Stacked area plot of the Number of poisoned search results that lead to RivaClick over time. Each colored area represents a unique affiliate ID. The y-axis is truncated at 150 to show details (the max y-value for an affiliate is 1,231).

advertising platforms is that RivaClick provides little guarantees about the quality of the traffic being sold, which allows publishers to dump traffic obtained through search result poisoning. Based on the URL extracted from a previous SEO kit and the HTTP GET parameters from the URL, it appears that the botmaster is an affiliate of RivaClick with ID 2277.

With this affiliate identifier in hand, we retroactively examined poisoned search results from the Trajectory crawler starting in March 2011. One pass of the search crawler captures the entire redirect chain for poisoned search results, from the doorway, returned when the user first clicks the search result, through all the intermediary hops, and finally the landing page. In this section, we focus on redirect chains that landed on RivaClick.

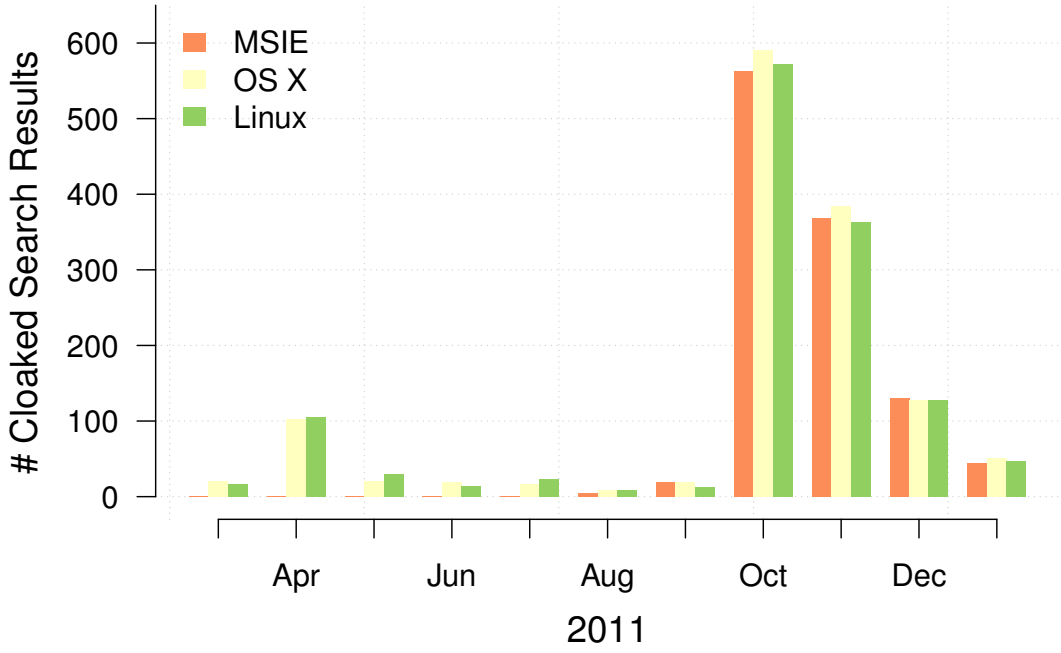


Figure 4.10. Number of poisoned search results that lead to RivaClick depending on the OS/browser.

Figure 4.9 shows the quantity of poisoned search results funneled into RivaClick per day for the four most-frequently seen affiliates: 810, 1678, 2277, and 2504. Because we found these affiliates performing search result poisoning, we assume they are running similar black hat SEO campaigns. Therefore, we compare the four affiliates to provide a sense for the relative size of the GR botnet and its peers and competitors in terms of the number of search results leading to RivaClick (a more focused comparison than Figure 4.6, which is in terms of all poisoned search results).

The GR botnet, as affiliate 2277, redirected a small but steady number of search results to RivaClick for much of 2011, but then significantly increases results to RivaClick starting in October 2011 after the fake AV takedown. Meanwhile, the other affiliates were burstier. 1678 directed a small burst from April–May, and 2504 directed a burst

from July–August. Finally, 810 redirected bursts from March–June, but with far more intensity (max of 1231 on June 2). As a result, it appears that 2277 is a long lasting, relatively mid-size SEO affiliate of RivaClick.

Figure 4.10 focuses more closely on affiliate 2277. The Trajectory search crawler visited each poisoned search result while mimicking three different browsers: Microsoft Internet Explorer running on Windows, Mozilla Firefox running on Mac OS X, and Mozilla Firefox running on Linux. These three visits enable us to analyze the traffic segmentation policy employed by the botmaster based on browser and operating system. Indeed, it appears that such demultiplexing occurred from March through September. As seen in Figure 4.10, only Mac OS X and Linux traffic led to RivaClick. Starting in August, when the botmaster could no longer monetize Windows traffic through fake AV scams, traffic from all platforms were redirected.

4.5 Summary

Overall, we find that with modest resources the GR botnet can be very effective in poisoning search results, becoming for months at a time the dominant source of poisoned results. At the same time, we have seen two kinds of intervention against the SEO botnet. The first targets the botnet directly, its infrastructure (compromised sites) and vector (poisoned search results). Given that sites remain compromised for months, cleaning up sites has not been effective at undermining the botnet; indeed, even when we explicitly notified site owners about the malware, few reacted or responded. Google, however, is more responsive, tagging poisoned search results within a couple of days—but that window is still presumably effective for the botmaster given the intensity of the SEO activity. The second undermines monetization, and appears to be much more effective. With evidence of the importance of a "killer scam" in monetizing and driving innovation in SEO campaigns, we observe substantially more activity from the botnet when the fake

anti-virus market is stable, whereas the botmaster appears to scramble to monetize traffic when the fake anti-virus market is in flux and the GR botnet becomes relatively idle. Undermining monetization appears to be a potent response to these types of attacks.

Chapter 4, in part, is a reprint of the material as it appears in Proceedings of the Network and Distributed System Security Symposium 2013. Wang, David Y.; Savage, Stefan; Voelker, Geoffrey M. The dissertation author was the primary investigator and author of this paper.

Chapter 5

Search and Seizure: The Effectiveness of Interventions on SEO Campaigns

In this chapter, we focus on interventions. However, unlike previous chapters, which concentrate squarely on the lower-level mechanisms supporting search poisoning, this chapter instead focuses on the attacker's high-level operation, their SEO campaign. In particular we perform an ecosystem-level analysis, focusing specifically on the cat-and-mouse relationship between SEO campaigns and interventions.

5.1 Introduction

Every new communications medium inevitably engenders a new form of abuse — telephones led to unsolicited "robocalls", e-mail begat spam, and so on. In turn, new mechanisms and policies are invariably brought to bear to restrict such activities (e.g., spam filters or, in the U.S., the national do-not-call registry). Today, one of the most dynamic such conflicts is playing out in the medium of online search.

In particular, as online marketing has become the leading mechanism by which sellers of goods and services engage potential consumers online, search engines, such as Google and Bing, have become the primary platform of this engagement. Because search engine results are presented directly in response to user queries, they offer the opportunity

to precisely target consumers at the moment of their interest. As a testament to this, search engines received over $16B in revenue in 2012 (46% of the total online advertising expenditures) for clicks on sponsored advertisements appearing in their search engine result pages (SERPs) [46].

However, while criminal use of sponsored advertisements occurs, the more fertile ground for abuse is the so-called "organic" search results, which are unpaid. These results are generated and ranked automatically based on the content and structure of the visible Web (e.g., based on the PageRank algorithm, the presence of user-generated content, etc.) and produce far more click traffic than sponsored ads. Unsurprisingly, techniques for improving the ranking of particular Web sites in these organic search results — termed search engine optimization (SEO) — are extremely popular. While some SEO techniques are completely benign (e.g., keyword friendly URLs), quite a few are actively abusive (e.g., the use of compromised Web sites as "doorway" pages, "cloaking", farms of "back links", etc.). As a result, such "black hat" SEO campaigns are frequently able to poison search results so that one or more highly-ranked results for key search terms will direct traffic to their sites. This traffic can then be monetized by infecting the user with malware [20, 23, 47], defrauding the user via phishing [62], or through the marketing of counterfeit or illegal goods (e.g., pharmaceuticals [38]).

In this chapter, we focus on a range of such SEO campaigns that are the principal means of marketing for organizations selling counterfeit luxury and lifestyle fashion goods. To wit, at the time of this writing, typing "cheap louis vuitton" into Google produces a list of ten results. Fully seven of these are fraudulent and ultimately direct user clicks to storefronts selling counterfeit knockoffs of Luis Vuitton products. This is no exception and similar search result poisoning is evident for a range of luxury brand names. Indeed, the combination of both high demand and high margins (a counterfeit of a handbag that might retail for $2400 will sell for $250, but will typically cost as little as

$20 to produce) make this a vibrant and profitable scam; we have evidence that a *single* fulfillment organization delivered over 250,000 such items over a nine-month period. However, such actors are not unopposed and there are a range of interventions they must contend with including labeling and deranking of their sites by search engine operators, and site or domain takedowns driven by brand holders. It is understanding the interplay of SEO campaigns and these interventions that motivates our research.

Concretely, this chapter makes three contributions. First, we provide the first large-scale empirical characterization of SEO abuse for luxury brands. In particular, we explain how such scams work (identifying how they differ from existing markets technically and operationally), analyze their search placement success over time and, using the prior "purchase pair" technique [26], gather indicators of order flow volumes. Second, we develop a methodology for using this data to evaluate the impact of interventions on the effectiveness of these SEO campaigns. Finally, we apply our methodology to a range of existing anti-counterfeiting actions, identify why these prior efforts have had limited impact and make suggestions for improving their utility in the future.

5.2 Luxury SEO and Interventions

Abusive SEO campaigns, by definition, can victimize two groups, users and search engine providers. The former because they may be convinced to purchase goods or services that are of low quality or illegal, the latter because their ability to provide high quality search results is imperiled. However, within the niche of counterfeit luxury goods another potential victim is the luxury brands themselves (both in terms of lost potential sales and brand damage). Consequently, in addition to interventions from search engines (driven by general concerns about search quality), brands also drive interventions to protect their economic interests. In this section we discuss what makes this market distinct, both in terms of how counterfeit luxury SEO campaign are structured and the

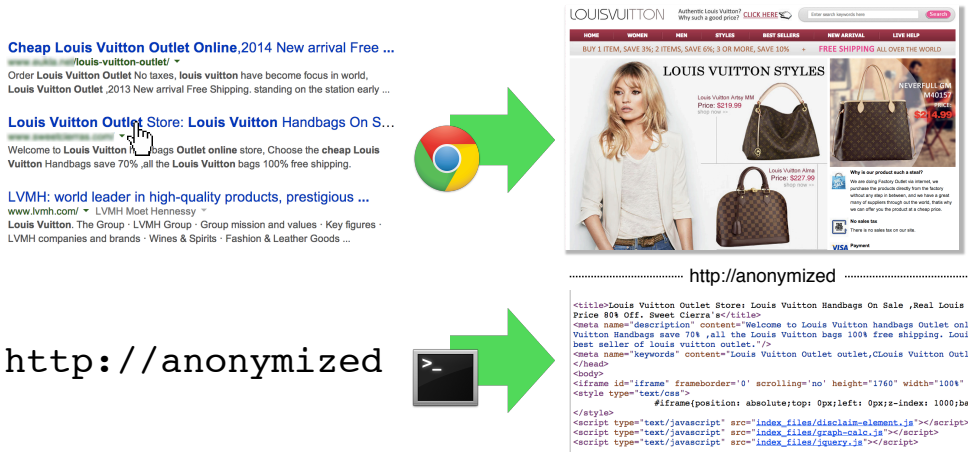


Figure 5.1. An example of iframe cloaking where the same URL returns different content for different visitor types. Above, a user clicks through a search result and loads a counterfeit Louis Vuitton store. While, below, a search engine crawler visits the same URL directly, receiving a keyword-stuffed page because the crawler does not render the page. Our crawlers mimic both types of visits.

kinds of interventions used in response.

5.2.1 SEO Campaigns

The SEO campaigns funded by the counterfeit luxury goods market operate similarly to other SEO campaigns (see Chapter 4 for one such example), with a couple of noteworthy differences. First, they introduce distinct cloaking and evasion techniques designed to undermine existing defenses. Second, the businesses that ultimately fund these campaigns appear to be organized differently than the open affiliate marketing programs that have been endemic in prior studies of underground economics (e.g., counterfeit pharmaceuticals [38], software [37] or FakeAV [56]). We discuss each of these in turn.

Cloaking

At its essence, cloaking refers to any mechanism for delivering different content to different user segments. For the purposes of SEO, cloaking's primary objective is

to deceive search engines by providing different content to the search engine crawler than to users clicking on search results. For example, the most widely-used cloaking technique, called *redirect cloaking*, arranges that search engine crawlers (e.g., Googlebot) receive content crafted to rank well for targeted query terms, while normal users who access the site are instead redirected to another site hosting a particular scam (e.g., a storefront selling counterfeit goods). In some cases, particularly when the doorway is on a compromised site, a visitor will only be redirected after arriving via a search results page. Otherwise, the original legitimate site content is returned, enabling compromised sites to remain compromised longer by appearing unchanged to normal visitors.

However, cloaking is a violation of most search engine's content guidelines and, when such activity is discovered, the cloaked sites are typically deranked automatically in search results. As with any adversarial process, though, attackers adapt to new defenses. In contrast to cloaking techniques we have previously observed in Chapter 3, we have identified a new method of cloaking, which we call *iframe cloaking*, which bypasses traditional means of detection. In particular, iframe cloaking does not redirect the user and frequently returns the same content to both search engines and users.[1] Instead of redirecting a user to a landing store site, the store is simply loaded within an iframe element on top of the existing doorway page content. Typically the iframe visually occupies the entire height and width of the browser to provide the illusion that the user is browsing the store (Figure 5.1 shows a simple example of iframe cloaking using JavaScript). The JavaScript implementation is frequently obfuscated to further complicate analysis and in some cases the iframe itself is dynamically generated. Taken together, these countermeasures require any detection mechanism to run a complete browser that

[1] A complementary feature of iframe cloaking is that it reduces the requirements for cloaking on compromised sites. Traditional cloaking uses network features (e.g., IP address or user agent) to identify crawlers, requiring specialized server side code. In contrast iframe cloaking runs entirely on the client, relying on the assumption that crawlers do not fully render pages at scale.

evaluates JavaScript and fully renders a page (a set of requirements that greatly increase the overhead of detection at scale).[2] We found the use of iframe cloaking to be pervasive within the domain of counterfeit luxury, but a more comprehensive study of the use of iframe cloaking for other domains remains an open question.

Business structure

Traditionally, a broad range of online scams have been organized around an affiliate marketing model in which an affiliate program is responsible for creating site content, payment processing and fulfillment, while individual affiliates are responsible for delivering the user to storefronts (e.g., via e-mail spam, SEO, etc.). Core to this business model is the notion that affiliates are independent contractors agents paid on a commission basis, and thus affiliate programs work to attract a diverse set of affiliates. This model is commonly used today in a broad range of scams with a nexus in Eastern Europe and Russia including pharmaceuticals, pirated software, books, music and movies, herbal supplements, e-cigarettes, term paper writing, fake anti-virus and so on [50].

However, there are many indications (albeit anecdotal) that the structure of organizations in the counterfeit luxury market are distinct.[3] First, the marketing portion of these scams can span both an array of brands and types of merchandise. For example, from infiltrating their command and control (C&C) infrastructure using the same approach as described in Chapter 4, we find a single SEO campaign may shill for over ninety distinct storefronts selling thirty distinct brands ranging from apparel (Abercrombie), luxury handbags (Louis Vuitton), and electronics (Beats By Dre). Moreover, the same

[2]Even after rendering a page, the ubiquity of iframes in online advertising make distinguishing benign from malicious content a challenge.

[3]There is a range of evidence suggesting that the big counterfeit luxury organizations have a nexus in Asia, unlike the Eastern European origin of many other scams. Our evidence includes the use of Asian language comments in SEO kit source code, the choice of Asian payment processors, fulfillment and order tracking from Asia and direct experience interviewing an Asian programmer working for one of these organizations. We surmise that a distinct cybercrime ecosystem has evolved separately in East Asia with its own standard practices and behaviors.

campaign will commonly host localized sites catering to international markets (e.g., United Kingdom, Germany, Japan, and so forth). Unlike other kinds of counterfeit sales, which centralize payment processing within the affiliate program [26, 38, 56], we find each counterfeit luxury storefront allocates order numbers independently and engages directly with payment processors (merchant identifiers exposed directly in the HTML source on storefront pages allowed us to confirm). Finally, in the traditional affiliate program model, fulfillment is managed internally by the program, but in our investigations we have found at least one fulfillment site for luxury goods that appears to be designed to support outside sales on an *á la carte* basis (i.e., the site is designed to support wholesale ordering and allows each member to track the order status of their customer's shipments). Overall, we suspect the counterfeit luxury ecosystem does not use an affiliate program and instead the ecosystem is composed of several independent advertisers (SEO campaigns) contracting with third parties for fulfillment and payment processing.

5.2.2 Interventions

As we have observed, the two groups with natural incentives to disrupt SEO campaigns targeting counterfeit luxury goods are search engine providers and luxury brand holders. Search engines maintain the value of their page views (and hence the pricing they can charge for advertising) by providing consistently high quality search results for their users. Thus, all major search engines have active anti-abuse teams that try to reduce the amount of search spam appearing in their results. When an SEO campaign is detected, search engines attempt to disrupt the campaign by either demoting their doorway pages in search results or even removing those pages from the search index entirely. Brand holders have a far less privileged technical position and they are neither able to analyze the Web at scale nor directly influence search results. However, as brand

and trademark holders they have unique legal powers that allow them to target particular pieces of infrastructure from SEO campaigns. These two techniques, search and seizure, represent the de facto standard methods of intervention against SEO campaigns, with pressure applied at different strata in the business model.

Search Engine

In addition to allowing the search ranking algorithm to demote doorways performing black hat SEO, search engines commonly have special handling for certain classes of malicious content. For example, starting in 2008, Google's Safe Browsing service (GSB) has detected and blacklisted sites leading to malware or phishing sites with the aim of preventing users from being defrauded through search. GSB labels search results leading to malware or phishing pages as *malicious*, appends the subtitle "This site may harm your computer" to the result, and prevents the user from visiting the site directly by loading an interstitial page rather than the page linked to by the result.

In 2010, Google instituted a similar effort to detect compromised Web sites and label them as *hacked* by similarly appending the subtitle "This site may be hacked" in the result [18]. The motivation is to curb the ability of compromised sites to reach unsuspecting users, while simultaneously creating an incentive for innocent site owners to discover their site has been compromised and clean it. In principle, this notification could undermine black hat SEO since users may be wary to click on links with a warning label.

However, there are important differences between these two seemingly similar efforts, which more likely reflect policy decisions rather than technical limitations. First, contrary to malicious search results, users can still click through hacked search results without an interstitial page. Second, typically only the root of a site is labeled as hacked; e.g., while http://⟨anonymized⟩ may be labeled as a hacked site, http://

⟨anonymized⟩/customize.php will not. Unfortunately, often only the non-root search results are compromised and redirect users, while the root search result is clean. In Section 5.4.2 we examine the implications and limitations of these policy decisions on search interventions against SEO campaigns.

Seizure

As the name suggests, seizures reflect the use of a legal process to obtain control of an infringing site (typically by seizing their domain name, but occasionally by seizing control of servers themselves) and either shut it down or, more commonly, replace it with a seizure notification page. In the context of counterfeit luxury, seizures prevent users from visiting seized domains, thereby hindering the store's ability to monetize traffic. Although we have witnessed brand holders performing seizures directly, typically they contract with third party legal counsel or with companies who specialize in brand protection, such as MarkMonitor [35], OpSec Security [44] and Safenames [48], to police their brand.

However, there are significant asymmetries in this approach. For example, a new domain can be purchased for a few dollars, but the cost to serve a legal process to seize it can cost 50–100 times more. Similarly, while a new domain name can be allocated within a few minutes and effectively SEO'ed in 24 hours (from Chapter 4), a seizure first requires finding the site, filing a legal claim and then waiting (from days to weeks) for the docket to be picked up by the federal judge to whom the case has been assigned. Presumably to amortize these costs, a manual review of court documents shows that domain name seizures commonly occur in bulk (hundreds or thousands at a time) and are not performed on a reactive basis. Finally, it is worth noting that doorway sites based on compromised Web servers present their own challenges since seizing the domain of an innocent third party can carry liability. Thus, while brands sometimes seize doorway

Table 5.1. A breakdown of the verticals monitored highlighting the number of poisoned search results, doorways, stores, and campaigns identified throughout the course of the study. Note that the KEY campaign targeted all verticals except those with an '*'.

Vertical	# PSRs	# Doorways	# Stores	# Campaigns
Abercrombie	117,319	2,059	786	35
Adidas	102,694	1,275	462	22
Beats by Dre	342,674	2,425	506	16
Clarisonic	10,726	243	148	6
Ed Hardy*	99,167	1,828	648	31
Golf	11,257	679	318	20
Isabel Marant	153,927	2,356	1,150	35
Louis Vuitton*	523,368	5,462	1,246	34
Moncler	454,671	3,566	912	38
Nike	180,953	3,521	1,141	32
Ralph Lauren	74,893	1,276	648	27
Sunglasses	93,928	3,585	1,269	34
Tiffany	37,054	1,015	432	22
Uggs*	405,518	4,966	1,015	39
Watches	109,016	3,615	1,470	35
Woolrich	55,879	1,924	888	38
Total	2,773,044	27,008	7,484	52

pages, it is more common for them to target the storefront advertised. Section 5.4.3 explores these asymmetries in greater depth.

5.3 Data Sets

The basis of our study relies upon extensive crawls of Google search results to discover poisoned search results that lead to counterfeit storefront sites. We then use a combination of manual labeling and supervised learning to map storefront sites into the different SEO campaigns that promote them. On a subset of storefront sites, we also use a combination of test orders and actual purchases to reveal information about customer order volume and payment processing. Finally, we crawl the site of a supplier to provide insight into the scale of order fulfillment and high-level customer demographics. This section describes each of these efforts in detail.

5.3.1 Google Search Results

Our primary data set comes from daily crawls of Google search results using a system that we previously developed for detecting search cloaking as described in Chapter 3. Each day we issue queries to Google using search terms targeted by counterfeit sites, crawl the sites listed in the search results, and identify sites using cloaking as depicted in Figure 5.1. We repeat this process for five months from November 13, 2013 through July 15, 2014.

In the rest of this section we define the notion of counterfeit luxury *verticals* for organizing search queries, and describe our methodology for selecting the search terms that comprise the verticals, the implementation of our crawlers and the information they collect, and our heuristics for detecting counterfeit stores in poisoned search results.

Note that we search exclusively using Google for a couple of reasons. In prior work we found that Google is the most heavily targeted search engine by attackers performing search poisoning and black hat SEO. Furthermore, Google is the leading search engine for the United States and many European countries, the preeminent markets receiving counterfeit products (based on shipping data from a large supplier as discussed in Section 5.3.5).

Search Terms

Any work measuring search results is biased towards the search terms selected because the selected terms represent just a subset of the entire search index. In our study, we monitor search results for counterfeit luxury *verticals*, a set of search terms centered around a single brand (e.g., Ralph Lauren) or a category composite of several brands (e.g., Sunglasses is a composite of Oakley, Ray-Ban, Christian Dior, etc.). For our study, each vertical consists of a static set of 100 representative terms that we determined were targeted by SEO campaigns.

Due to the early prominence of the KEY campaign, a large SEO botnet responsible for most of the PSRs manually observed in Sept-ember 2013, we initially compiled terms for each of the 13 verticals it targeted as listed in Table 5.1. Similarly, we followed the KEY campaign's approach in determining whether to center a verticals' terms around a single brand or a composite. To select these terms we extracted keywords from the URLs of the doorway pages of the KEY campaign. For a given vertical, we manually queried Google to find ten KEY doorways redirecting to the same store selling counterfeit merchandise (related to the vertical). Then we issued site queries (e.g., "site:doorway.com") for each doorway to collect all search results originating from the doorway. And for each search result we extracted search terms from the URL path (e.g., "cheap beats by dre" from http://doorway.com/?key=cheap+beats+by+dre) to assemble a large collection of terms. We then randomly selected 100 unique terms as a representative set for each vertical.

To extend the scope of our study to other campaigns, we included three additional verticals that we saw counterfeiters targeting: Ed Hardy, Louis Vuitton, and Uggs. Since the KEY campaign does not target these brands, we adopted a different approach in selecting search terms by using Google Suggest, a keyword autocomplete service. We first fetched suggestions for a targeted brand (e.g., "Louis Vuitton wallet"). Then we recursively fetched suggestions for the suggestions. In addition, we fetched suggestions for the concatenation of a commonly used adjective (e.g., cheap, new, online, outlet, sale or store) and the brand name to form search strings (e.g., "cheap Louis Vuitton"). From the combined set of these various search strings, we randomly selected 100 unique strings as our search set for each vertical.

To evaluate any bias introduced from these two different approaches, we take the ten original KEY verticals that are not composites, generate alternate search terms using the Google Suggest approach, and run the crawlers using those alternate terms for one day on April 25, 2014. Among the ten verticals, we find four out of a thousand total

terms overlap. Additionally, when comparing the percentange of PSRs detected after crawling, for both classified and unknown, and the distribution of PSRs associated to specific campaigns, we find no significant difference between results from the original and alternate terms over the same time range. Despite using two different approaches for selecting search terms, in the end we find the same campaigns poisoning search results. This overlap highlights both the pervasiveness of these campaigns and the representativeness of terms selected in spite of the KEY campaign's early influence on our methodology.

Crawling Search Results

For each search term, we query Google daily for the top 100 search results. Then, for each search result, we crawl each page link using an updated version of the Dagger cloaking detection system from Chapter 3. Dagger uses heuristics to detect cloaking by examining semantic differences between versions of the same page fetched first as a user and then as a search engine crawler (distinguished by the User-Agent field in the HTTP request).

A previous limitation of Dagger was that it did not render the page and, as a consequence, did not follow JavaScript (JS) redirects. Thus, we extended Dagger by rendering each cloaked search result detected using HtmlUnit [22], essentially a headless browser complete with a JavaScript interpreter. (Since rendering a page is an expensive operation, we only render pages we detect as cloaked.)

To detect iframe cloaking (Section 5.2.1), we implemented a second crawler, VanGogh. VanGogh also uses HtmlUnit to render pages. To detect iframe cloaking, it identifies any iframes attempting to occupy the entire page visually (hiding the original content). Specifically, we classify pages as using iframe cloaking if they load iframes where the height and width attributes are both either set to 100% or larger than 800 pixels.

And again, due to the high overhead of rendering pages, we only crawl a subset of search results using VanGogh. In particular, for each measurement we crawl at most three randomly selected pages from the same doorway domain to reduce the crawling workload. We further trim the workload by not crawling domains previously seen and not detected as poisoned by either VanGogh or Dagger. This approach has proved reasonable due to the low daily churn in search results for each vertical (on average 1.84% newly seen domains are found in search results each day).

Store Detection

Ultimately we want to identify counterfeit luxury storefronts advertised through PSRs. We detect stores by applying two heuristics to the set of PSRs discovered from crawling. First, we inspect cookies from each landing site (the page eventually loaded in a user's browser after redirection through the doorway page) to look for cookies commonly used by counterfeit luxury storefronts such as those related to payment processing (e.g., Realypay, Mallpayment), e-commerce (e.g., Zen Cart, Magento), and Web analytics (e.g., Ajstat, CNZZ). Second, we search for either of the substrings "cart" or "checkout" on the landing pages. If either of the heuristics succeed, we treat the landing site as a counterfeit luxury store advertised through search poisoning. Note that this approach identifies stores from the search results within a vertical irrespective of brand. For example, we may identify a counterfeit Christian Louboutin store within Louis Vuitton search results.

We validate our detection methodology by manually inspecting sampled search results from three popular verticals, Beats By Dre, Isabel, and Louis Vuitton. For each vertical, we randomly chose three search terms, and compared the search results for those terms from two measurements taken at least two months apart (e.g., one from November 23, 2013 and one from February 24, 2014). In total we examined 1.8K search results and detected 532 storefronts advertised using cloaked search results. Among these we found

no false positives (instances where a benign page is mistakenly labeled as a doorway to a storefront) and 21 (1.2%) false negatives (instances where a doorway to a storefront is not labeled). These results are reassuring because errors are likely skewed towards underrepresenting the number of storefronts.

5.3.2 Campaign Identification

Our targeted crawls of Google search results produce a large collection of doorway pages and counterfeit storefronts. We know that behind these thousands of doorways and storefronts lurk a much smaller number of distinct SEO campaigns, and the goal of our work is to understand the full ecosystem of campaigns operating in this counterfeit luxury market rather than focus on a singular campaign, e.g., the KEY campaign.

A brute-force approach to this understanding would require a domain expert to examine each Web page in our collection and use domain-specific heuristics to infer the SEO campaign behind it. The manual labeling of Web pages, however, is a time-consuming and laborious endeavor that does not scale well to the many thousands of examples in our collection. Instead we take a statistical approach, and the rest of this section describes an automated, data-driven method to identify the SEO campaigns behind individual doorway and storefront Web pages.

To build a statistical model, we need a data set of *labeled examples*. Though manual labeling is tedious, we created such a data set by identifying the SEO campaigns behind a small subset of 491 Web pages in our much larger collection of crawled results. From this small data set, we learned a classifier that mapped the remaining thousands of doorway and storefront Web pages to the 52 SEO campaigns for which we had manually labeled examples. The results of this analysis (discussed in later sections) provide a comprehensive understanding of the ecosystem of SEO campaigns in the counterfeit luxury market.

Our classifier makes its predictions by extracting textual features of HTML content and analyzing the statistics of these features that distinguish Web pages from different campaigns. The following subsections describe the classifier in more detail, focusing in particular on the individual stages of feature extraction, model estimation, and model validation.

Feature Extraction

The premise of our statistical approach is that doorway and storefront Web pages contain predictive signatures of the SEO campaigns behind them. Motivated by previous work [2], we looked for these signatures in their HTML source. We expect HTML-based features to be predictive in this domain for two reasons: first, because SEO campaigns use highly specialized strategies to manipulate the search rankings of doorways, and second, because campaigns often develop in-house templates for the large-scale deployment of online storefronts (e.g., customized templates for Zen Cart or Magento providing a certain look and feel).

To extract HTML features, we follow a conventional "bag-of-words" approach. In particular, we construct a dictionary of all terms that appear in the HTML source code, and for each Web page, we count the number of times that each term appears. In this way, each Web page is represented as a sparse, high-dimensional vector of feature counts. We implemented a custom bag-of-words feature extractor based on tag-attribute-value triplets [10] for the Web pages in our data set.

One might also expect to find predictive signatures of SEO campaigns in network-based features (e.g., IPv4 address blocks, ASes). However, we found that such features were ill-suited to differentiate SEO campaigns due to the growing popularity of shared hosting and reverse proxying infrastructure (e.g., CloudFlare). Therefore, after a brief period of experimentation, we did not pursue the use of such features.

Model Estimation

We learned linear models of classification from our data set of labeled examples. Specifically, we used the LIBLINEAR package [14] to learn L1-regularized models of logistic regression. The L1-regularization encourages sparse linear models in which the predictions of SEO campaigns are derived from only a handful of HTML features. Thus the resulting models are highly interpretable: for each campaign, the regularization serves to identify the most strongly characteristic HTML features from the tens of thousands of extracted ones.

We evaluated the predictive accuracy of the classifier by performing 10-fold cross-validation on the data set of labeled examples. The average accuracy on held-out data was 86.8% for multiway classification of Web pages into 52 different SEO campaigns. (Note that uniformly random predictions would have an accuracy of 1/52 = 1.9%.) Our model's high accuracy on held-on examples gave us confidence to classify the remaining (unlabeled) Web pages that we collected from poisoned search results.

Model Validation and Refinement

We used the above models, trained on a small subset of labeled Web pages, to infer the SEO campaigns behind the remaining unlabeled Web pages. To do so, we extracted HTML features from the unlabeled Web pages and used the classifiers to predict the most likely campaign behind each example. To validate these predictions, we manually inspected additional subsets of unlabeled examples. This step can be done efficiently by first validating the top-ranked predictions for each SEO campaign (as reflected by the probabilities that the logistic regressions attach to each prediction).

We briefly describe how we validated the classifier's predictions on unlabeled Web pages. Primarily we assume that distinct SEO campaigns are unlikely to share certain infrastructure such as SEO doorway pages and C&Cs, payment processing, and

customer support. We also consider less robust indicators such as unique templates, WHOIS registrant, image hosting, and Web traffic analytics (e.g., 51.la, cnzz.com, statcounter.com, etc.).

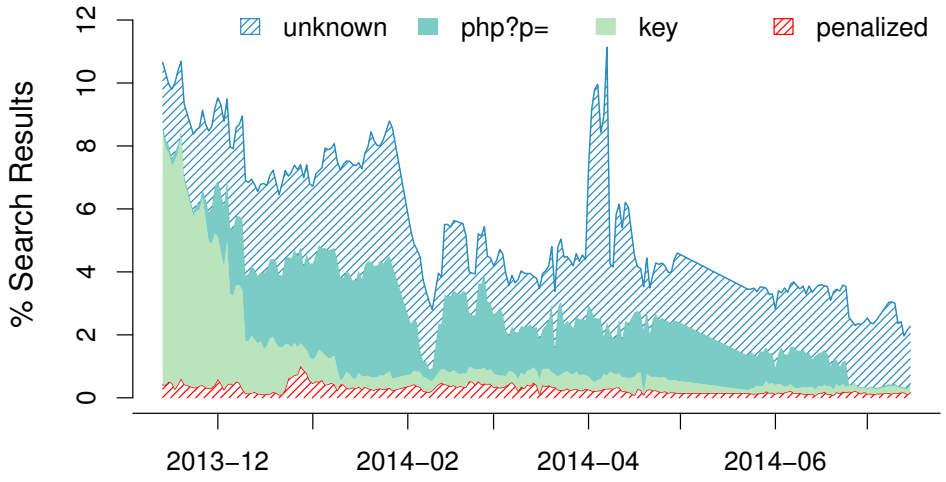
A final stage is to refine the model, using the manually verified predictions to expand the set of labeled Web pages, retraining the classifier on this expanded set, and repeating this process in rounds. With each iteration of this process we obtain a more accurate classifier and also one with greater coverage of distinct SEO campaigns. Though some manual labeling is unavoidable, this overall approach (of repeated human-machine interaction) is far more efficient than a brute-force expert analysis.

5.3.3 Purchases

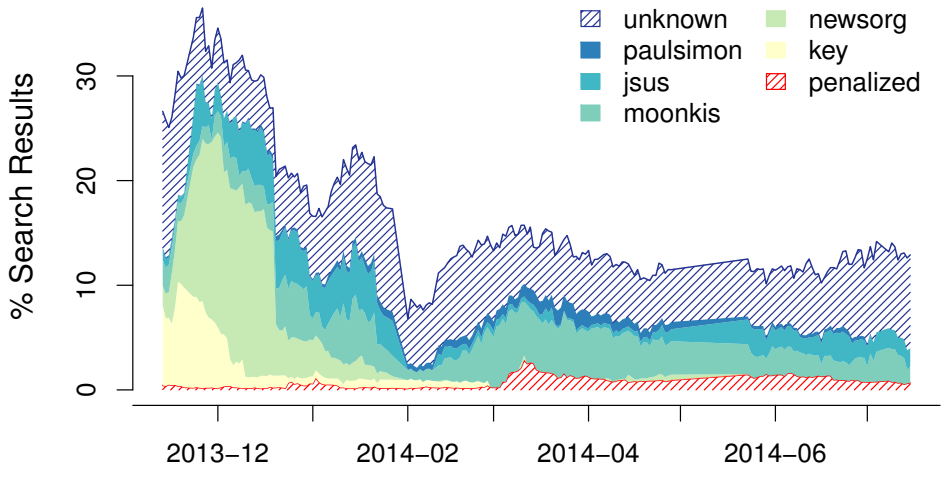
From previous work studying e-mail spam advertising illicit pharmaceutical and software storefronts [26, 31], we found that making orders on sites can shed light on normally opaque facets of underground businesses: order volume, payment processing, and order fulfillment. This information serves two important roles. First, it reveals the interplay between the various actors in the counterfeit luxury ecosystem (SEO campaigns, payment processors, and suppliers). Second, the estimated order volume serves as a vital metric in measuring the effectiveness of interventions (e.g., does labeling doorway search results as "hacked" lead to lower campaign order volume?). Similar to this prior work, we created test orders on counterfeit stores to estimate their order volume over time, and made actual purchases to reveal the payment processors used by these storefronts and the quality of the merchandise they sell.

Order Volume

We use the "purchase pair" technique [26] to estimate the order volume of individual stores over time. This technique exploits the fact that stores use monotonically

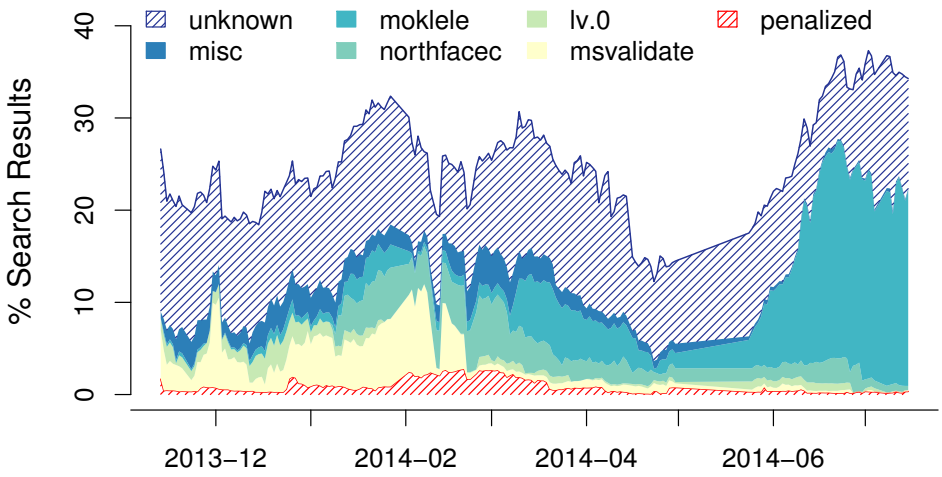


(a) Abercrombie

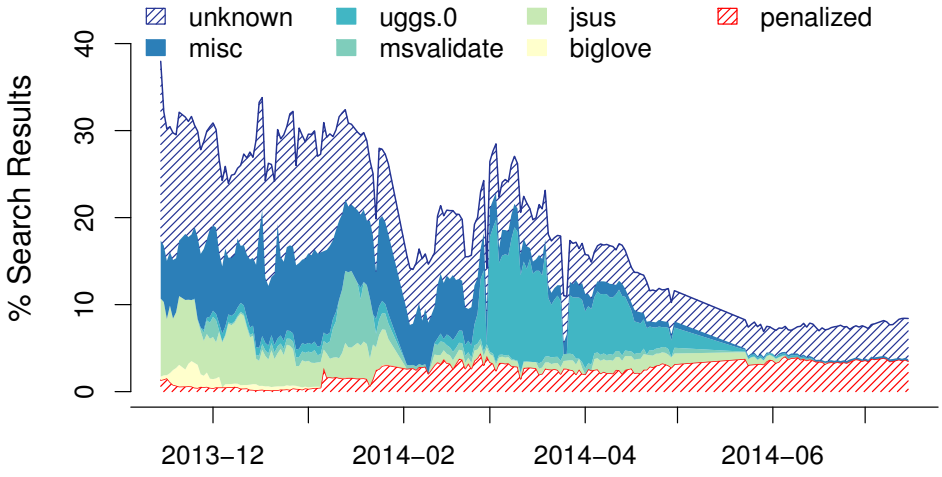


(b) Beats By Dre

Figure 5.2. Stacked area plots attributing PSRs to specific SEO campaigns within the labeled vertical. The red area represents the percentage penalized, either through search or seizure. The remainder of the areas represents active PSRs, where the filled areas are attributed to specific campaigns and the unfilled area is the remainder unclassified.



(c) Louis Vuitton



(d) Uggs

Figure 5.2. Stacked area plots attributing PSRs to specific SEO campaigns within the labeled vertical. The red area represents the percentage penalized, either through search or seizure. The remainder of the areas represents active PSRs, where the filled areas are attributed to specific campaigns and the unfilled area is the remainder unclassified. (Continued)

increasing order numbers, where the difference between order numbers represents the total number of orders created over the time delta between orders.

Note that stores give users an order number before processing their credit cards, and users still have an opportunity to back out. Therefore this metric represents an upper bound on orders placed and overestimates the absolute number of orders at a given store. However, it is still useful to quantify the rate at which orders are created, as well as the changes in order rate over time when correlated with interventions.

Using this technique, we created 1,408 orders from 290 stores touching 24 distinct campaigns and 13 verticals, between November 29, 2013 and July 15, 2014. We created 343 orders by hand and 1,065 orders using scripts. Operationally, for both manual and automated orders, we visit each store using via TOR and create orders at weekly intervals, and we limit orders to three per day per campaign to reduce the chance of being detected by the store or payment processor. We take the orders all the way to the payment processing page, which requires credit card details, before finally leaving the site. The order and customer information we provide are semantically consistent with real customers, but fictional and automatically generated [13].

Transactions

To shed light on payment processing and order fulfillment in the counterfeit luxury ecosystem, we successfully placed product orders from 16 unique stores covering 12 different campaigns. In total, we received 12 knock offs of low to medium quality, all shipped from China. From the bank identification numbers (BINs) in our transactions, we found that our purchases were processed through three banks (two in China, one in Korea). This concentration suggests payment processing is another viable area for interventions as in [37], but investigating such an intervention remains future work.

5.3.4 User Traffic

As described in Section 2.2, SEO campaigns poison search results to acquire user traffic that can then be monetized through scams — in this case, counterfeit luxury stores. The order volume data shows that counterfeit luxury stores do successfully convert user traffic into sales, and indirectly measures an SEO campaign's effectiveness in attracting traffic via PSRs.

For a small number of stores, we were also able to collect user traffic data that directly measures an SEO campaign's effectiveness in attracting customers to their stores. Specifically, we were able to periodically collect AWStats data for 647 storefronts in 12 campaigns. AWStats is a Web analytics tool [1] that uses a Web site's server logs to report aggregated visitor information (e.g., the number of visitors, visitor durations, visitor geolocations, referrers of visitors, etc.). From our crawled data, we discovered that these stores left their AWStats pages publicly accessible, and we were able to fetch visitor data for each store by visiting the publicly accessible default AWStats URL (e.g., http://⟨site⟩/awstats/awstats.pl?config=⟨site⟩).

5.3.5 Supply Side Shipments

To better understand the suppliers, customers, and operational relationship among storefronts and suppliers, we collected longitudinal shipment data from a supplier partnering with MSVALIDATE, one of largest SEO campaigns peddling counterfeit Louis Vuitton.

We discovered the supplier site from the packing slip of two of our purchases. Upon visiting the site, we noticed it contains a scrolling list of fulfilled orders and a mechanism to lookup shipping records for valid order numbers in bulk (20 orders at a time). Each record contains a timestamp and information regarding current location and delivery status.

Using this mechanism, we collected over 279K shipping records for nine months of orders placed through the supplier between July 5, 2013 and March 28, 2014. In summary, 256K orders successfully reached their destination, 4K were seized at the source (China), 15K were seized at the destination, and of the delivered, 1,319 were returned by the customer. From country data listed in the records, the three largest destinations are the United States, Japan, and Australia, with 90k, 57K, and 39K orders, respectively. If we combine these with the countries from Western Europe (41K), these regions account for over 81% of orders.

5.4 Results

In this section we use our crawler data to characterize the activities of SEO campaigns that use search to promote stores selling counterfeit luxury goods, and we further use our order data to study the effects of both search engine and domain seizure interventions on these SEO activities. In short, we find instances where both can have the desired effect of disrupting counterfeit sites, but they need to be far more reactive in time and comprehensive in coverage to undermine the entire ecosystem of SEO campaigns exploiting search engines for customers.

5.4.1 Ecosystem

We start with classifying poisoned search results into campaigns, how those campaigns target verticals, and what the PSRs reveal about the operations of the campaigns.

In terms of raw data, we crawled search results for eight months from November 13, 2013 through July 15, 2014 and detected 2.7M PSRs, across all verticals, using 27K doorways from unique domains and sending users to 7,404 different stores selling counterfeit luxury merchandise. Applying the classifier described in Section 5.3.2 to this data, we identified 52 distinct SEO campaigns that account for 828 stores, 11K doorway

domains, and 1.6M PSRs. Table 5.2 lists the campaigns using a name we derived from a pattern in their URLs, the domain names used for their C&C, or some other telltale aspect of their operation. For each campaign, the table lists the number of doorway domains, storefronts, and brands targeted. Note that, although we ascribed more than half (58%) of all PSRs to their respective campaigns, these PSRs only account for 11% of all stores. This disparity suggests that the ecosystem has a skewed distribution where a handful of large campaigns account for the majority share of PSRs that redirect users to a concentrated set of storefronts.

From the perspective of brands, we attributed 16% to 69% of PSRs in each vertical to known campaigns. Figure 5.2 visualizes our classification results for four verticals: Abercrombie (64.2% of PSRs classified to campaigns), Beats By Dre (62.2%), Louis Vuitton (66%), and Uggs (58%). We chose these verticals for their diversity in merchandise, campaigns, and search term selection meth-odology. For each vertical, the filled areas in the stacked area plots show the fraction of search results poisoned by the major campaigns targeting the vertical; note that the "misc" label collapses multiple campaigns into a single category to reduce clutter.

Each graph presents the PSRs detected, classified, and penalized over time at the granularity of a day. For example, in Figure 5.2b on December 1, 2013, 34.6% of search results for the Beats By Dre vertical were poisoned. Of these PSRs, 85.3% redirect users to counterfeit stores operated by five campaigns: KEY (16.8%), NEWSORG (53.8%), MOONKIS (5.8%), JSUS (8.0%), and PAULSIMON (0.3%). The remaining 14.7% PSRs redirect users to counterfeit stores we have yet to classify. The bottom shaded area shows that just 0.6% are penalized either through Google labeling the search result as "hacked" (Section 5.4.2) or a brand has seized the storefront domain name (Section 5.4.3).

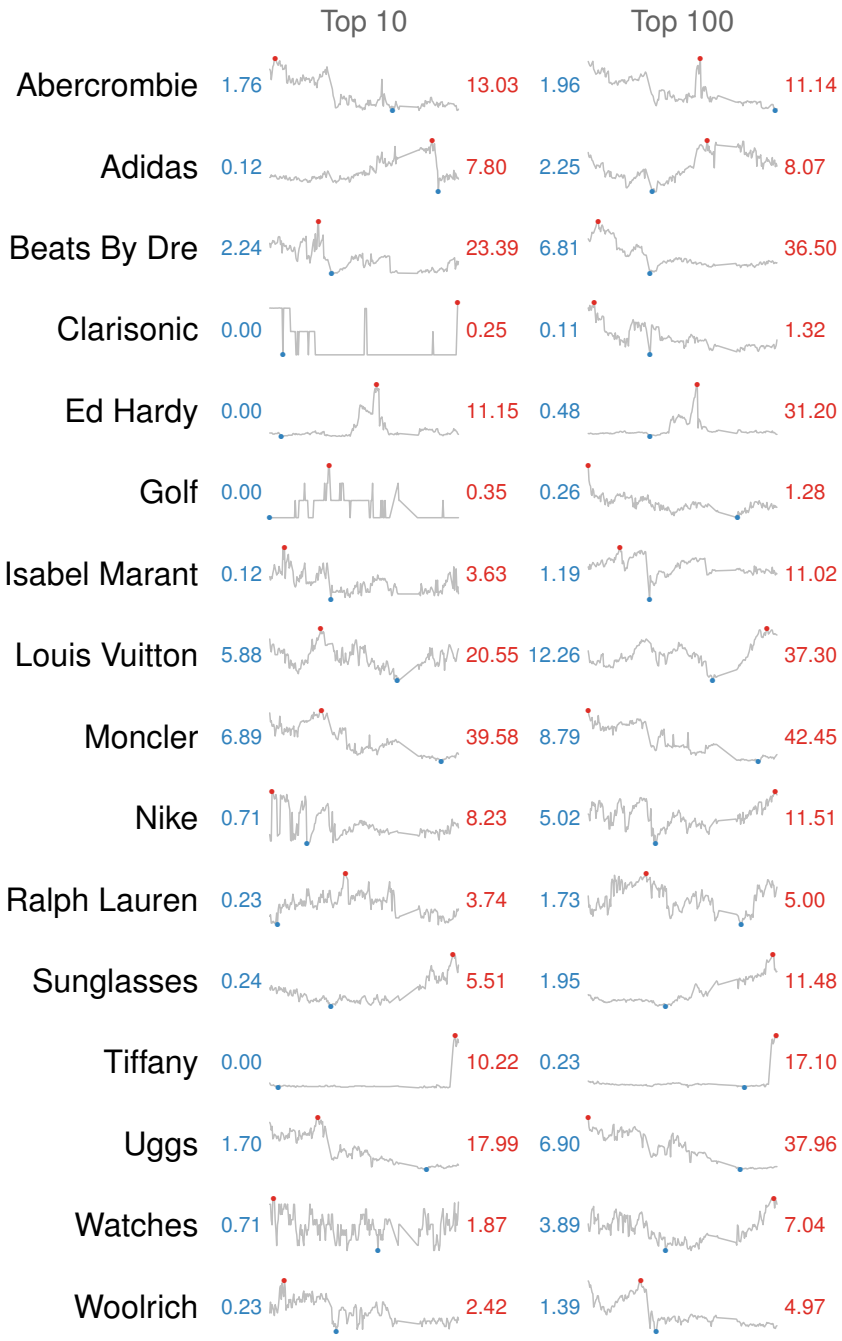


Figure 5.3. Percentage of search results poisoned for each brand vertical, shown as sparklines. Each sparkline is a daily time series showing relative values over five months. The left number is the minimum value across time and the right is the maximum (also shown as dots on the line).

Table 5.2. Classified campaigns along with # doorways seen redirecting on behalf of a specific campaign, # stores monetizing traffic from the campaign, # brands whose trademarks are abused by the campaign, and # days of peak poisoning duration, for campaigns with 25+ doorways.

Campaign	# Doorways	# Stores	# Brands	Peak
171760	30	14	7	44
ADFLYID	100	18	4	66
BIGLOVE	767	92	30	92
BITLY	190	40	15	89
CAMPAIGN.02	26	4	3	61
CAMPAIGN.10	94	18	5	99
CAMPAIGN.12	118	5	1	59
CAMPAIGN.14	39	8	2	67
CAMPAIGN.15	364	10	10	8
CAMPAIGN.17	61	8	3	44
CHANEL.1	50	10	4	24
G2GMART	916	28	3	53
HACKEDLIVEZILLA	43	49	9	56
IFRAMEINJS	200	2	1	39
JAROKRAFKA	266	55	3	87
JSUS	439	59	27	68
KEY	1,980	97	28	65
LIVEZILLA	420	33	16	70
LV.0	42	3	1	62
LV.1	270	12	9	90
M10	581	35	8	30
MOKLELE	982	15	4	36
MOONKIS	95	7	4	99
MSVALIDATE	530	98	6	52
NEWSORG	926	7	5	24
NORTHFACEC	432	2	1	60
NYY	29	14	5	40
PAGERAND	122	7	4	43
PARTNER	62	9	5	33
PAULSIMON	328	33	12	128
PHP?P=	255	55	24	96
ROBERTPENNER	56	7	12	50
SCHEMA.ORG	46	17	7	54
SNOWFLASH	271	14	1	48
STYLESHEET	222	9	6	63
TIFFANY.0	26	1	1	4
UGGS.0	428	6	5	30
VERA	155	38	12	156

Verticals

Figure 5.3 shows the percentage of search results that were poisoned for each vertical as pairs of sparklines. Each sparkline is a time series showing relative values over the five-month time span of the study at the granularity of days. The left number is the minimum value across time and the right is the maximum (also shown as dots on the line). Each column of lines shows the percentage of PSRs among top 10 search results (left) or top 100 (right). For example, in the Abercrombie vertical in the top left, at most 13% of the top 10 search results in the vertical were poisoned, while at least 2% were poisoned. The sparkline shows that the first three months were closer to the 13%, while the latter five months were much lower.

Overall, heavily targeted verticals are particularly vulnerable to poisoned search results. In 13 out of 16 verticals, about 5% of search results are poisoned at some point in time. But for the five verticals most vulnerable to search poisoning, at different points in time 31–42% of the top 100 search results in those verticals were poisoned. Also, as expected, it is easier in general to poison search results from outside the top 10; Beats By Dre, for instance, had at most 23% of its top 10 results poisoned while at one point 37% of its top 100 results were poisoned.

Brands face multiple "adversaries". Whether targeted by many campaigns (14 and 17 for Louis Vuitton and Uggs, respectively) or just a few (three and six for Abercrombie and Beats By Dre), all verticals are targeted by multiple campaigns all competing to SEO their doorway pages into search results to lure customers for their counterfeit goods.

SEO Campaigns

SEO campaigns employ considerable infrastructure to maintain their businesses. As shown in Table 5.2, SEO campaigns use hundreds to thousands of doorway sites to redirect users to dozens of storefronts (similar in scale to other abusive SEO botnets

described in Chapter 4). Interestingly, we do not find a strong correlation between the number of doorways and the campaign's efficacy in poisoning search results. For example, as shown in Figure 5.2b, MOONKIS poisoned search results for Beats By Dre from the start of 2014 onwards with 95 doorways, while two larger campaigns, JSUS and NEWSORG, used 439 and 926 doorways, respectively, in the same time period.

The operators of the campaigns successfully SEO their doorways in concentrated time periods. Although we observe campaigns poisoning search results for multiple months, their SEO effectiveness varies over time as exemplified by the campaigns targeting Beats By Dre in Figure 5.2b. To capture this notion of bursty SEO behavior, we compute a "peak range" for each campaign defined as the shortest contiguous time span that includes 60% or more of all PSRs from the campaign. For example, NEWSORG's peak range lasts 24 days from November 23 to December 17, 2013, with a daily average of 1676 PSRs during this span. Table 5.2 summarizes the peak duration in terms of number of days for each campaign. Using this metric, we find campaigns run at their peak for 51.3 days on average.

The campaign operators also run a diversified business that gives them flexibility in the face of disruption. A single campaign, for example, will use its doorways to poison search results from multiple verticals simultaneously. For instance, the MSVALIDATE and BIGLOVE campaigns both successfully poison search results for Louis Vuitton (Figure 5.3c) and Uggs (Figure 5.3d). As a result, campaigns possess multiple revenue streams, giving them flexibility in the event a setback disrupts one revenue stream (e.g., domain seizures from one brand, problems with a supplier for Beats By Dre headphones, etc.). The campaign can adjust and continue monetizing traffic by simply reallocating resources towards stores selling counterfeit merchandise from other verticals.

Moreover, campaigns often operate multiple storefronts targeting the same vertical and selling the same merchandise. Sometimes the goal is to localize for a market, such

as a Japanese Uggs storefront catering to Japanese consumers. More often, though, these redundant stores can serve as backups in the event of interventions, which we explore further in Section 5.4.3.

5.4.2 Search Engine Interventions

Since poisoned search results manipulate and degrade user experience, search engines have an incentive to identify and penalize PSRs used by the SEO campaigns that lead users to counterfeit sites. Two options available to search engines for reacting to PSRs are to demote them in search rank, and to add warning labels to search results to alert users before clicking through.

Search Result Demotion

Figure 5.4 shows the prevalence of poisoned search results for four SEO campaigns over time, and the corresponding order activity at storefront sites gleaned from creating test orders as described in Section 5.3.3. The bottom two rows of graphs show the number of PSRs per day for each SEO campaign: the lowest row focuses on PSRs in just the top 10 search results, and the row above focuses on the full top 100 search results. The dark portion of the bars at the bottoms of the graphs corresponds to PSRs labeled by Google as “hacked”.

The top two rows show the results of sampling order numbers from a handful of representative stores promoted by each campaign; the stores that are both visible in PSRs and have high order activity relative to other stores from within the same campaign. The top “Volume” row shows the actual samples over time for the handful of representative stores and reflects the combined cumulative volume of order numbers created (recall that these numbers are an upper bound of actual orders made by customers). As another way of looking at the same data, the lower “Rate” row shows the order data as a histogram:

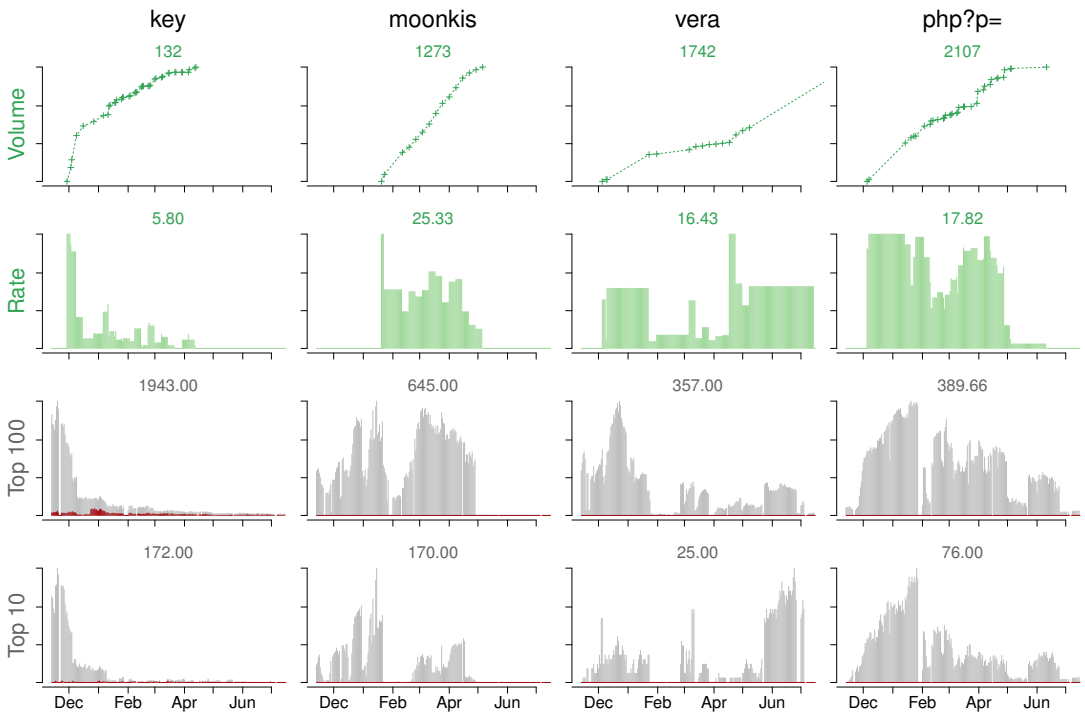


Figure 5.4. Correlation between a store's visibility in PSRs and order activity for four SEO campaigns. Each column of graphs is associated with an SEO campaign. Bottom two rows of graphs depict the prevalence of PSRs among the top 100 and top 10 search results, respectively. Top two rows reveal cumulative changes in sampled order numbers, as well as histograms binning order number changes into extrapolated daily rates, respectively.

we bin the number of estimated orders per week, interpolating in regions where we lack samples. The number at the top of each graph is the maximum value across the time series.

In all four campaigns, we see correlation between PSR prevalence and order activity, which suggests that search penalizations can be effective. This is most evidedent with the KEY campaign. The rate of orders (slope of the line in the "volume" graph) decreases in mid-December, soon after its PSR activity drops precipitously. We do not know the actual cause of the drop in PSRs: whether the KEY campaign stopped actively performing SEO on its doorway sites, Google aggressively demoted its doorways in

search result rank, or the "hacked" warning added to its PSRs dissuaded users from clicking on search results. However, it appears that the penalization pressure from Google—demoting most of the PSRs out of the top 100 and labeling half of the remaining as hacked—did have an effect. From attempts to actually purchase items, the stores promoted by the KEY campaign stopped processing orders after the drop in PSRs.

Penalizing PSRs by demoting them in search rank follows the conventional SEO wisdom that highly-ranked results are by far the most valuable. On this topic, the bottom two rows of Figure 5.4 show the prevalence of PSRs in the top 10 and top 100 results. For the most part, the shapes of both histograms are similar: campaigns are successful in SEOing poisoned search results throughout search page ranks, and it is difficult to conclude whether order volumes seen at stores are primarily due to the much smaller number in the top 10 or the much larger number across the top 100. One example, though, suggests that there is value in having PSRs across the full top 100. For the MOONKIS campaign, during most of March 2014 it had negligible PSRs in the top 10 but hundreds in the top 100. Nonetheless, order volumes seen at its stores remained high and steady. In this instance at least, search rank penalization would need to be even more aggressive to demote the PSRs from the top 100.

Warning Labels

Google uses the "hacked" label on search results to warn users about suspicious sites. This form of intervention faces two key challenges—coverage and reaction time—and, based upon our cra-wling results, overall appears to be ineffective for this type of abusive SEO activity.

Although most doorways are hacked sites, Google only penalizes 2.5% of the PSRs we crawled with a "hacked" label. From the perspective of brands, Figure 5.2 showed that penalized PSRs labeled with the "hacked" warning were a small fraction

Table 5.3. Summary of domain seizures initiated by brand holders from Feb. 2012 – Jul. 2014, aggregating the following per seizing entity: number of court cases initiating seizures (# Cases), number of brands protecting their trademarks through such cases (# Brands), and total number of store domains seized as reported in cases (# Seized). For overlap with the eight months of our crawled data set (Nov. 2013 – Jul. 2014), we also list the subset of store domains seized and directly observed in our crawled PSRs (# Stores), the number of those stores we classified into campaigns (# Classified Stores), and the number of SEO campaigns affected by seizures (# Campaigns).

	Green, Burns, & Crain	**SMGPA**
# Cases	69	47
# Brands	17	11
# Seized	31,819	8,056
# Stores	214	76
# Classified Stores	40	20
# Campaigns	17	12

of all PSRs at any point in time for four large brand verticals. From the perspective of campaigns, Figure 5.4 shows a similar result: except for the KEY campaign, both the absolute number and fraction of penalized PSRs are quite small.

One issue that undermines coverage is that Google only labels the root of a Web site as "hacked", and does not label search results that link to sub-pages within the same root domain. In the PSR data set, we found 68,193 "hacked" search results. When counting the number of PSRs that share the same root as a penalized site, Google could have labeled 102,104 search results (an additional 49%).

A second challenge is reaction time. A key metric of any reactive intervention is the time delay between when a campaign starts SEOing a doorway and when the search provider detects and penalizes the doorway with a label. This delta represents the window of opportunity for an SEO campaign to monetize traffic obtained through PSRs without any warnings to users.

For doorways penalized with a label, campaigns have multiple weeks in which to monetize traffic through PSRs. Of the 1,282 "hacked" doorways in the PSRs data,

588 doorways were already labeled when we first saw them and we cannot determine when they were first labeled. The remaining 694 have lifetimes between 13–32 days on average until Google labeled them as “hacked”.

Note that the variance in the lifetime is due to the difficulty in determining exactly when Google penalizes a site. Using crawled search results, we know when we last saw a doorway prior to the penalty and when we first saw a doorway after the penalty. However we cannot always determine when the penalty occurred because it may be the case the doorway does not appear in our results for an extended period of time. As a result, we present two numbers, the smaller of which is the lifetime ending when we last saw the doorway actively redirecting, while the larger number is the lifetime when we first observed the labeling.

User Traffic

The correlation between search result visibility and order volume, observed in Figure 5.4, is an indirect measure of the ability of campaigns to attract and convert traffic via PSRs. Combining the AWStats data described in Section 5.3.4 with the crawled data and test purchases, we are able to examine this relationship in greater detail with a case study of a counterfeit Chanel store run by the BIGLOVE campaign that rotates across three storefront domains over time (cocoviphandbags.com, cocovipbags.com, and cocolovebags.com).

As above, in Figure 5.5 we present both the prevalence of PSRs attributable to this store and the corresponding extrapolated order activity from June 10, 2014 to August 31, 2014. Using the AWStats data, in the bottom-most graph we also present the daily user traffic seen by the store in terms of the number of HTML pages fetched by users each day. We use color gradients in the PSRs and traffic graphs of Figure 5.5 to distinguish separate instances of coco*.com, where each instance represents a different domain name used

for the storefront. As seen by the change in gradients, the BIGLOVE campaign rotated domains for this storefront twice, at the end of June and the middle of August, updating its doorways found in PSRs to redirect to the new instances. We see similar changes in traffic coinciding with each of the domain name changes.

Although there is not sufficient evidence to discern the campaign's intent, one possibility is that these domain name changes are a proactive countermeasure against domain name seizures. As discussed in Section 5.2.2, luxury brand holders frequently seize domain names to curtail counterfeit sales. However, as we will show in Section 5.4.3, SEO campaigns are well aware of the ongoing seizures and oftentimes react within days of the initial seizure by simply redirecting to another domain. And being proactive ensures that there is no downtime: the first domain cocoviphandbags.com was seized on July 11, yet by that time the doorways were already redirecting users to the second domain cocovipbags.com.

Inspecting the detailed user traffic data from AWStats, we make rough estimates on conversion metrics from coco*.com that are consistent with those reported by marketers [7]. During the months of July and August 2014, coco*.com combined received 93,509 visits, 60% of which properly set the HTTP referrer header.[4] Extracting the referrers reveals the complete set of doorways supplying traffic for this store, and we find 83 out of 174 doorway domains (47.7%) were seen in our crawled PSRs data (recall that we limit the number of terms we search for a given vertical, and so it is not surprising that we do not capture all doorways). Examining user visits more closely, we find each visit generates 5.6 HTML page fetches on average. And when combining the traffic data with the order data from test purchases, we estimate this store had a 0.7% conversion rate, roughly a sale every 151 visits.

[4]The HTTP referrer header is not properly set in many situations, including transitioning from HTTPS to HTTP, visiting through an e-mail client, visiting through a proxy that strips the header, or simply typing the URL directly into the browser.

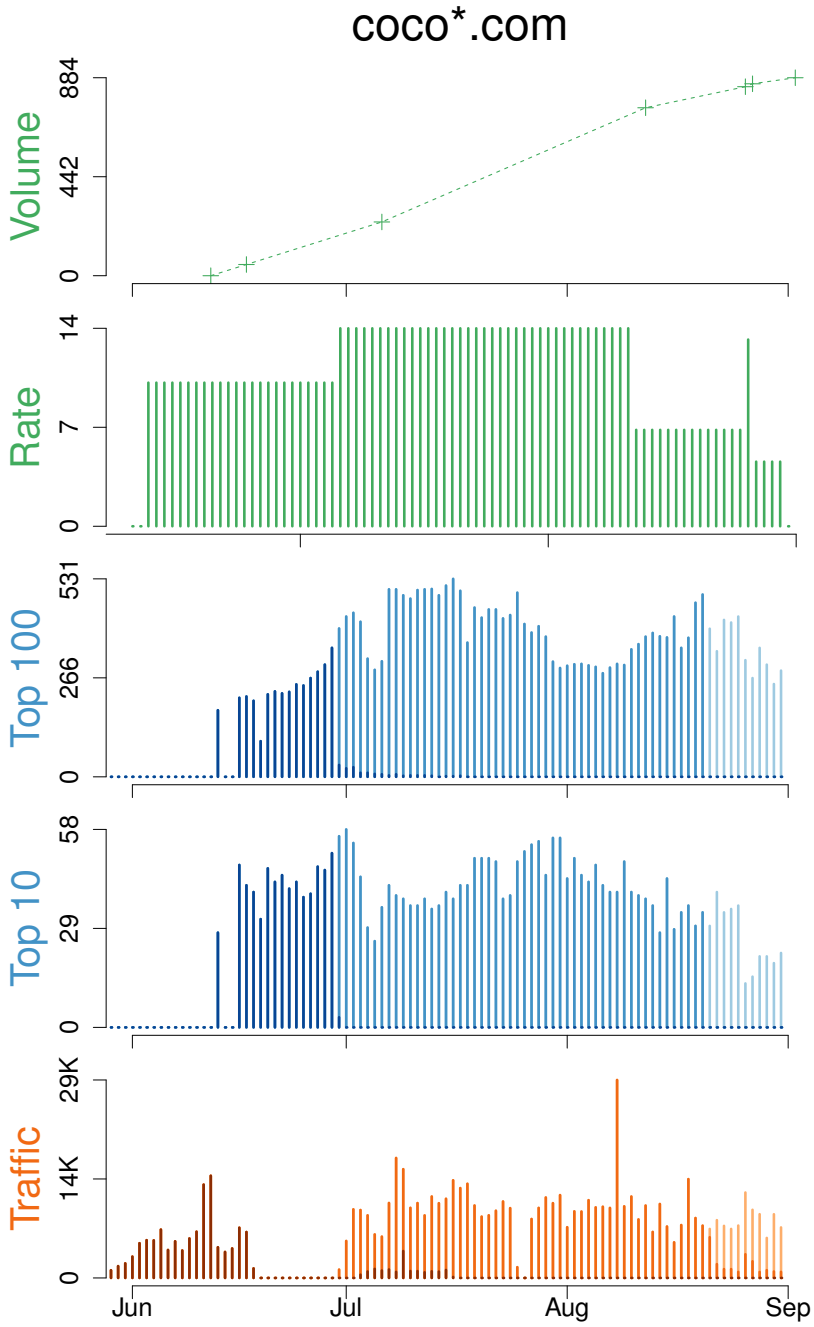


Figure 5.5. A detailed example of the correlation between a store's prominence in search results (Top 100, Top 10), the resulting user traffic seen by the store (Traffic), and the monetization of user traffic through orders (Volume, Rate), for a counterfeit Chanel store run by the BIGLOVE campaign from June – September 2014. Each color gradient in the PSRs and traffic graphs is associated with separate instances of coco*.com, where each instance used a different domain name.

5.4.3 Domain Seizure Interventions

As discussed in Section 5.2.2, brands have the most incentive for undermining online counterfeiters, and a highly visible intervention they can use is to seize the domains of counterfeit storefronts. With this intervention, brands use legal means to seize domain names of stores violating brand holder trademarks, thereby preventing users from visiting sites selling counterfeit merchandise.[5]

We use two sources of data for studying domain seizure by brand holders. The first is the set of PSRs from our crawled search data. Mechanistically, it is straightforward to determine whether a store is seized by checking whether the site redirects users to a serving notice provided by one of the third-party brand protection services (e.g., Greer, Burns & Crain [19], SMGPA [54]) or the brand holders themselves. The second is a set of seized domains listed in court documents embedded in the serving notice pages; these documents list all domains involved in a seizure, and enable us to obtain a broader view of domain seizure activity by brand holders spanning up to two years.

By extracting the brand holders and the timestamps from seizure notices, we can also infer how brands use brand protection services. For both GBC and SMGPA, we find brand holders initiate domain name seizures on a periodic basis, typically on the order of months between rounds of seizures. Although a handful of brands seized domains more frequently—Oakley issued 6 court cases at monthly intervals, Uggs issued 19 court cases at bi-weekly intervals, and Chanel issued 18 court cases also at bi-weekly intervals—they tend to be the exception.

To assess the completeness of observing domain seizures using PSRs, we compared the court cases seen in PSRs against ground truth we collected by enumerating all court orders from GBC, which are publicly available through their Web site. During

[5] Another option would be to seize doorway domains, but doing so presents two obstacles: there are two orders of magnitude more doorway domains than stores (Table 5.2), and the doorways are often compromised sites.

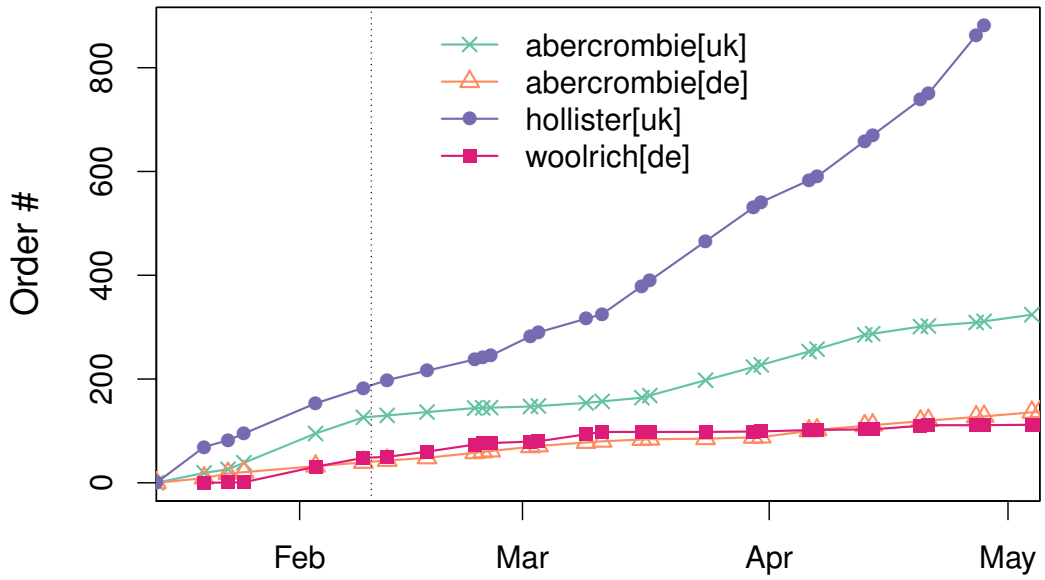


Figure 5.6. Order number samples over time in early 2014 for the PHP?P= campaign. Each curve corresponds to one of four international stores, where three sell Abercrombie (United Kingdom, Germany) while the remaining sells Woolrich (Italy).

the time frame of our study (November 2013 to July 2014), we observe 47 cases in PSRs out of the 50 total cases initiated by GBC (94%) during the same time frame. This overlap indicates that our crawled search data captures the bulk of seizure activity by brand holders.

Coverage

Brand holders have been aggressive in seizing storefront domains. Table 5.3 shows a breakdown of seized domains across brands, storefronts, and campaigns. From manually examining the court documents embedded in seizure notice pages, brands arranged to have almost 40,000 domains seized over two years. Specifically, while

representing 17 brand holders GBC seized 31K domains using 69 court orders from December 2012 to June 2014. Similarly, while representing 11 brand holders SMGPA seized 8K domains using 47 court orders from February 2012 to July 2014.

Seizing domains can disrupt online counterfeit stores. For example, Figure 5.6 shows order numbers over time for four stores promoted by the PHP?P= campaign. The domain for its Abercrombie UK store was seized on February 9, 2014 (vertical dotted line), and its rate of new order numbers declined immediately (it did not stop completely due to reaction by the SEO campaign, as discussed below in Section 5.4.3).

Despite their aggressiveness, though, brand holders must be far more aggressive when seizing domains. From the PSRs crawled, we directly observed 290 seizures over our eight-month period: 214 seized by GBC and 76 by SMGPA. Compared with the total number of storefronts we observed (7,484), though, these domain seizures represent just a small percentage (3.9%) of stores used by SEO campaigns. As a result, unless brands comprehensively seize domains, stores promoted by SEO campaigns remain unaffected and they continue to attract traffic and customers via search. Referring back to the example in Figure 5.6, the orders remained steady at other stores whose domains were not seized and the SEO campaign remained effective overall.

Reaction Time

Even if brands eventually seize all storefront domains, though, they take such a long time to seize domains, and attackers respond so quickly to having domains seized, that the current environment still strongly favors the attackers.

As with labeling sites as “hacked”, the time from when a storefront goes live and when a brand seizes the store domain represents the window of opportunity for a counterfeit store to monetize traffic. As in Section 5.4.2, we can compute the lifetime of seized stores to measure their earning potential before seizure. We define the lifetime of

seized stores as the delta between the first time that the storefront appears in PSRs to the time the domain was seized.

We find the average lifetime of seized stores lies between 58–68 days for GBC and 48–56 days for SMGPA. Again, due to the nature of our data collection, we can only observe seizures when crawling search results redirects our crawlers to seized stores. Therefore, we cannot determine exactly when a seizure takes place. Instead, we present two bounded numbers approximating the true lifetime. The smaller is the duration ending when we last saw the store actively poisoning search results, while the larger is the lifetime ending when we first definitively observed the seizure.[6]

However, even when brands seize storefront domains, the SEO campaigns possess backup domains in anticipation of such an intervention and can quickly react to continue to monetize traffic without significant interruption. Returning to the example in Figure 5.6, when the Abercrombie UK domain was seized, the PHP?P= campaign changed their doorways to forward to a new store domain within 24 hours. This domain agility represents a critical weakness of seizures: even though a store domain is seized, SEO campaigns can easily modify their doorways to redirect users to their backups rather than the seized domains.

Indeed, we found widespread evidence of attackers exploiting this weakness as a countermeasure to domain seizures. Specifically, of the 214 seized stores from GBC, 130 were redirected to new stores (59 of which were subsequently seized) and, among the 76 seized from SMGPA, 57 were redirected to new stores (22 of which were subsequently seized). These responses by the counterfeiters happened on average within 7 and 15 days of the initial seizure, respectively, for GBC and SMGPA. Such domain agility suggests the counterfeiters are well prepared for domain seizures, and as a result such interventions

[6]Given how quickly campaigns react to domain seizures, it is also possible for campaigns to redirect doorways to new store domains faster than our crawler can detect that the initial domain was seized.

are not likely to undermine their business.

5.5 Summary

Online business in counterfeit luxury goods is brisk: from the site of just one supplier, we saw over 250,000 successfully delivered orders in nine months. Such businesses prosper by poisoning search results for popular luxury goods to attract customers to their online storefronts; for heavily-targeted brands, a third of the top 100 search results are so poisoned for months at a time.

In this paper we presented techniques for detecting poisoned search results that lead users to counterfeit stores, and a classification approach for mapping the Web sites of these stores into distinct SEO campaigns that promote these sites. From eight months of crawled search results for 16 brand verticals, we detected 2.7 million PSRs using 27 thousand doorway pages that redirect users to 7,484 storefronts, and classified over half of the PSRs into 52 distinct SEO campaigns. Simultaneously, we created test orders on stores to sample their order number sequence space to estimate their order volume over time.

Finally, we used our crawler and order data to study the effects of both search engine and domain seizure interventions on these abusive SEO activities. Although we find instances where both can have the desired effect of disrupting counterfeit sales activity, overall neither are currently employed with the level of coverage or responsiveness necessary to be broadly effective against the actors in this market. Search engines and brand holders should take into account that these activities are organized as business campaigns, that effective interventions should target their infrastructure at the granularity of these campaigns, and that they are being targeted by dozens of campaigns. Otherwise campaigns have shown great agility in adapting to partial intervention, and in filling in gaps left by the disappearance of other campaigns. We believe that the measurement and

classification techniques we describe in this paper for identifying campaigns and their infrastructure could provide the improved targeting required for more robust intervention.

Chapter 5, in part, is a reprint of the material as it appears in Proceedings of the ACM Internet Measurement Conference 2014. Wang, David Y.; Der, Matthew; Karami, Mohammad; Saul, Lawrence; McCoy, Damon; Savage, Stefan; Voelker, Geoffrey M. The dissertation author was the primary investigator and author of this paper.

Chapter 6

Conclusion

Due to the continued profitability of abusive advertising and inadequate defenses, search result poisoning remains a highly pervasive form of search engine abuse used to acquire user traffic through deceptive means. In the counterfeit luxury goods ecosystem alone, we identified over 52 SEO campaigns responsible for 1.7M PSRs over an eight-month span. This is not the exception; many diverse ecosystems exist, including those funded by counterfeit pharmaceuticals, counterfeit software, malware distribution, and so forth. And in fact, SEO campaigns are alarmingly successful in these markets as well. For example, over a 10-day stretch, we find GR was the principal contributor of PSRs redirecting users to fake anti-virus from trending search results.

In this dissertation, we present a framework for understanding and addressing the root causes of search result poisoning. In support of this, we analyze search poisoning from three distinct perspectives ranging from lower-level mechanisms (PSRs, SEO botnets) to high-level operations (ecosystem). Starting from the point-of-view of PSRs, we implement Dagger, a crawler-based system to detect semantic cloaking in search results, and use the data collected as the basis for our study on modern cloaking. From this, we find attackers frequently use cloaking in poisoned search results to exploit search engines and acquire traffic surreptitiously. Then, by infiltrating the GR SEO botnet, we characterize both the composition of an influential SEO botnet (scale, churn) and the machinery

generating PSRs at massive scale. And by correlating contemporaneous data, we quantify GR's effectiveness in poisoning search results and the subsequent response it engenders from search engines. Lastly, we present a methodology to evaluate the effectiveness of interventions in disrupting SEO campaigns and perform an ecosystem-level analysis on the counterfeit luxury goods market. Focusing on the cat-and-mouse relationship between SEO campaigns and anti-counterfeiting efforts, we find current defenses are a step behind as they lack the necessary comprehensiveness and responsiveness to disrupt campaigns, thereby leading to an ecosystem that remains rife with abuse.

6.1 Future Directions

Search result poisoning remains a threat to multiple parties (e.g., search engines, users, brand holders, etc.). Due to the adversarial nature of security, we expect future work will focus heavily on studying attacker countermeasures to the methodologies and techniques presented here. However, the most exciting opportunities for future research perhaps lie in understanding other forms of search engine abuse (e.g., information manipulation services), which will likely require either new or modified working models and approaches.

6.1.1 Attacker Countermeasures

While studying search result poisoning over these four years, we have witnessed more than enough evidence that attackers take countermeasures against security researchers as a whole and even to our specific efforts. As an example, during these four years, we have observed a gradual evolution in the cloaking techniques attackers used, where each advancement requires greater sophistication to detect. Specifically, attackers originally used user-agent and referrer cloaking, which are easily detected by spoofing HTTP headers. Eventually, they switched to IP cloaking, which requires having access

to a specific IP address to retrieve a specific version of a page (e.g., a page destined for search crawlers). And recently, to further raise the bar, attackers use iFrame cloaking, thereby requiring a JavaScript interpreter.

This cat-and-mouse scenario is prevalent throughout computer security and, interestingly we experienced this directly with our own data collection efforts. For example, in response to our extensive crawling of search results, attackers have placed our UC San Diego IP address range on blacklists to try to prevent us from crawling their sites. Consequently, we started crawling out of proxies from a more diverse IP range.

This continual escalation of capabilities introduces multiple research opportunities. First, how will attackers adapt to more impactful interventions (e.g., payment intervention as suggested by McCoy et al. [37]), including the high-level interventions targeting SEO campaigns presented in this work? If we assume the attacker will continue the arms race until their operation is no longer profitable, then can we estimate when will we reach that point? Furthermore, we have shown an approach for disrupting one ecosystem; what will it take to generalize our results and disupt all ecosystems?

6.1.2 Beyond Abusive Advertising

Throughout this dissertation we exclusively focus on a specific instance of search engine abuse — the phenomenon of search result poisoning. Even though we find a variety of different scams (e.g., sale of illicit goods, malware distribution, etc.) fund the SEO campaigns behind search poisoning, all the campaigns studied essentially perform abusive advertising.

Nevertheless, attackers can abuse search engines in alternative ways, such as through *negative SEO* and *information censorship*. Rather than promoting a client's site through SEO, negative SEO uses black hat SEO on a client's competiton. Thus, the attacker potentially incurrs penalities on the competition by smearing their reputation, in

hopes that the client will now outrank their competition. In a related service, attackers censor information by using SEO to promote low quality content. This pollutes the search results returned, in effect hindering the user's ability to find relevant information.

Unlike in abusive advertising, where the goal is to acquire user traffic, these attacks are motivated by the prospect of information manipulation. By exploiting the search engine's role of connecting users with information, attackers have developed methods to promote a client by smearing the reputation of competitors and to censor information through polluting search results with noise. Due to the differences in motive behind abusive advertising and information manipulation, it is unlikley that the models and approaches from this work will translate. Thus, as a first step towards research in this space, we should establish the model of abuse and extent of the problem. From this, we can then craft prevention strategies.

6.2 Final Thoughts

Given the contributions from this dissertation and all that we have learned from similar related work studying online underground economies [6, 21, 31, 33, 37, 40], we are at an opportune time to apply the findings of our research in the real world. The methodologies and techniques described here have been tested and iteratively refined over the past four years. Admittedly there are several challenges to transfer research into production. Specifically, in this case scaling up the infrastructure to operate closer to Web-scale. Also, in the event our work gains widespread adoption, attackers will surely develop additional countermeasures. However, we believe that targeting SEO campaigns at the ecosystem-level will lead to meaningful impact. Because there are orders of magnitude fewer SEO campaigns than PSRs, interventions can concentrate more thoroughly on fewer instances. This along with accurately ascribing PSRs to their campaigns allows each effort to comprehensively erode the campaign's infrastructure. Furthermore, since

the framework to identify and target the principal SEO campaigns within an ecosystem already exists, the next logical step is to apply an intervetion operating at the ecosystem-level (e.g., payment), to see if it as successful as expected [31, 37].

Bibliography

[1] AWStats — Free log file analyzer for advanced statistics (GNU GPL). http://awstats.sourceforge.net/.

[2] Sushma Nagesh Bannur, Lawrence K. Saul, and Stefan Savage. Judging a site by its content: learning the textual, structural, an d visual features of malicious Web pages. In *Proceedings of the ACM Workshop on Artificial Intelligence and Security (AISEC)*, Chicago, IL, October 2011.

[3] John Bethencourt, Jason Franklin, and Mary Vernon. Mapping Internet Sensors with Probe Response Attacks. In *Proceedings of the 14th USENIX Security Symposium*, Baltimore, MD, July 2005.

[4] Andrei Z. Broder. On the Resemblance and Containment of Documents. In *Proceedings of the Compression and Complexity of Sequences (SEQUENCES'97)*, June 1997.

[5] Lee G. Caldwell. *The Fast Track to Profit*. Pearson Education, 2002.

[6] Neha Chachra, Damon McCoy, Stefan Savage, and Geoffrey M. Voelker. Empirically Characterizing Domain Abuse and the Revenue Impact of Blacklisting. In *Proceedings of the Workshop on the Economics of Information Security (WEIS)*, State College, PA., June 2014.

[7] Dave Chaffey. Ecommerce conversion rates. http://www.smartinsights.com/ecommerce/ecommerce-analytics/ecommerce-conversion-rates/, March 2014.

[8] Kumar Chellapilla and David Maxwell Chickering. Improving Cloaking Detection Using Search Query Popularity and Monetizability. In *Proceedings of the SIGIR Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, Seattle, WA, August 2006.

[9] Marco Cova, Corrado Leita, Olivier Thonnard, Angelos Keromytis, and Marc Dacier. An Analysis of Rogue AV Campaigns. In *Proceedings of the 13th International Symposium on Recent Advances in Intrusion Detection (RAID)*, Ottawa, Canada, September 2010.

[10] Matthew F. Der, Lawrence K. Saul, Stefan Savage, and Geoffrey M. Voelker. Knock It Off: Profiling the Online Storefronts of Counterfeit Merchandise. In *Proceedings of the ACM SIGKDD Conference*, New York, NY, August 2014.

[11] Amir Efrati. Google Penalizes Overstock for Search Tactics. http://online.wsj.com/article/SB10001424052748704520504576162753779521700.html, February 24, 2011.

[12] eMarketer. eMarkter Press Release: Google's Share of US Search Revenues is Still Growing. http://www.emarketer.com/PressRelease.aspx?R=1008258, March 2011.

[13] Generate a Random Name — Fake Name Generator. http://www.fakenamegenerator.com/.

[14] R. E. Fan, K. W. Chang, C. J. Hsieh, X. R. Wang, and C. J. Lin. LIBLINEAR: A Library for Large Linear Classification. http://www.csie.ntu.edu.tw/~cjlin/liblinear/.

[15] Danny Goodwin. Search Ad Spending Could Hit $19.51 Billion in 2012. http://searchenginewatch.com/article/2143093/Search-Ad-Spending-Could-Hit-19.51-Billion-in-2012-Report, February 2012.

[16] Google Safe Browsing API. http://code.google.com/apis/safebrowsing/.

[17] Google Search Engine Optimization Starter Guide. http://www.google.com/webmasters/docs/search-engine-optimization-starter-guide.pdf.

[18] Google. Results labeled "This site may be hacked". http://support.google.com/websearch/answer/190597.

[19] Greer, Burns, & Crain. Anti-counterfeiting legal strategies, enforcement and remedies. http://gbclaw.net/practiceareas/anti-counterfeiting.

[20] Chris Grier, Lucas Ballard, Juan Caballero, Neha Chachra, Christian J. Dietrich, Kirill Levchenko, Panayiotis Mavrommatis, Damon McCoy, Antonio Nappa, Andreas Pitsillidis, Niels Provos, M. Zubair Rafique, Moheeb Abu Rajab, Christian Rossow, Kurt Thomas, Vern Paxson, Stefan Savage, and Geoffrey M. Voelker. Manufacturing Compromise: The Emergence of Exploit-as-a-Service. In *Proceedings of the ACM Conference on Computer and Communications Security*, Raleigh, NC, October 2012.

[21] Cormac Herley and Dinei Florencio. Nobody Sells Gold for the Price of Silver: Dishonesty, Uncertainty and the Underground Economy. In *Proceedings of the Workshop on the Economics of Information Security (WEIS)*, London, UK, June 2009.

[22] HtmlUnit — Welcome to HtmlUnit. http://htmlunit.sourceforge.net/.

[23] Luca Invernizzi, Stefano Benvenuti, Marco Cova, Paolo Milani-Comparetti, Christopher Kruegel, and Giovanni Vigna. EvilSeed: A Guided Approach to Finding Malicious Web Pages. In *Proceedings of the IEEE Symposium and Security and Privacy*, San Francisco, CA, May 2012.

[24] John P. John, Fang Yu, Yinglian Xie, Arvind Krishnamurthy, and Martin Abadi. deSEO: Combating Search-Result Poisoning. In *Proceedings of the USENIX Security Symposium*, San Francisco, CA, August 2011.

[25] Chris Kanich, Christian Kreibich, Kirill Levchenko, Brandon Enright, Vern Paxson, Geoffrey M. Voelker, and Stefan Savage. Spamalytics: an Empirical Analysis of Spam Marketing Conversion. In *Proceedings of the ACM Conference on Computer and Communications Security*, Alexandria, VA, October 2008.

[26] Chris Kanich, Nicholas Weaver, Damon McCoy, Tristan Halvorson, Christian Kreibich, Kirill Levchenko, Vern Paxson, Geoffrey M. Voelker, and Stefan Savage. Show Me the Money: Characterizing Spam-advertised Revenue. In *Proceedings of the USENIX Security Symposium*, San Francisco, CA, August 2011.

[27] Jon M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. In *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[28] Brian Krebs. Huge Decline in Fake AV Following Credit Card Processing Shakeup. http://krebsonsecurity.com/2011/08/huge-decline-in-fake-av-following-credit-card-processing- shakeup/, August 2011.

[29] Chris Larsen. Latest SEP (Search Engine Poisoning) Research, Part 1-7. http://www.bluecoat.com/security/security-archive/2012-02-15/latest-sep-search-engine-poisoning- research-part-1, February 2012.

[30] Nektarios Leontiadis, Tyler Moore, and Nicolas Christin. Measuring and Analyzing Search-Redirection Attacks in the Illicit Online Prescription Drug Trade. In *Proceedings of the USENIX Security Symposium*, San Francisco, CA, August 2011.

[31] Kirill Levchenko, Andreas Pitsillidis, Neha Chachra, Brandon Enright, Mrk Flegyhzi, Chris Grier, Tristan Halvorson, Chris Kanich, Christian Kreibich, He Liu, Damon McCoy, Nicholas Weaver, Vern Paxson, Geoffrey M. Voelker, and Stefan Savage. Click Trajectories: End-to-End Analysis of the Spam Value Chain. In *Proceedings of the IEEE Symposium and Security and Privacy*, Oakland, CA, May 2011.

[32] Jun-Lin Lin. Detection of Cloaked Web Spam by using Tag-Based Methods. *Expert Systems with Applications*, 36(4):7493–7499, May 2009.

[33] He Liu, Kirill Levchenko, Mark Felegyhazi, Christian Kreibich, Gregor Maier, Geoffrey M. Voelker, and Stefan Savage. On the Effects of Registrar-level Intervention.

In *Proceedings of the USENIX Workshop on Large-scale Exploits and Emergent Threats (LEET)*, Boston, MA, March 2011.

[34] Long Lu, Roberto Perdisci, and Wenke Lee. SURF: Detecting and Measuring Search Poisoning. In *Proceedings of the ACM Conference on Computer and Communications Security*, Chicago, IL, October 2011.

[35] MarkMonitor Brand Protection. https://www.markmonitor.com/services/brand-protection.php.

[36] Mark Maunder. Zero Day Vulnerability in many WordPress Themes. http://markmaunder.com/2011/08/01/zero-day-vulnerability-in-many-wordpress-themes/.

[37] Damon McCoy, Hitesh Dharmdasani, Christian Kreibich, Geoffrey M. Voelker, and Stefan Savage. Priceless: The Role of Payments in Abuse-advertised Goods. In *Proceedings of the ACM Conference on Computer and Communications Security*, Raleigh, NC, October 2012.

[38] Damon McCoy, Andreas Pitsillidis, Grant Jordan, Nicholas Weaver, Christian Kreibich, Brian Krebs, Geoffrey M. Voelker, Stefan Savage, and Kirill Levchenko. PharmaLeaks: Understanding the Business of Online Pharmaceutical Affiliate Programs. In *Proceedings of the USENIX Security Symposium*, Bellevue, WA, August 2012.

[39] Gene McKenna. Experiment Shows Up To 60% Of "Direct" Traffic Is Actually Organic Search. http://searchengineland.com/60-direct-traffic-actually-seo-195415.

[40] Tyler Moore and Richard Clayton. The consequence of non-cooperation in the fight against phishing. In *Proceesings of the eCrime Researchers Summit (eCRS)*, Atlanta, GA, October 2008.

[41] Tyler Moore, Nektarios Leontiadis, and Nicolas Christin. Fashion Crimes: Trending-Term Exploitation on the Web. In *Proceedings of the ACM Conference on Computer and Communications Security*, Chicago, IL, October 2011.

[42] Marc A. Najork. System and method for identifying cloaked web servers, United States Patent number 6,910,077. Issued June 21, 2005.

[43] Yuan Niu, Yi-Min Wang, Hao Chen, Ming Ma, and Francis Hsu. A Quantitative Study of Forum Spamming Using Contextbased Analysis. In *Proceedings of 15th Network and Distributed System Security (NDSS) Symposium*, San Diego, CA, February 2007.

[44] OpSec. Brand protection from manufacturing to retail. http://www.opsecsecurity.com/brand-protection.

[45] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.

[46] PricewaterhouseCoopers. IAB Internet Advertising Revenue Report, 2012 Full Year Results. http://www.iab.net/media/file/IABInternetAdvertisingRevenueReportFY2012POSTED.pdf.

[47] Moheeb Abu Rajab, Lucas Ballard, Panayiotis Marvrommatis, Niels Provos, and Xin Zhao. The Nocebo Effect on the Web: An Analysis of Fake Anti-Virus Distribution. In *Proceedings of the USENIX Workshop on Large-scale Exploits and Emergent Threats (LEET)*, San Jose, CA, April 2010.

[48] Safenames. Mark Protect. http://www.safenames.net/BrandProtection/MarkProtect.aspx.

[49] Nathan Safran. Update: Organic Search Is Actually Responsible for 64% of Your Web Traffic. http://www.conductor.com/blog/2014/07/update-organic-search-actually-responsible-64-web-traffic/.

[50] Dmitry Samosseiko. The Partnerka – What Is It, And Why Should You Care? In *Proceedings of the Virus Bulletin Conference*, September 2009.

[51] Search Engine Marketing Professional Organization (SEMPO). State of Search Engine Marketing Report Says Industry to Grow from $14.6 Billion in 2009 to $16.6 Billion in 2010. http://www.sempo.org/news/03-25-10, March 2010.

[52] SEOmoz. PageRank, Link Patterns & the New Flow of Link Juice. http://www.seomoz.org/blog/pagerank-link-patterns-the-new-flow- of-link-juice, May 2007.

[53] Craig Silverstein, Monika Henzinger, Hannes Marais, and Michael Moricz. Analysis of a Very Large Web Search Engine Query Log. *ACM SIGIR Forum*, 33(1):6–12, 1999.

[54] SMGPA. http://smgpa.net/.

[55] Julien Sobrier. Tricks to easily detect malware and scams in search results. http://research.zscaler.com/2010/06/tricks-to-easily-detect-malware-and.html, June 3, 2010.

[56] Brett Stone-Gross, Ryan Abman, Richard Kemmerer, Christopher Kruegel, Douglas Steigerwald, and Giovanni Vigna. The Underground Economy of Fake Antivirus Software. In *Proc. of the 10th Workshop on the Economics of Information Security (WEIS)*, Washington D.C., 2011.

[57] Ao-Jan Su, Y. Charlie Hu, Aleksandar Kuzmanovic, and Cheng-Kok Koh. How to Improve Your Google Ranking: Myths and Reality. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, Toronto, Canada, August 2010.

[58] Danny Sullivan. Search Engine Optimization Firm Sold For $95 Million. http://searchenginewatch.com/ 2163001, September 2000. Search Engine Watch.

[59] Jason Tabeling. Keyword Phrase Value: Click-Throughs vs. Conversions. http://searchenginewatch.com/ 3641985, March 8, 2011.

[60] Yi-Min Wang and Ming Ma. Detecting Stealth Web Pages That Use Click-Through Cloaking. Technical report, Microsoft Research, December 2006.

[61] Yi-Min Wang, Ming Ma, Yuan Niu, and Hao Chen. Spam Double-Funnel: Connecting Web Spammers with Advertisers. In *Proceedings of the International World Wide Web Conference (WWW)*, Banff, Alberta, May 2007.

[62] Colin Whittaker, Brian Ryner, and Marria Nazif. Large-Scale Automatic Classification of Phishing Pages. In *Proceedings of the Network and Distributed System Security Symposium*, San Diego, CA, February 2010.

[63] Wordtracker. Five Reasons Why Wordtracker Blows Other Keywords Tools Away. http://www.wordtracker.com/find-the-best-keywords.html.

[64] Baoning Wu and Brian D. Davison. Cloaking and Redirection: A Preliminary Study. In *Proceedings of the SIGIR Workshop on Adversarial Information Retrieval on the Web*, Chiba, Japan, May 2005.

[65] Baoning Wu and Brian D. Davison. Detecting Semantic Cloaking on the Web. In *Proceedings of the International World Wide Web Conference (WWW)*, Edinburgh, United Kingdom, May 2006.