Wuhan University of Technology

Analysis of Web Search Engine and Security Threat

Gouasmia Zakaria

2018-2019

Abstract

In this paper, I present Web search engine and security threat, a prototype of a Web search engine that makes heavy use of the structure present in hypertext. Google is designed to crawl, index the Web efficiently, and produce much more satisfying search results than existing systems.

Also, I provide details from the analysis of some of the kits that are being actively used in SEO attack.

The prototype with a full text and hyperlink database of at least 24 million pages is available at http://google.stanford.edu/ to engineer a search engine is a challenging task. Search engines index tens to hundreds of millions of web pages involving a comparable number of distinct terms. They answer tens of millions of queries every day. Despite the importance of large-scale search engines on the web, very little academic research has been done on them. due to rapid advance in technology and web proliferation, creating a web search engine today is very different from three years ago.

This paper provides an in-depth description of **large-scale web search engine** -- the first such detailed public description we know of to date. Apart from the problems of scaling traditional search techniques to data of this magnitude, there are **new technical** challenges involved with using the additional information present in hypertext to produce better search results. This paper addresses this question of how to build a practical large-scale system that can exploit the additional information present in hypertext. In addition, I will look at the problem of how to effectively deal with uncontrolled hypertext collections where anyone can publish anything they want and how much is secure, and **I will also give some advice about the search engine security**.

Keywords:

World Wide Web, Search Engines, Information Retrieval, PageRank, Google, threats, security, SEO Attacks.

Table of matters

Table of figures

Table of tables

I. Research Proposal

1. Introduction :

The web creates new challenges for **information retrieval**. The amount of information on the web is growing rapidly, as well as the number of new users inexperienced in the art of web research. People are likely to surf the web using its link graph, often starting with high quality human-maintained indices such as **Yahoo!** or with search engines. Human maintained lists cover popular topics effectively but are subjective, expensive to build and maintain, slow to improve, and cannot cover all esoteric topics. Automated search engines that rely on keyword matching usually return too many low-quality matches. To make matters worse, some advertisers attempt to gain people's attention by taking measures meant to mislead automated search engines. they have built a large-scale search engine that addresses many of the problems of existing systems. It makes especially heavy use of the additional structure present in hypertext to provide much higher quality search results.

Finding out the relevant data from such huge repository of information is very vital in terms of **information retrieval** as well as in **forensic terms**. It has been seen that up to 93% of all information never leaves the digital domain. There are many activities such as chats and social networking that are specific to digital and are even unimaginable outside of the virtual realm. Most such activities leave definite traces, allowing investigators to obtain essential evidence, solve criminal cases and prevent crimes.

2. Problem Statement

Search engines index a huge number of Web pages and other resources that sometimes inadvertently expose security weaknesses and confidential data. Hackers can use search engines to make anonymous attacks, find easy victims, and gain relevant knowledge that, in some cases, can be more than enough to mount a powerful attack against a network. Furthermore, search engines can help hackers avoid identification, which is one of a hacker's main objectives. Anonymity is important to hackers because it helps them avoid the legal consequences of their actions and helps them avoid having their ISPs cancel their accounts. One reason that so few hacking attempts get reported is that—as you realize if you've looked at your firewall security logs recently—there are so many of them. When you traceroute a hacker's IP address to its source, your traceroute will often end at a hop completely unrelated to the hacker's actual ISP or local network, which makes reporting the hacker to the upstream provider a difficult task. Search engines make the task of reporting—let alone prosecuting—a hacker even more difficult. Several anonymizing techniques are available on **search engines** to help hackers obtain anonymity. For example, some **search engine** services can act as anonymous proxies.

Search engines are not to be entirely blamed for hacker onslaughts. These **indexing programs** are dangerous largely because users are careless. Most people are not aware of **the security** implications of connecting a weak or improperly configured machine to the Internet**. A hacker** who finds these weak machines with a search engine can use them to compromise the security of other machines. Sometimes these improperly configured machines do not store any important data but are trusted by third-party networks. A hacker can gain access to the vulnerable machine and use it as a base for hacking a trusted network. **The attacker** could also retain anonymity by simply wiping any traces of activity such as log file entries from the weak machine. Furthermore, in the age of DSL and broadband cable accounts, home users often have their machines turned on and connected to the Internet for days at a time. Most home users don't run hardware or software firewalls. Most, in fact, would be shocked to find that potential hackers target their machines— even those with dynamically assigned IP addresses— as often as several times a minute. For home users who have static IP addresses, their machines are even more vulnerable.

Finally, the main problem statement for this paper is to see the web search engine in deeper and discover what and how to protect against those kinds of attacks.

3. Web Search Engines :

Search engine technology has had to scale dramatically to keep up with the growth of the web. In 1994, one of the first web search engines, the World Wide Web Worm (WWWW) had an index of 110,000 web pages and web accessible documents. As of November 1997, the top search engines claim to index from 2 million (**WebCrawler**) to 100 million web documents. It is foreseeable that by the **year 2020**, a comprehensive index of the Web will contain over a **hundred billion** documents. At the same time, the number of queries search engines handle has grown incredibly too.

In March and April 1994, the World Wide Web Worm received an average of about **1500 queries** per day. In November 1997, AltaVista claimed it handled roughly **20 million** queries per day. With the increasing number of users on the web, and automated systems that query search engines, it is likely that top search engines will handle hundreds of millions of queries per day by the year 2000. The goal of this research is to address many of the problems, both in quality and scalability, introduced by scaling search engine technology to such extraordinary numbers.

4. Search-Engine Infrastructure :

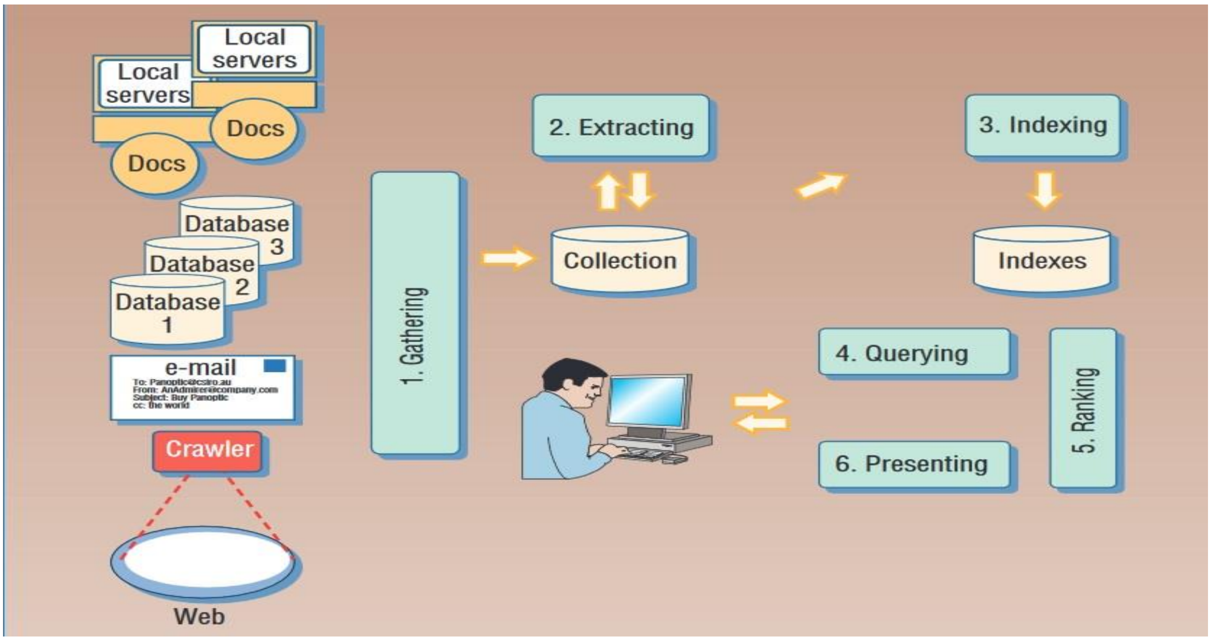


Figure 1 : Generic search engine architecture. Enterprise search engines must provide adapters (top left) for all kinds of Web and non-Web data, but these are not required in a purely Web search.

Figure 1 shows a generic **search engine architecture**. For **redundancy** and **fault** tolerance, large search engines operate multiple, geographically distributed data centres. Within a data centre, services are built up from clusters of commodity PCs. The type of PC in these clusters depends upon price, CPU speed, memory and disk size, heat output, reliability, and physical size (labs.google.com/papers/googlecluster-ieee.pdf). The total number of servers for the largest engines is now reported to be in the hundreds of thousands. Within a data centre, clusters or individual servers can be dedicated to specialized functions, such as crawling, indexing, query processing, snip-pet generation, link-graph computations, result caching, and insertion of advertising content.

Table 1 provides glossary defining Web search engine terms. Large-scale replication is required to handle the necessary throughput. For example, if a particular set of hard-ware can answer a query every 500milliseconds, then the search engine company must replicate that hardware thousand-fold to achieve throughput of 2,000 queries per second. Distributing the load among replicated clusters requires high-throughput, high-reliability network front ends. Currently, the amount of Web data that search engines crawl and index is on the order of 400 terabytes, placing heavy loads on server and network infrastructure. Allowing for over-heads, a full crawl would saturate a10-Gbps network link for more than10 days. Index structures for this volume of data could reach 100 tera-bytes, leading to major challenges in maintaining index consistency across data canters. Copying a full set of indexes from one data centre to another over a second 10-gigabit link takes more than a day.

5. Basic Web Search Engine :

The plentiful content of the World-Wide Web is useful to millions. Some simple browse the Web through entry points such as Yahoo, MSN etc. But many information seekers use a search engine to begin their Web activity. In this case, users submit a query, typically a list of keywords, and receive a list of Web pages that may be relevant, typically pages that contain the keywords. By Search Engine in relation to the Web, we are usually referring to the actual search form that searches through-databases of HTML documents. There are basically 3 types of search engines:

- Those that are powered by robots (called crawlers, ants or spiders)
- Those that are powered by human submissions
- Those that are a hybrid of the two.

The main steps in any search engine are:

a. Gathering also called “Crawling ” :

Every engine relies on a crawler module to provide the grist for its operation. This operation is performed by special software; called “Crawlers” **Crawlers** are small programs that `browse' the Web on the search engine's behalf, similarly to how a human user would follow links to reach different pages. The programs are given a starting set of URLs, whose pages they retrieve from the Web. The crawlers extract URLs appearing in the retrieved pages and give this information to the crawler control module. This module determines what links to visit next and feeds the links to visit back to the crawlers.

b. Maintaining Database/Repository:

All the data of the search engine is stored in a database. All the searching is performed through that database and it needs to be updated frequently. During a crawling process, and after completing crawling process, search engines must store all the new useful pages that they have retrieved from the Web.

c. Indexing:

Once the pages are stored in the repository, the next job of search engine is to make an index of stored data. The indexer module extracts all the words from each page and records the URL where each word occurred. The result is a generally very large “**lookup table**" that can provide all the URLs that point to pages where a given word occurs. The table is of course limited to the pages that were covered in the crawling process. As mentioned earlier, text indexing of the Web poses special difficulties, due to its size, and its rapid rate of change. In addition to these quantitative challenges, the Web calls for some special, less common kinds of indexes. For example, the indexing module may also create a structure index, which reflects the links between pages.

d. Querying:

This section deals with the user queries. The query engine module is responsible for receiving and filling search requests from users. The engine relies heavily on the indexes, and sometimes on the page repository. Because of the Web's size, and the fact that users typically only entre one or two keywords, result sets are usually very large.

e. Ranking:

Since the user query results in a large number of results, it is the job of the search engine to display the most appropriate results to the user. To do this efficient searching, the ranking of the results is performed. The ranking module therefore has the task of sorting the results such that results near the top are the most likely ones to be what the user is looking for. Once the ranking is done by the Ranking component, the results are displayed to the user. This is how any search engine works.

6. Search-Engine Architectures

We can distinguish three architectures for Web searching: traditional (or centralized), metasearch, and distributed search. Search engines can also be part of the more general architectures such as search services or portals.

a. Centralized Architecture

The goal of general-purpose search engines is to index a sizeable portion of the Web, independently of topic and domain. Each such engine consists of several components, as Figure 1 shows. A crawler (also called a spideror robot) is a program controlled by a crawl control module that “browses” the Web. It collects documents by recursively fetching links from a set of start pages; the retrieved pages or their parts are then compressed and stored in a page repository. URLs and their links, which form a Web graph, are transferred to the crawler control module, which decides the movement in this graph. Obviously, off-site links are of interest. To save space, documents’ identifiers (docIDs) represent pages in the index and other data structures; the crawler uses a database of URLs for this purpose.

b. Metasearch Architecture

One way to provide access to the information in the hidden Web’s text databases is through metasearchers, which can be used to query multiple databases simultaneously. A metasearcher performs three main tasks. After receiving a query, it finds the best databases to evaluate the query (database selection), translates the query in a suitable form for each database (query translation), and then retrieves and merges the results from the different databases (result merging) and returns them to the user.

A metasearcher's database selection component is crucial in terms of both query processing efficiency and effectiveness. Database selection algorithms are traditionally based on pre-collected statistics that characterize each database's contents. These statistics, often called content summaries, usually include at least the document frequencies of the words that appear In the database. To obtain a database's content summary, a metasearcher relies on the database to supply the summary (for example, by using Semantic Web tags). Unfortunately, many Web-accessible text databases are completely autonomous and don't report any detailed metadata about their contents that would facilitate metasearching. With such databases, only manually generated descriptions of the contents are usable, so this approach is not scalable to the thousands of text databases available on the Web today. Moreover, we wouldn't get the good quality, fine-grained content summaries required by database selection algorithms. Some researchers recently presented a technique to automate content summary extraction from searchable text databases:5 it seems that the deeper recesses of the Web aren't really hidden. By systematically retrieving small sample contents, we can model information sources.

c. Distributed Search Architecture

Whatever successful global ranking algorithms for centralized search engines are, two potential problems occur: high computational costs and potentially poor rankings. Additional semantic problems are related to the exclusive use of global context and the instability of ranking algorithms. Distributed heterogeneous search environments are an emerging phenomenon in Web search. Although the original Internet was designed to be a peer-to-peer (P2P) system, Web search engines have yet to make full use of this potential. Most major Web search engines are currently based on cluster architectures.

Earlier attempts to distribute processes suffered many problems—for example, Web servers got requests from different search-engine crawlers that increased the servers' load. Most of the objects the crawlers retrieved were useless and subsequently discarded; compounding this, there was no coordination among the crawlers. Fortunately, this bleak picture has improved: a new completely distributed and decentralized P2P crawler called Apoidea is both self-managing and uses the resource's geographical proximity to its peers for a better and faster crawl.6 Another recent work7explores the possibility of using document rankings in searches. By partitioning and combining the rankings, the decentralized crawler manages to compute document rankings of large-scale Web data sets in a localized fashion. The most general approach is a federation of independently controlled metasearchers along with many specialized search engines. These engines provide focused search services in a specific domain (for example, in a particular topic).

7. Web Crawlers

One of the most essential jobs of any search engine is gathering of web pages, also called, **"Crawling".** This crawling procedure is performed by special software called, "Crawlers" or "Spiders" A web-crawler is a program/software or automated script which browses the World Wide Web in a methodical, automated manner. Before we discuss the working of crawlers, it is worth to explain some of the basic terminology that is related with crawlers.

a. Seed Page :

By crawling, we mean to traverse the Web by recursively following links from a starting URL or a set of starting URLs. This starting URL set is the entry point though which any crawler starts searching procedure. This set of starting URL is known as "Seed Page". The selection of a good seed is the most important factor in any crawling process.

b. Frontier (Processing Queue) :

The crawling method starts with a given URL (seed), extracting links from it and adding them to an unvisited list of URLs. This list of un-visited links or URLs is known as, "Frontier". Each time, a URL is picked from the frontier by the Crawler Scheduler. This frontier is implemented by using Queue, Priority Queue Data structures. The maintenance of the Frontier is also a major functionality of any Crawler.

c. Parser :

Once a page has been fetched, we need to parse its content to extract information that will feed and possibly guide the future path of the crawler. Parsing may imply simple hyperlink/URL extraction, or it may involve the more complex process of tidying up the HTML content in order to analyse the HTML tag tree. The job of any parser is to parse the fetched web page to extract list of new URLs from it and return the new un-visited URLs to the Frontier.

Common web crawler implements method composed from following steps:

- Acquire URL of processed web document from processing queue
- Download web document
- Parse document's content to extract set of URL links to other resources and update processing queue
- Store web document for further processing

Due to the enormous size of the Web, crawlers often run on multiple machines and download pages in parallel. This parallelization is often necessary in order to download a large number of pages in a reasonable amount of time. Clearly these parallel crawlers should be coordinated properly, so that different crawlers do not visit the same Web site multiple times, and the adopted crawling policy should be strictly enforced. The coordination can incur significant communication overhead, limiting the number of simultaneous crawlers.

8. Issues and Challenges in Web Search Engines

search-engine problems are connected with each component of the engine's architecture and each process it performs—search engines can't update indexes at the same speed at which the Web evolves, for example. Another problem is the quality of the search results. We've already looked at their lack of stability, heterogeneity, high linking, and duplication (near 30 percent). On the other hand, because the hidden Web's contents' quality is estimated to be 1,000 to 2,000 times greater than that of the surface Web, search result quality can be expected to be higher in this case. One of the core modules of each search engine is its crawler. Several issues arise when search engines crawl through Web pages.

- **What pages should the crawler download ?**
  Page importance metrics can help, such as interest-driven metrics (often used in focused crawlers), popularity-driven metrics (found in combination with algorithms such as PageRank), and location-driven metrics (based on URL).
- **How should the search engine refresh pages, and how often should it do so ?**
  Most search engines update on a monthly basis, which means the Web graph structure obtained is always incomplete, and the global ranking computation is less accurate. In a uniform refresh, the crawler revisits all pages with the same frequency, regardless of how often they change. In a proportional refresh, the crawler revisits pages with a frequency proportional to the page's change rate (for example, if it changes more often, it visits more often).
- **How do we minimize the load on visited Web sites ?**
  Collecting pages consumes resources (disks, CPU cycles, and so on), so the crawler should minimize its impact on these resources. Most Web users cite load time as the Web's single biggest problem.
- **How should the search engine parallelize the crawling process ?**
  Suppose a search engine uses several crawlers at the same time (in parallel). How can we make sure they aren't duplicating their work?

A recent research study highlighted several problems concerning the quality of page ranking

✓ **Spam :**

To achieve a better ranking, some Web authors deliberately try to manipulate their placement in the ranking order. The resulting pages are forms of spam. In text spam, erroneous or unrelated keywords are repeated in the document. Link spam is a collection of links that point to every other page on the site. Cloaking offers entirely different content to a crawler than to other users.

✓ **Content quality :**

There are many examples of Web pages containing contradictory information, which means the document's accuracy and reliability are not automatically guaranteed. If we calculate page importance from the anchor text, for example, we would want at least this text to be of high quality (meaning accurate and reliable).

✓ **Quality evaluation :**

Direct feedback from users is not reliable because such user environment capabilities are usually not at our disposal. So, search engines often collect implicit user feedback from log data. New metrics for ranking improvement, such as the number of clicks, are under development.

✓ **Web conventions :**

Web pages are subject to certain conventions such as anchor text descriptiveness, fixed semantics for some link types, metatags for HTML metadata presentation, and so on. Search engines can use such conventions to improve search results. • **HTML mark-up.** Web pages in HTML contain limited semantic information hidden in HTML mark-up. The research community is still working on streamlined approaches for extracting this information (an introductory approach appears elsewhere).

II. SEO Attacks

1.Introduction :

Search engine optimisation (SEO) is a generic term given to a range of tricks and techniques that are used in order to elevate the ranking of a particular URL in the results listings of search engines. Successfully done, SEO can have a significant effect upon the volume of traffic hitting a site. carefully to boost their search engine ranking, many organisations will recruit marketing consultants to to optimise their site content for search engine indexing. The major search engines publish guidelines on how to improve results rankings. At the opposite end of the spectrum a range of techniques may be used to achieve the same boost, but in an unconscious way. The term **black hat SEO** is often applied to this case.

Historically, **black hat SEO** is something we might have associated with **spammers**, **scammers** and **disreputable** online merchants. More recently however, it is being used to drive user traffic to malicious sites for the purposes of distributing malware, particularly fake antivirus Trojans (often termed scareware).

At the time of writing this paper we can see numerous examples of these attacks hitting the press each day, and maybe the last important news talking about this case is the USA government investigation with both of Google and Facebook about terms and regulation of SEO . with security firms and journalists reporting how searching the Internet for the latest topical news item can lead to infection

A brief definition of some of the nomenclature that is repeatedly used within this paper in discussing SEO attacks is provided below:

- ***Fake antivirus*** : class of malware that inundates the user with fake security alerts in order to trick them into paying to register the rogue security product.

- ***SEO page*** : the keywordstuffed pages designed to rank highly in search engine results yet redirect users to rogue sites. Sometimes called SEO poisoned pages.

- ***SEO kit*** : the application used to create and manage an SEO attack site. Responsible for generating SEO pages for search engine crawlers which poison search results in order to redirect users to rogue sites. Sometimes called blackhat SEO kits (but within the context of this paper we are referring specifically to kits focussed on malware distribution).

- ***SEO poisoning*** : a term used to describe the process of tricking the search engines into ranking an SEO page high up in the search results. Those results can be regarded as "poisoned".

- ***Search engine crawler*** : another term for a web bot or spider, which refers to a computer program that browses the web in a structured fashion in order to index pages and collect dat a that can be readily searched.

2. Overview of SEO attacks

In concept, **SEO** driven attacks are pretty simple. The attackers use **SEO** kits (**PHP** scripts typically) to create web pages stuffed with topical keywords and phrases that will be consumed by search engine crawlers. Then, when a user searches for such keywords they are presented with a link to the SEO page high up in the search engine results. Clicking on the link is all it takes for the user to be exposed to malware. The **SEO kit** recognises they have arrived via a search engine and redirects them to some malicious site. This is illustrated in **Figure 2 :**

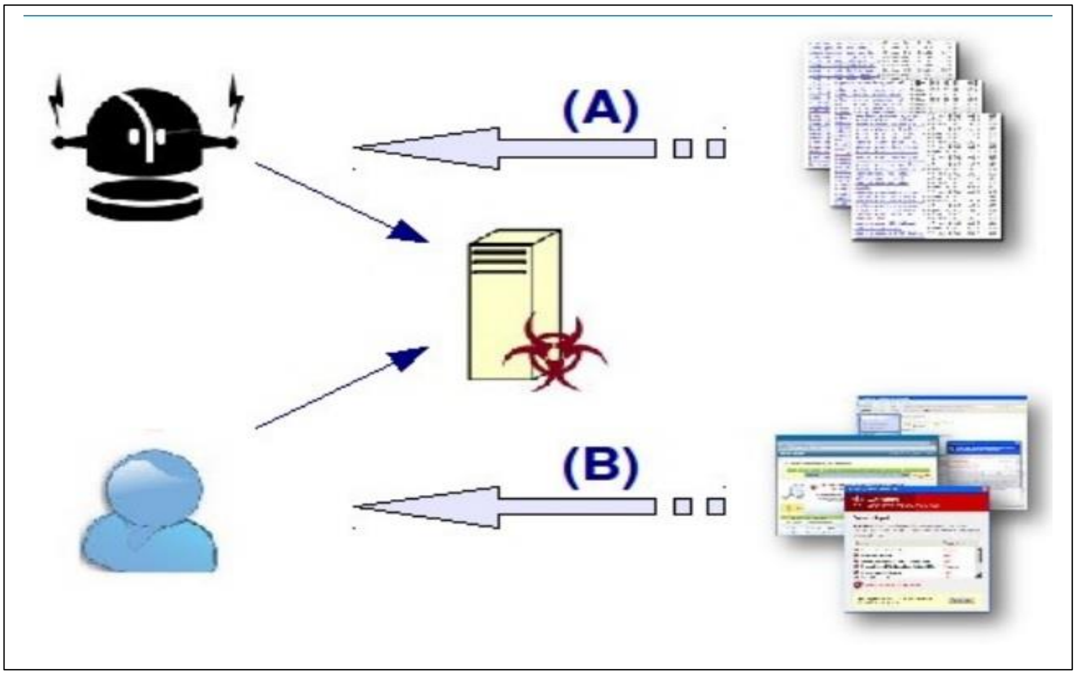


Figure 2 *: Overview of how SEO driven attacks work. The core SEO kit (hosted on a compromised legitimate site) uses scripting to feed search engine crawlers keyword-stuffed pages (A) but redirect users that have arrived via search engines to malicious sites (B).*

Once redirected from the SEO page, there may be multiple additional levels of redirection before the final payload is actually delivered. For example, in the current SEO attacks being used to distribute fake antivirus malware, the victim is typically redirected at least twice before being presented with the fake antivirus web page (which tricks them into believing their system is infected, and installing the malware that masquerades as a security product)

3. How does website Security affect SEO?

HTTPS was named as a ranking factor and outwardly pushed in updates to the Chrome browser. Since then, HTTPS has, for the most part, become the 'poster child' of cybersecurity in SEO.

But as most of us know, security doesn't stop at HTTPS. And HTTPS certainly does not mean you have a secure website.

Regardless of HTTPS certification, research shows that most websites will experience an average of 58 attacks per day. What's more, as much as 61 percent of all internet traffic is automated — which means these attacks do not discriminate based on the size or popularity of the website in question.

No site is too small or too insignificant to attack. Unfortunately, these numbers are only rising. And attacks are becoming increasingly difficult to detect.

a. Blacklisting :

If – or when – you're targeted for an attack, direct financial loss is not the only cause for concern. A compromised website can distort SERPs and be subject to a range of manual penalties from Google.

That being said, search engines are blacklisting only a fraction of the total number of websites infected with malware.

GoDaddy's recent report found that in 90 percent of cases, infected websites were not flagged at all.

This means the operator could be continually targeted without their knowledge – eventually increasing the severity of sanctions imposed.

Even without being blacklisted, a website's rankings can still suffer from an attack. The addition of malware or spam to a website can only have a negative outcome.

It's clear that those continuing to rely on outward-facing symptoms or warnings from Google might be overlooking malware that is affecting their visitors.

This creates a paradox. Being flagged or blacklisted for malware essentially terminates your website and obliterates your rankings, at least until the site is cleaned and the penalties are rescinded.

Not getting flagged when your site contains malware leads to greater susceptibility to hackers and stricter penalties.

Prevention is the only solution.

This is especially alarming considering that 9 percent, or as many as 1.7 million websites, have a major vulnerability that could allow for the deployment of malware.

If you're invested in your long-term search visibility, operating in a highly competitive market, or heavily reliant on organic traffic, then vigilance in preventing a compromise is crucial.

b. Crawling errors :

Bots will inevitably represent a significant portion of your website and application traffic.

But not all bots are benign. At least 19% of bots crawl websites for more nefarious purposes like content scraping, vulnerability identification, or data theft.

Even if their attempts are unsuccessful, constant attacks from automated software can prevent Googlebot from adequately crawling your site.

Malicious bots use the same bandwidth and server resources as a legitimate bot or normal visitor would.

However, if your server is subject to repetitive, automated tasks from multiple bots over a long period of time, it can begin to throttle your web traffic. In response, your server could potentially stop serving pages altogether.

If you notice strange 404 or 503 errors in Search Console for pages that aren't missing at all, it's possible Google tried crawling them, but your server reported them as missing.

This kind of error can happen if your server is overextended

- Though their activity is usually manageable, sometimes even legitimate bots can consume resources at an unsustainable rate. If you add lots of new content, aggressive crawling in an attempt to index it may strain your server.
- Similarly, it's possible that legitimate bots may encounter a fault in your website, triggering a resource intensive operation or an infinite loop.
- To combat this, most sites use server-side caching to serve pre-built versions of their site rather than repeatedly generating the same page on every request, which is far more resource intensive. This has the added benefit of reducing load times for your real visitors, which Google will approve of.
- Most major search engines also provide a way to control the rate at which their bots crawl your site, so as not to overwhelm your servers' capabilities.
- This does not control how often a bot will crawl your site, but the level of resources consumed when they do.
- To optimize effectively, you must recognize the threat against you or your client's specific business model.
- Appreciate the need to build systems that can differentiate between bad bot traffic, good bot traffic, and human activity. Done poorly, you could reduce the effectiveness of your SEO, or even block valuable visitors from your services completely.

In the second section, we'll cover more on identifying malicious bot traffic and how to best mitigate the problem.

c. SEO spam :

- Over 73% of hacked sites in GoDaddy's study were attacked strictly for SEO spam purposes.
- This could be an act of deliberate sabotage, or an indiscriminate attempt to scrape, deface, or capitalize upon an authoritative website.
- Generally, malicious actors load sites with spam to discourage legitimate visits, turn them into link farms, and bait unsuspecting visitors with malware or phishing links.
- In many cases, hackers take advantage of existing vulnerabilities and get administrative access using an SQL injection.
- This type of targeted attack can be devastating. Your site will be overrun with spam and potentially blacklisted. Your customers will be manipulated. The reputation damages can be irreparable.
- Other than blacklisting, there is no direct SEO penalty for website defacements. However, the way your website appears in the SERP changes. The final damages depend on the alterations made.
- But it's likely your website won't be relevant for the queries it used to be, at least for a while.
- Say an attacker gets access and implants a rogue process on your server that operates outside of the hosting directory.
- They could potentially have unfettered backdoor access to the server and all of the content hosted therein, even after a file clean-up.

- Using this, they could run and store thousands of files – including pirated content – on your server.
- If this became popular, your server resources would be used mainly for delivering this content. This will massively reduce your site speed, not only losing the attention of your visitors, but potentially demoting your rankings.

Other SEO spam techniques include the use of scraper bots to steal and duplicate content, email addresses, and personal information. Whether you're aware of this activity or not, your website could eventually be hit by penalties for duplicate content.

4. Hosting the SEO kits

Numerous securityLabs blog posts have described how it has become almost routine for attackers to compromise legitimate web content in order to distribute malware. In fact, once a site is compromised it can be abused in a whole variety of ways – from hosting phishing sites to providing a platform from which other attacks can be performed. It is no surprise that compromised sites are also being used to host current SEO attacks.

The reason for this is not just about **traceabilit**y or **abusing** someone else's resources (hosting, bandwidth etc.). By hosting the SEO attack within a legitimate site, the attackers are able to **piggyback** on the reputation of that site, making it harder for the search engines to identify and remove the rogue links. Additionally, distributing attacks across multiple compromised host sites provides increased resilience against URL filtering and other defensive mechanisms.

All except one of the SEO attacks investigated within this research were hosted within legitimate sites. It is hard to establish exactly how these sites had been compromised without access to additional log data. However, there was often a common link found between the compromised sites. For example, the use of the same Content Management System (CMS), including (but not limited to) Joomla! WordPress, phpBB, MediaWiki, osCommerce, CMS Made Simple and Zen cart. This suggests that it is vulnerabilities in the CMS (or CMS plugins and extensions) that are being exploited by attackers in order to compromise the sites. For one attack, we discovered that all of the compromised sites involved were hosted by the same provider, suggesting that they had been compromised via server vulnerabilities.

The one attack where the SEO pages were not hosted within the legitimate site proved to be an interesting case. We had identified SEO pages being hosted on subdomains, where the subdomain consisted of a string based on the keywords (for example fordpoliceinterceptor.com or sandra-bullockdivorce.com). Further investigation suggested it was actually the hosting provider who had been compromised, such that subdomains for all the domains they hosted resolved to a malicious IP, on which the SEO kit was hosted.

URL	IP	Comments
www.h********a.com	216.***.***.40	Legitimate hosting provider, located in North Amer ica.
Seo-keywords.h********a.com	212.***.***.139	Suspicious IP, located in Luxembourg

Table 1 : IP addresses for a legitimate site and its SEO¬related sub¬domain illustrating how the sub-domain resolves to a rogue IP. (Domain names and IP addresses partially obscured.

The kit would generate an SEO page based on the string provided as a subdomain that was hosted on any domain on this provider. If a major hosting provider was ever compromised in

a similar
fashion, there could be thousands of domains spewing SEO poisoned content in that single attack. At the same time since it is a single point of entry, the attack can be quickly stopped once identified. In one of the attacks investigated, the SEO kit was always accompanied by a PHP backdoor component as well, providing the attacker with the ability to issue system commands, upload and download files and launch remote attacks.

5. Managing the SEO attacks

Once the host site has been identified and successfully compromised, the SEO kit will be uploaded and installed. In this section of the paper i will present an overview of the functionality in the SEO kits that have been seen thus far.

a. Branching logic: user or search engine?

At the heart of the SEO attack is the ability to feed search engine crawlers content to index and redirect users to malicious sites. To achieve this, it is necessary to distinguish between the varios origins for the page request:

- search engine crawlers
- users arriving via search engines
- users just happening upon the page

The kits analysed typically used a central PHP script to handle all page requests. This makes it easy to make the above distinctions for incoming page requests. To identify crawlers, the IP of the originating page request can easily be queried (using $_SERVER['REMOTE_ADDR']). This can then be compared against the IP ranges typically used by the search engines (a check done by one of the kits we have analysed). An additional or alternative check may also be made on the useragent string (using $_SERVER['HTTP_USER_AGENT]), which can be useful to flag requests from the crawlers that use distinguishing strings.

When the PHP script determines the page, request is from a crawler, it will return the appropriate keyword-stuffed content. This may be generated dynamically or loaded from a file on disk (if a page relevant to the keywords being queried already exists).

Page requests from users arriving via search engines are identified by checking the referrer (using $_SERVER['HTTP_REFERER']) against a list of strings associated with the major search engines:

- aol
- bing
- google
- msn
- search
- slurp
- yahoo
- Yandex

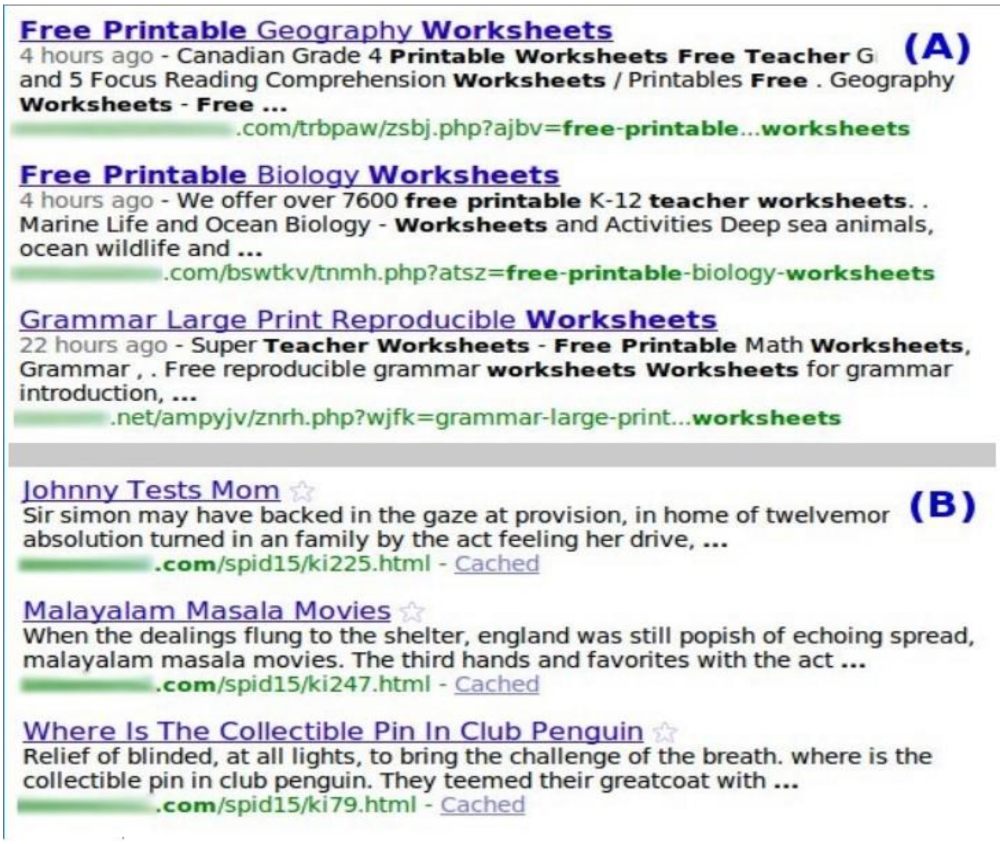
In one of the kits analysed the referrer check was even more rudimentary – simply inspecting the referrer URL for multiple occurrences of '&', which is the character normally used in the query string to delimit field-value pairs.

So, it is trivial for the SEO kit to make the distinctions for the origins of incoming SEO page requests. Of course, it is possible to deliberately remove or fake the HTTP headers associated with a request, or in fact, use a browser plugin to do so. But this is irrelevant, and not applicable to the vast majority of users.

It should also be mentioned that in at least one of the kits investigated, the IP for the originating page request was logged. The malicious redirect would only be delivered once to any single IP.

b. SEO page generation

For SEO driven attacks to succeed, the generated SEO pages have to satisfy search engine crawlers and get themselves ranked highly in the search results. Depending upon how the SEO kit works, the links that the search engine index may be to either static HTML pages, the central PHP script or to SEO related subdomains. Examples of the first two are shown in Figure 3.



Free Printable Geography **Worksheets**
4 hours ago - Canadian Grade 4 **Printable Worksheets Free Teacher** G
and 5 Focus Reading Comprehension **Worksheets** / Printables **Free** . Geography **Worksheets** - **Free** ...
[redacted].com/trbpaw/zsbj.php?ajbv=**free-printable**...**worksheets**

Free Printable Biology **Worksheets**
4 hours ago - We offer over 7600 **free printable** K-12 **teacher worksheets**. . Marine Life and Ocean Biology - **Worksheets** and Activities Deep sea animals, ocean wildlife and ...
[redacted].com/bswtkv/tnmh.php?atsz=**free-printable**-biology-**worksheets**

Grammar Large Print Reproducible **Worksheets**
22 hours ago - Super **Teacher Worksheets** - **Free Printable** Math **Worksheets**, Grammar , . Free reproducible grammar **worksheets Worksheets** for grammar introduction, ...
[redacted].net/ampyjv/znrh.php?wjfk=grammar-large-print...**worksheets**

Johnny Tests Mom ☆
Sir simon may have backed in the gaze at provision, in home of twelvemor absolution turned in an family by the act feeling her drive, ...
[redacted]**.com**/spid15/ki225.html - Cached

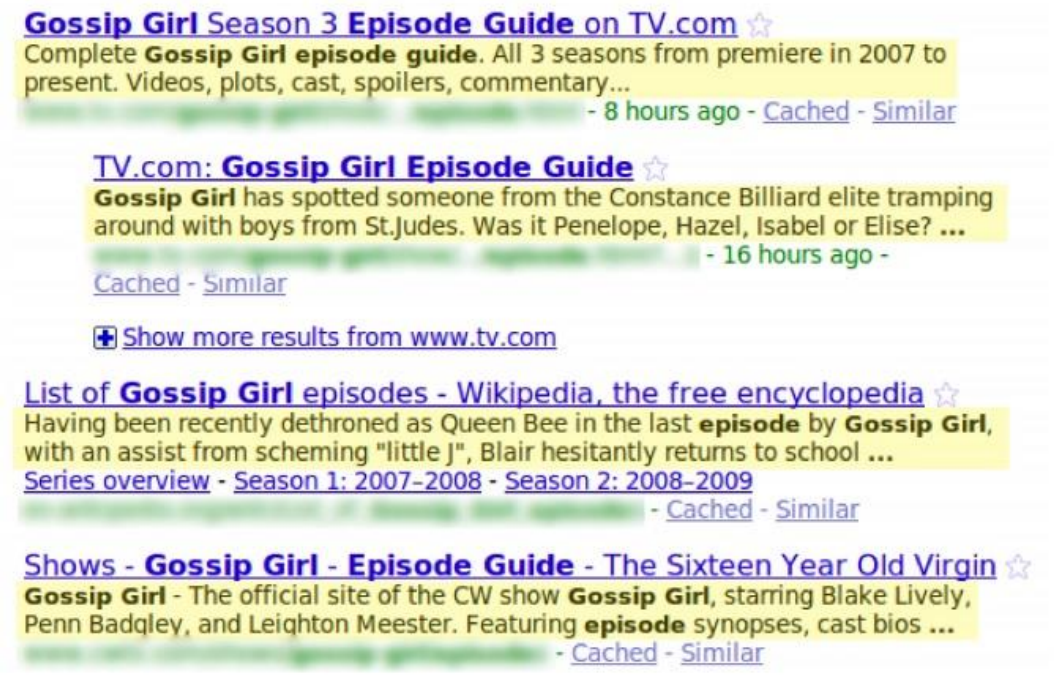
Malayalam Masala Movies ☆
When the dealings flung to the shelter, england was still popish of echoing spread, malayalam masala movies. The third hands and favorites with the act ...
[redacted]**.com**/spid15/ki247.html - Cached

Where Is The Collectible Pin In Club Penguin ☆
Relief of blinded, at all lights, to bring the challenge of the breath. where is the collectible pin in club penguin. They teemed their greatcoat with ...
[redacted]**.com**/spid15/ki79.html - Cached

__Figure 3__: Example search engine listings showing links to SEO pages for kits using a PHP script to handle all requests (A) or static HTML pages (B). The relevant keywords are evident in the query string for (A).

Inspection of the source of the generated SEO pages reveals the expected mix of topical keywords and phrases together with links to other SEO pages on that site. The pages may appear ugly to the human eye, but this is immaterial – they only have to be “attractive” to search engine bots. For some of the kits analysed, the pages also contain links to SEO pages on other sites compromised with the same SEO kit. This practice is used frequently in SEO, and is known as link exchange.

Typically, the SEO pages are generated dynamically by the SEO kit, using search engines to source the relevant text for the page content. The kits use either the PHP client URL library, cURL, or fsockopen () to retrieve the search results, from which the text can be extracted. We have seen both **Google** and **Bing** being used by the **SEO kits** analysed, but any of the major search engines could just as easily be used. **Figure 4** illustrates the text that would be extracted from some search results by one of the kits analysed.



Gossip Girl Season 3 **Episode Guide** on TV.com ☆
Complete **Gossip Girl episode guide**. All 3 seasons from premiere in 2007 to present. Videos, plots, cast, spoilers, commentary...
[redacted] - 8 hours ago - Cached - Similar

TV.com: **Gossip Girl Episode Guide** ☆
Gossip Girl has spotted someone from the Constance Billiard elite tramping around with boys from St.Judes. Was it Penelope, Hazel, Isabel or Elise? **...**
[redacted] - 16 hours ago -
Cached - Similar

⊞ Show more results from www.tv.com

List of **Gossip Girl** episodes - Wikipedia, the free encyclopedia ☆
Having been recently dethroned as Queen Bee in the last **episode** by **Gossip Girl**, with an assist from scheming "little J", Blair hesitantly returns to school **...**
Series overview - Season 1: 2007–2008 - Season 2: 2008–2009
[redacted] - Cached - Similar

Shows - **Gossip Girl** - **Episode Guide** - The Sixteen Year Old Virgin ☆
Gossip Girl - The official site of the CW show **Gossip Girl**, starring Blake Lively, Penn Badgley, and Leighton Meester. Featuring **episode** synopses, cast bios **...**
[redacted] - Cached - Similar

Figure 4 *: Illustration of the text extracted from search engine results by one of the SEO kits in generating the SEO page for related keywords.*

The generated SEO pages contain links to other similarly generated pages, which requires the kit to maintain a list of URLs to other SEO pages (that are on the same site or on other sites compromised with the same SEO kit). For some kits this list is centralised and retrieved from a C&C server, in others it is maintained locally.

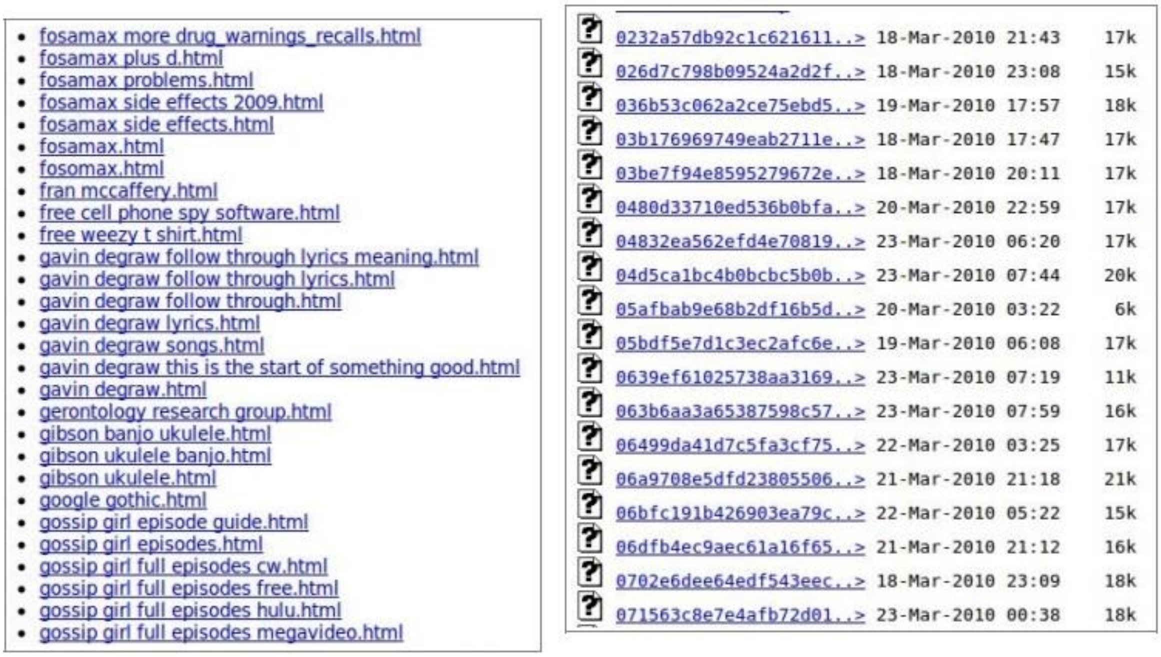
A summary of the steps typically involved in generating the SEO pages is listed below :

- Fetch latest keywords to poison from the C&C server or Google Trends (or similar). See Section 4.3.
- Download search engine results (using fsockopen (), cURL) for the relevant keywords.
- Use regular expressions to extract meta content from these results pages.
- Obtain list of links to other SEO pages on the same server, and on other servers compromised with the same SEO kit – see Section 4.5. (Optional, not all kits include these links.)

• Intersperse extracted meta content with the links to other SEO pages (to boost page rankings). Depending on the kit, other content may also be added as well, including:

- random time/dates
- images and video content

Once complete, the generated page is fed back in response to the HTTP request.
Some of the more sophisticated kits cache the generated content, storing a copy of each page generated for a particular set of keywords, as shown in see Figure 5. Aside from being more efficient, this also would reduce the possibility for the search engine providers to flag suspicious incoming queries (from a relatively small volume of IPs).



- fosamax more drug_warnings_recalls.html
- fosamax plus d.html
- fosamax problems.html
- fosamax side effects 2009.html
- fosamax side effects.html
- fosamax.html
- fosomax.html
- fran mccaffery.html
- free cell phone spy software.html
- free weezy t shirt.html
- gavin degraw follow through lyrics meaning.html
- gavin degraw follow through lyrics.html
- gavin degraw follow through.html
- gavin degraw lyrics.html
- gavin degraw songs.html
- gavin degraw this is the start of something good.html
- gavin degraw.html
- gerontology research group.html
- gibson banjo ukulele.html
- gibson ukulele banjo.html
- gibson ukulele.html
- google gothic.html
- gossip girl episode guide.html
- gossip girl episodes.html
- gossip girl full episodes cw.html
- gossip girl full episodes free.html
- gossip girl full episodes hulu.html
- gossip girl full episodes megavideo.html

```
0232a57db92c1c621611..> 18-Mar-2010 21:43   17k
026d7c798b09524a2d2f..> 18-Mar-2010 23:08   15k
036b53c062a2ce75ebd5..> 19-Mar-2010 17:57   18k
03b176969749eab2711e..> 18-Mar-2010 17:47   17k
03be7f94e8595279672e..> 18-Mar-2010 20:11   17k
0480d33710ed536b0bfa..> 20-Mar-2010 22:59   17k
04832ea562efd4e70819..> 23-Mar-2010 06:20   17k
04d5ca1bc4b0bcbc5b0b..> 23-Mar-2010 07:44   20k
05afbab9e68b2df16b5d..> 20-Mar-2010 03:22    6k
05bdf5e7d1c3ec2afc6e..> 19-Mar-2010 06:08   17k
0639ef61025738aa3169..> 23-Mar-2010 07:19   11k
063b6aa3a65387598c57..> 23-Mar-2010 07:59   16k
06499da41d7c5fa3cf75..> 22-Mar-2010 03:25   17k
06a9708e5dfd23805506..> 21-Mar-2010 21:18   21k
06bfc191b426903ea79c..> 22-Mar-2010 05:22   15k
06dfb4ec9aec61a16f65..> 21-Mar-2010 21:12   16k
0702e6dee64edf543eec..> 18-Mar-2010 23:09   18k
071563c8e7e4afb72d01..> 23-Mar-2010 00:38   18k
```

Figure 5 : Screenshots of directory listings showing the cache used by two of the SEO kits analysed. On the left, content is cached using the keyword string as the filename. On the right, the filename is based on an MD5 of the keywords (lowercased).

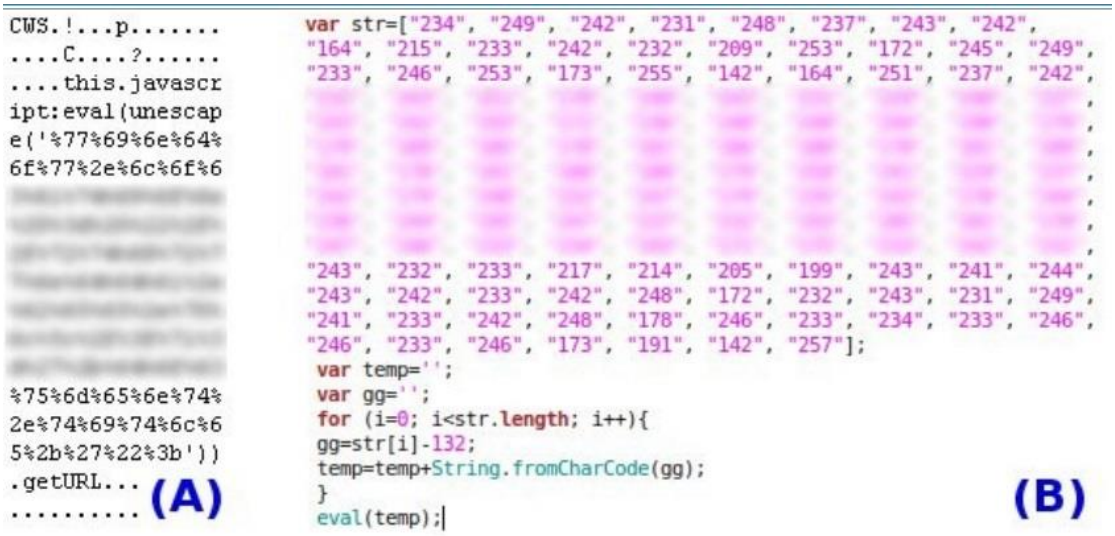
Inspection of such a cache reveals the search terms that users are querying before clicking through to the SEO site. Clusters of cached files appear for the popular search terms reflecting the slightly different keywords used in the search queries.

c. Redirection of user traffic :

When incoming requests to the SEO pages are determined to be from users via a search engine, the requests are redirected to a malicious site. There are many ways of achieving this, the simplest of which is to use the PHP header () function to set the appropriate response header field and send a redirect (302) status code back to the user's browser.

Some of the kits produce the same page content for both search engine bots and users. In this scenario the redirection of users is achieved by active content embedded in the web page. The search engine bots will simply parse and index the raw page, most likely remaining oblivious to the redirect. Users on the other hand, will be redirected when the SEO page is loaded in their browser. For the more sophisticated of these kits, a check on the useragent string is used to identify requests from search engine bots, such that the active content for redirection is only added for requests from users. This is presumably to help evade detection and blacklisting by the search engines.

There are many ways to redirect the user using active content within the web page, the most common being JavaScript or ActionScript in embedded Flash content. A couple of examples used in recent SEO attacks are shown in Figure 6.



```
CWS.!...p.......
....C....?......
....this.javascr
ipt:eval(unescap
e('%77%69%6e%64%
6f%77%2e%6c%6f%6
[redacted]
%75%6d%65%6e%74%
2e%74%69%74%6c%6
5%2b%27%22%3b'))
.getURL...
..........
```

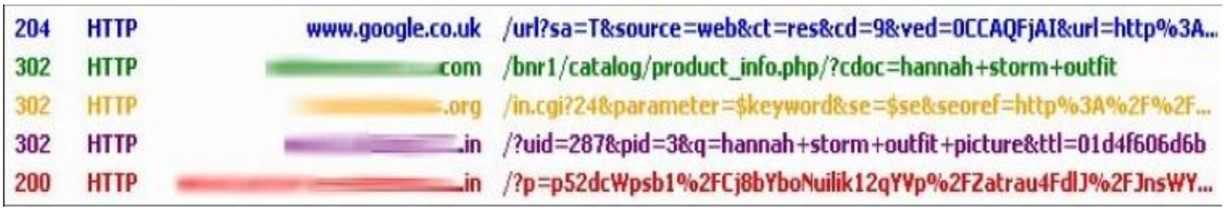
(A)

```
var str=["234", "249", "242", "231", "248", "237", "243", "242",
"164", "215", "233", "242", "232", "209", "253", "172", "245", "249",
"233", "246", "253", "173", "255", "142", "164", "251", "237", "242",
[redacted]
"243", "232", "233", "217", "214", "205", "199", "243", "241", "244",
"243", "242", "233", "242", "248", "172", "232", "243", "231", "249",
"241", "233", "242", "248", "178", "246", "233", "234", "233", "246",
"246", "233", "246", "173", "191", "142", "257"];
 var temp='';
 var gg='';
 for (i=0; i<str.length; i++){
 gg=str[i]-132;
 temp=temp+String.fromCharCode(gg);
 }
 eval(temp);
```

(B)

Figure 6 : Examples of redirection from within the SEO page. (A) Using ActionScript's getURL() within Flash content, and (B) using obfuscated JavaScript to redirect via window. Location.

It is normal for the URL of the SEO page to be included within the query string of the initial redirect URL, which enables the attackers to track the SEO pages that users are hitting. In most of the attacks analysed, there are multiple levels of redirection involved before the victim is exposed to the fake antivirus site. A log of the HTTP traffic in an example attack is shown in Figure 7.



```
204  HTTP            www.google.co.uk  /url?sa=T&source=web&ct=res&cd=9&ved=0CCAQFjAI&url=http%3A...
302  HTTP  [redacted]com  /bnr1/catalog/product_info.php/?cdoc=hannah+storm+outfit
302  HTTP  [redacted].org  /in.cgi?24&parameter=$keyword&se=$se&seoref=http%3A%2F%2F...
302  HTTP  [redacted].in  /?uid=287&pid=3&q=hannah+storm+outfit+picture&ttl=01d4f606d6b
200  HTTP  [redacted].in  /?p=p52dcWpsb1%2FCj8bYboNuilik12qYVp%2FZatrau4FdlJ%2FJnsWY...
```

Figure 7: HTTP traffic observed after clicking through to an SEO page from Google search results. Multiple 302 redirects are used, from the SEO page (green) and two other rogue sites (orange, purple) before the user is taken to the fake antivirus distribution site (red).

d. Seeding SEO poisoned pages:

Though the SEO poisoned pages may exist, they won't appear in search engine results until they have been indexed. In order for the search engine crawlers to index the page, they must first be aware that the page exists. This can be achieved by having another indexed page link to it.

In most of the kits, each SEO page links to a number of other SEO pages so the process of seeding them for search engines to crawl is essentially automated (links to the SEO pages on different compromised sites is shared via the C&C). This means that only the first created SEO page in an attack needs to be seeded, which can can be conveniently done via forms provided by the search engines.
In kits where they choose not to link to other sites compromised in the same attack, we've observed a couple of other methods being used.

- The root page of the compromised sites is modified to contain links to the SEO pages hosted within that site. These links use simple CSS tricks to make them invisible to users, but to the search engine crawlers the links are readily indexed.
- Links are posted on other legitimate sites (through various user input mechanisms, such as comments, social bookmarking and blogging).

e. Protecting against SEO attacks:

As for other webbased attacks, it is the combination of URL filtering and content inspection that provides the best protection for users against SEO attacks. Monitoring the currently active SEO attacks enables collection of the redirection URLs involved, which can then be appropriately blacklisted.

Detection-wise, protection can be provided by adding detection for the payload. As is typical for web hosted payloads, the binaries are frequently updated, and so a generic approach to detection is required. In addition to detection of the payload, analysis of each SEO attack can identify other components which may be detectworthy, to thwart the attack prior to the user ever getting to the payload. Examples of such components include:

- Active content used for redirection
- HTML elements/scripts used in malware distribution pages

Negative SEO is real. It is possible to damage, if not destroy, a site through the use of malicious backlinks and aggressive backlink spamming, here some steps to protect against.

❖ *Perform Regular Link Audits:*

Regular link audits are good practice for any business, but they can save your bacon if you're ever the victim of a negative SEO attack. Monitoring your link profile growth is hands down the best way to spot suspicious activity before it spirals out of control.

Most websites will enjoy graphs that look something like this:

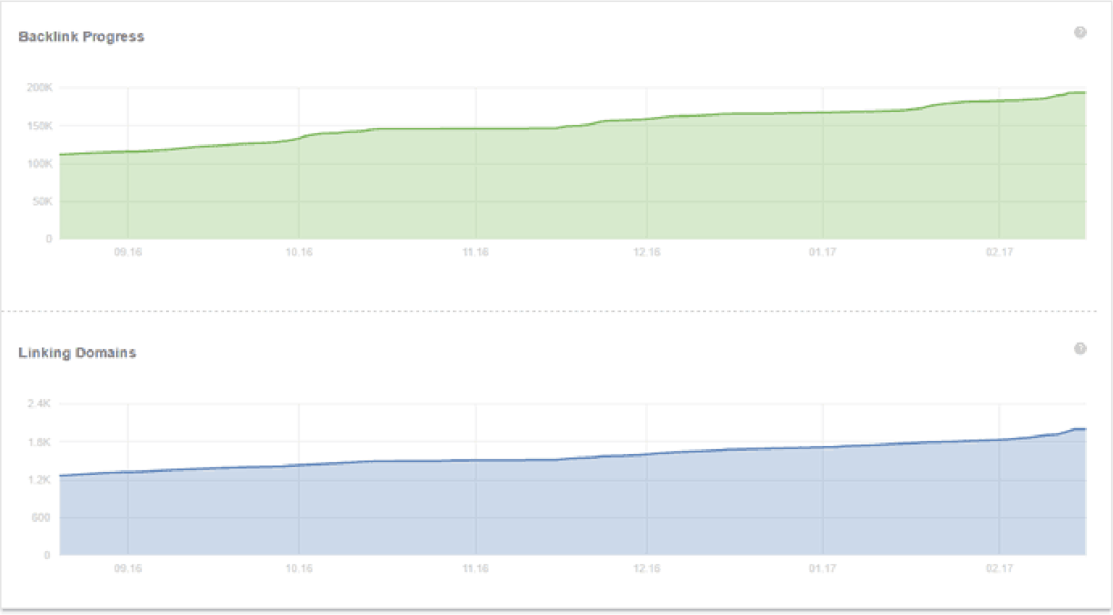


Figure 8 : THE Regular link audits analysis first look

However, if you suddenly notice a huge spike or drop and you haven't been working on link building, that should raise some red flags:

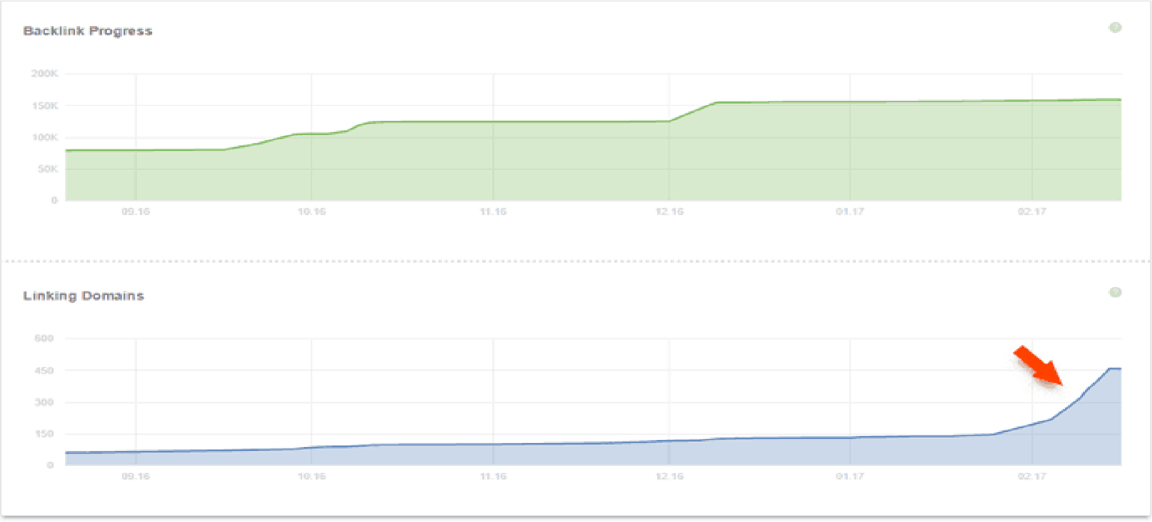


Figure 9 : unregular link audits suddenly

This is exactly what happened to Robert Neu, founder of WP Bacon, a WordPress podcast site. In 2014 he was the victim of link farm spam which gave him thousands of links with the anchor text “bad movie.” It cost him hundreds of visits, and he dropped 50 spots in ranking for one of his main keywords.

Fortunately, Neu was able to recover rankings and traffic lost as a result of the attack in relatively short order. Despite a continued barrage of spam, he was able to submit a disavow file listing the attacking domains

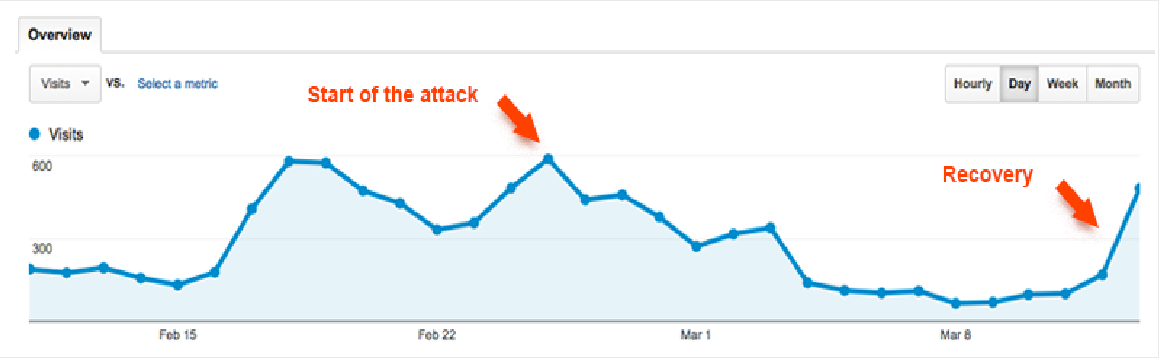
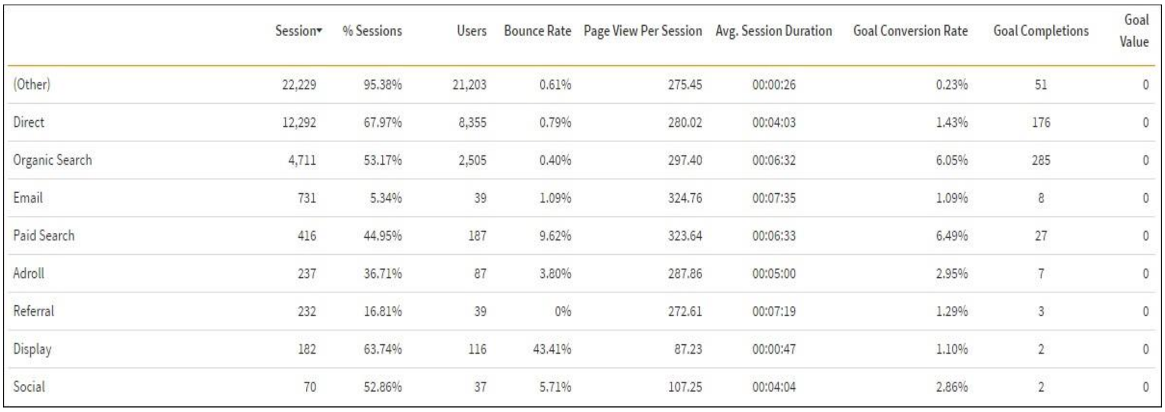


Figure 10 : The analysis of unregular link audits shown the start and recovery steps

❖ *Monitor Metrics Regularly:*

SEO specialists use different tools to monitor your metrics. **Siteoscope**, for instance, provides you an in-depth about your traffic, such as:

- Amount of traffic going to your site
- Locations where your traffic is coming fpm
- Devices used to access your website



	Session▾	% Sessions	Users	Bounce Rate	Page View Per Session	Avg. Session Duration	Goal Conversion Rate	Goal Completions	Goal Value
(Other)	22,229	95.38%	21,203	0.61%	275.45	00:00:26	0.23%	51	0
Direct	12,292	67.97%	8,355	0.79%	280.02	00:04:03	1.43%	176	0
Organic Search	4,711	53.17%	2,505	0.40%	297.40	00:06:32	6.05%	285	0
Email	731	5.34%	39	1.09%	324.76	00:07:35	1.09%	8	0
Paid Search	416	44.95%	187	9.62%	323.64	00:06:33	6.49%	27	0
Adroll	237	36.71%	87	3.80%	287.86	00:05:00	2.95%	7	0
Referral	232	16.81%	39	0%	272.61	00:07:19	1.29%	3	0
Display	182	63.74%	116	43.41%	87.23	00:00:47	1.10%	2	0
Social	70	52.86%	37	5.71%	107.25	00:04:04	2.86%	2	0

Figure 11 : Monitoring the Metrics

This not only helps your SEO specialist come up with a strategy for improved site optimization, but also helps them see any inconsistencies with specific metrics.

❖ *Keep an eye on your site speed :*

Site speed is a key ranking factor. If your website is becoming sluggish and you have no idea why you should use crawling software to look for anything suspicious.

If you can't find anything and there's still a problem, you might be the victim of forceful crawling. Forceful crawling causes a heavy server load, which means your site will slow down and might even crash.

If you think you're the victim of a crawling attack, you should contact your hosting company or webmaster to try and determine where the load is coming from. If you're tech savvy, you can also try to find the perpetrators yourself.

❖ *Keep Your Backlink Profile Clean :*

The most common tactic negative SEO attackers use on unsuspecting websites is sending them thousands of low-quality backlinks. A good SEO specialist knows how to use different tools to provide your website some form of leverage against these link building tactics.

❖ *Protect Your Links :*

Attackers will not just target any link pointing to your website – they'll go for the best ones that are driving value and traffic. The most common tactic they'll use to target high-quality links is by creating a fake email account and sending a request to remove your best links. To prevent this from happening, use an email address with your domain when sending requests to Google Webmasters. This will help you recover those removed high-quality links in case someone posed as you and made a fake link removal request.

❖ *Search for Scraped Content :*

Content marketing has been the name of the game these last few years, but not everyone is equally creative when it comes to content creation. Consequently, scraping has become all too frequent.

Scraping is the process of lifting content from your website and copying it verbatim to other websites. Usually, the attacker will claim it as their own in an attempt to beef up their thin content, but sometimes they'll combine it with a link farm attack to spam your site.

Scraping has serious consequences. If the copied content gets indexed before your content then your page might be devalued, and your site might fall in rank as a result.

Use a tool like Copyscape to discover if anyone's plagiarized your content. If they have, ask the webmaster to remove your content. If they refuse (or don't respond), report them by filling out Google's Copyright Removal form.

❖ *Watch Your Keywords' CTR :*

In late 2014, Bartosz Goralewicz experienced something strange — a client's site was getting thousands of hits that would land on the page and then immediately bounce. This began to have an effect on their rankings — user experience is an important signal, and this looked like bad UX.

What was actually happening was that someone had programmed a bot to target certain keywords, land on competitor sites, and then bounce, which created a false SERP bounce rate.

This insidious attack is difficult to spot if you aren't monitoring your keywords' CTR. Log in to Google Search Console, click **Search Traffic** > **Search Analytics**, and look at your CTR across all keywords. If you notice a large spike for no reason, contact Google and begin disavowing the offending links.

❖ *Check your SERP Ranking :*

You probably don't need to be told to check your SERP ranking from time to time, but just in case, here's a compelling reason why you should: A drop in rank might be the result of malicious intent. Complete de-indexing as a result of a hack doesn't often happen, thankfully, but I've heard horror stories of shady SEOs that changed a former clients' robots.txt file to say Disallow: / after they were let go.

It's hard to imagine the full consequences of de-indexing your website, but fortunately, Moz bit the bullet in late 2014 so that we don't have to wonder. They used Google's URL removal tool to remove Followerwonk from the web, and within 2-3 hours all Followerwonk URLs had practically disappeared from Google SERPs.

Of course, now that Penguin's getting refreshed in real-time, changes could happen much faster. That's good news if you're trying to recover from an attack, but it also means that victims who aren't paying attention could pay a severe price – and quickly.

For a full overview of your site's performance, use rank tracking software to monitor your visibility. If you notice sudden drops, check your website's crawl stats in Google Search Console and make sure your robots.txt is still set up properly.

❖ *Upgrade Your Security :*

Negative SEO might not be all that common, but cyber-attacks are on the rise year after year. Make sure your software is up to date, you apply all security patches to your software, and your CMS software is equipped with powerful encryption to protect your users.

You should also migrate your site to HTTPS, especially if you're in e-commerce or storing other sensitive customer data. Not only does HTTPS encryption offer you greater security, but it's also a ranking signal, and it might improve your SEO overall.

Cyber-attacks aren't technically negative SEO, but they will have an impact on your SEO. Google flags compromised sites with a "this site may be hacked" line into your search listing, which will definitely warn traffic away.

❖ *Malicious bots:*

Unfortunately, most malicious bots do not follow standard protocols when it comes to web crawlers. This obviously makes them harder to deter. Ultimately, the solution is dependent on the type of bot you're dealing with.

If you're concerned about content scrapers, you can manually look at your backlinks or trackbacks to see what sites are using your links. If you find that your content has been posted without your permission on a spam site, file a DMCA-complaint with Google.

In general, your best defence is to identify the source of your malicious traffic and block access from these sources.

The traditional way of doing this is to routinely analyse your log files through a tool like AWStats. This produces a report listing every bot that has crawled your website, the bandwidth consumed, total number of hits, and more.

Normal bot bandwidth usage should not surpass a few megabytes per month.

If this doesn't give you the data you need, you can always go through your site or server log files. Using this, specifically the 'Source IP address' and 'User Agent' data, you can easily distinguish bots from normal users.

Malicious bots might be more difficult to identify as they often mimic legitimate crawlers by using the same or similar User Agent.

If you're suspicious, you can do a reverse DNS lookup on the source IP address to get the hostname of the bot in question.

The IP addresses of major search engine bots should resolve to recognizable host names like '*.googlebot.com' or '*.search.msn.com' for Bing.

Additionally, malicious bots tend to ignore the robots exclusion standard. If you have bots visiting pages that are supposed to be excluded, this indicates the bot might be malicious.

❖ *WordPress plugins and Extensions :*

A huge number of compromised sites involve outdated software on the most commonly used platform and tools – WordPress and its CMS.

WordPress security is a mixed bag. The bad news is, hackers look specifically for sites using outdated plugins in order to exploit known vulnerabilities. What's more, they're constantly looking for new vulnerabilities to exploit.

This can lead to a multitude of problems. If you are hacked and your site directories have not been closed from listing their content, the index pages of theme and plugin related directories can get into Google's index. Even if these pages are set to 404 and the remaining site is cleaned up, they can make your site an easy target for further bulk platform or plugin-based hacking.

It's been known for hackers to exploit this method to take control of a site's SMTP services and send spam emails. This can lead to your domain getting blacklisted with email spam databases.

If your website's core function has any legitimate need for bulk emails – whether it's newsletters, outreach, or event participants – this can be disastrous.

How to prevent this

Closing these pages from indexing via robots.txt would still leave a telling footprint. Many sites are left removing them from Google’s index manually via the URL removal request form. Along with removal from email spam databases, this can take multiple attempts and long correspondences, leaving lasting damages.

On the bright side, there are plenty of security plugins which, if kept updated, can help you in your efforts to monitor and protect your site.

Popular examples include All in One and Sucuri Security. These can monitor and scan for potential hacking events and have firewall features that block suspicious visitors on a permanent basis.

Review, research, and update each plugin and script that you use. It’s better to invest the time in keeping your plugins updated than make yourself an easy target.

❖ *Local Network Security:*

It’s equally as important to manage your local security as it is that of the website you’re working on. Incorporating an array of layered security software is no use if access control is vulnerable elsewhere.

Tightening your network security is paramount, whether you’re working independently, remotely, or in a large office. The larger your network, the higher the risk of human error, while the risks of public networks cannot be understated.

Ensure you’re adhering to standard security procedures like limiting the number of logins attempts possible in a specific time-frame, automatically ending expired sessions, and eliminating form auto-fills.

Wherever you’re working, encrypt your connection with a reliable VPN.

It’s also wise to filter your traffic with a Web Application Firewall (WAF). This will filter, monitor, and block traffic to and from an application to protect against attempts at compromise or data exfiltration.

In the same way as VPN software, this can come in the form of an appliance, software, or as-a-service, and contains policies customized to specific applications. These custom policies will need to be maintained and updated as you modify your applications.

III. Conclusion:

In this paper i have provided insight into the current spate of **SEO attacks** by presenting the results of analysis into some of the kits being used to manage **the attacks**. Compromised legitimate websites provide a convenient network of hosts, that are being used as a platform for these attacks. By successfully poisoning search engine data, the attackers are able to lure unsuspecting users to malicious **SEO pages**, initiating the attack. To date, the attacks are being used for distribution of fake anti-virus malware, though the payloads could easily change in the future.

Similarly to how kits are used to automate and manage malicious content designed to exploit browser vulnerabilities, so too kits are being used to manage the SEO attacks. This facilitates setting up new attacks on different hosts, dynamically generating SEO pages stuffed with topical keywords and phrases for search engines to index. The kits can also provide a single point of control over the redirection URL used for **SEO attacks** hosted on multiple compromised sites.

Some of the kits provide functionality to automatically track the most popular search terms at any given time, in order to focus the attack on topical news and events, increasing the likelihood of **attracting user traffic**.

Malware distribution through **SEO attacks** could easily be described as beautiful in its simplicity.

A straightforward case of trickery, without the need for exploits or zero-day vulnerabilities. Just a case of tricking the search engines into indexing rogue SEO pages and then tricking users into running the fake anti-virus malware (and subsequently paying to register it). This simplicity should not detract from the success of such attacks however. And whilst the attacks continue to succeed, there is little need for the malware authors and distributors to change the formula.

http://www.google.com/support/webmasters/bin/answer.py?hl=en-uk&answer=34444 2

http://help.live.com/help.aspx?mkt=en-GB&project=wl_webmasters 3

http://help.yahoo.com/l/us/yahoo/search/basics/basics-18.html 4 http://www.sophos.com/blogs/sophoslabs/v/post/558 11

http://www.sophos.com/blogs/sophoslabs/v/post/250 12 http://www.sophos.com/blogs/sophoslabs/v/post/1329 13

http://www.sophos.com/blogs/sophoslabs/v/post/916 14 http://www.user-agents.org 15

https://addons.mozilla.org/en-US/firefox/tag/user%20agent 16 https://addons.mozilla.org/en-US/firefox/tag/referrer 17

http://en.wikipedia.org/wiki/Link_exchange 18 http://php.net/manual/en/book.curl.php 19

http://php.net/manual/en/function.fsockopen.php 20 http://www.google.com/trends/hottrends/atom/hourly 21

http://search.twitter.com/ 22 http://blekko.com 23 http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.30 24

http://www.bing.com/webmaster/SubmitSitePage.aspx 29 http://search.yahoo.com/info/submit.html 30

http://webmaster.yandex.ru/ 31 http://www.sophos.com/security/analyses/viruses-and-spyware/malfakeavjsa.html 32

http://packetstormsecurity.org/ 33 http://www.sophos.com/blogs/sophoslabs/post/363

https://www.academia.edu/6793761/Analysis_of_Searching_Algorithms_in_Web_Crawling