ISSN:0975-9646

Finding the Malicious URLs using Search Engines

Amruta Rajeev Nagaonkar [1], Umesh L. Kulkarni [2]

[1]*Department of Computer Science & Engineering, Shivaji University, Kolhapur*
[2]*Department of Computer Engineering, Mumbai University, Mumbai*

Abstract- Malicious URLs have been widely used to mount various cyber attacks including spamming, phishing and malware. Malicious URL is a URL created with malicious purposes, among them, to download any type of malware to the affected computer, to cause undesired effects. Such URL sites contains malicious code which describes a broad category of system security terms that includes attack scripts, viruses, worms, Trojan horses, backdoors, and malicious active content. Detection of malicious URLs and identification of threat types are critical to thwart these attacks. Detecting malicious URLs is now an essential task in network security intelligence. To maintain efficiency of web security, these malicious URLs have to be detected, identified as well as their corresponding links should be found out. Hence users get protected from it and effectiveness of network security gets increased.

For such identification there must be analyzer which should not only detect such URLs but analyze them also. And methods to detect corresponding links of malicious URLs .This approach will prevent the users from attacks and increase efficiency of web crawling phase.

Keywords: Malicious URLs, Analyzer, Web Security.

1. INTRODUCTION:

The World Wide Web and new communications technologies drive new opportunities for commerce; they inevitably create new opportunities for criminal actors as well. Today millions of rogue Web sites advance a wide variety of scams including marketing counterfeit goods, perpetrating financial fraud (e.g., “phishing”) and propagating malware (e.g., via “drive-by” exploits or social engineering). What all of these activities have in common is the use of the Uniform Resource Locator (URL) as a vector to bring Internet users into their influence.

The web is a very large place where new pages (both legitimate and malicious) are added at a daunting place. Attackers relentlessly scan for vulnerable hosts that can be exploited and leveraged to store malicious pages, which are than organized in complex malicious meshes to maximize the changes that a user will land on them. As a result, it is a challenging task to identify malicious pages as they appear on the web. However, it is critical to succeed at this task in order to protect web users.

As more users are connected to the Internet and conduct their daily activities electronically, computer users have become the target of an underground economy that infects hosts with malware or adware for financial gain. Unfortunately, even a single visit to an infected web site enables the attacker to detect vulnerabilities in the user’s applications and force the download a multitude of malware binaries. Frequently, this malware allows the adversary to gain full control of the compromised systems leading to the ex-filtration of sensitive information or installation of utilities that facilitate remote control of the host.

In most cases, a successful exploit results in the automatic installation of a malware binary, also called drive-by download. Such drive-by download attacks are caused by URLs that attempt to exploit their visitors and cause malware to be installed and run automatically.

The installed malware often enables an adversary to gain remote control over the compromised computer system and can be used to steal sensitive information such as banking passwords, to send out spam or to install more malicious executables over time. However, we have a situation closer to that of an earlier era where threats were propagated through diskettes, email attachments, innocuous-looking Trojans and drive by download attacks.

Such malware distribution on networks represents the hop-points used to lure users to the malware distribution site. By investigating these edges, so it can reveal a number of causal relationships that eventually lead to browser exploitation.

A serious security threat today is malicious executables, especially new, unseen malicious executables often arriving as email attachments. A malicious executable is defined to be a program that performs a malicious function, such as compromising a system’s security, damaging a system or obtaining sensitive information without the user’s permission. Such *malicious URL* is a URL created with malicious purposes, among them, to download any type of malware to the affected computer, which can be contained in spam or phishing messages, or even improve its position in search engines using Blackhat SEO techniques.

Malware is a common term for all types of malicious software, which in the area of computer security means Software which is used with the intention of violating a computer system’s security policy”. There are many other definitions for Malware, but all of them have some area in common in which Malware is malicious codes that has the potential to harm the machine or network on which it executes.

Attackers exploit vulnerabilities in web services, browsers and operating systems, or use social engineering techniques to make users run the malicious code in order to spread malwares. Malware authors use obfuscation techniques like dead code insertion, register reassignment, subroutine reordering, instruction substitution, code transposition, and code integration to evade detection by traditional defences like firewalls, antivirus and gateways which typically use signature based techniques and are unable to detect the previously unseen malicious executables. Commercial antivirus vendors are not able to offer immediate protection for zero day malwares as they need to analyze these to create their signatures.

In another typical Internet attack scenario, attackers set up a phishing website and lure unsuspecting users into entering sensitive information such as online banking credentials and credit card numbers. The phishing website often has the look and feel of the targeted legitimate website (e.g., an online banking service) and a domain name that sounds similar.

Current anti-virus systems attempt to detect these new malicious programs with heuristics generated by hand. This approach is costly and oftentimes ineffective. One of the primary problems faced by the virus community is to devise methods for detecting new malicious programs that have not yet been analyzed. Hence such malicious executables cannot be detected by anti-virus system.

The technology in current virus scanner has two parts: a signature-based detector and a heuristic classifier that detects new viruses. This kind of analysis can be time-consuming and oftentimes still fail to detect new malicious executables.

We designed a framework that used data mining algorithms to train multiple classifiers on a set of malicious and benign executables to detect new examples. Our goal in the evaluation of this method was to simulate the task of detecting new malicious executables.

In particular, we are not only interested in finding compromised (vulnerable), legitimate pages, but also malicious pages that are directly set up by attackers.

In this paper, we present a data-mining framework that detects new, previously unseen malicious executables accurately and automatically.

Using data mining methods, our goal is to automatically design and build a scanner that accurately detects malicious executables before they have been given a chance to run. Our framework uses *classifiers* to detect malicious executables. A classifier is a rule set, or detection model, generated by the data mining algorithm that was trained over a given set of training data.

Searching for malicious web pages is a three-step process, in which URLs are first collected, then examined in depth using specialized analyzers for classifying web pages or URLs as malicious and benign then last step is submitting these malicious URLs to different methods to search their corresponding links said to be as initial seed. To collect URLs, one typically uses web crawlers, which are programs traversing the web in a systematic fashion. Starting from a set of initial pages, this program follows hyperlinks to find as many (different) pages as possible.

To address Web-based attacks, a great effort has been directed towards detection of malicious URLs. A common countermeasure is to use a blacklist of malicious URLs, which can be constructed from various sources, particularly human feedbacks that are highly accurate yet time-consuming. The malware analysis techniques help the analysts to understand the risks and intensions associated with a malicious code sample.

Analytical method used in this paper, a classification model based on features which built with through machine learning.

2. Research Method:

This approach is going to be used to search the web more efficiently for pages that are likely to be malicious as well as their corresponding links.

In proposed work, analyzer classifies web pages into malicious and benign. This also classifies new malicious content added into web pages. To find out other corresponding pages different types of methods has been added for better classification. These collected pages or URLs will be stored in dataset by proxy server. This will avoid direct contact with search engine. Proxy server will act as firewall between user and malicious contents. The following figure indicates proposed system architecture.

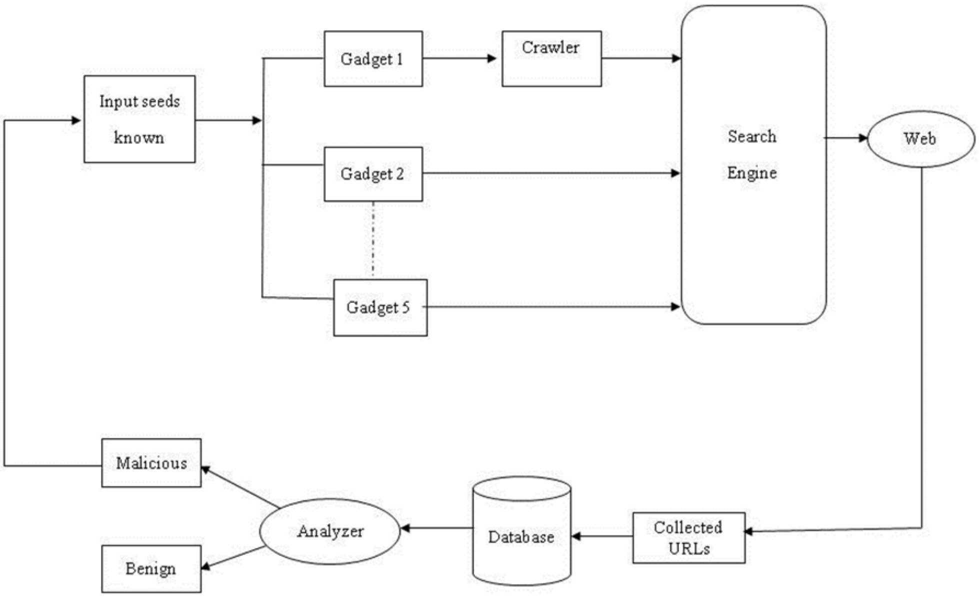

Fig1. Proposed System architecture

2.1 Developing Methods:

In this module, methods are developed. These methods are used to extract the information from initial seed of known malicious pages. These five methods perform the processing steps to guide the search for additional, malicious contents from search engine through web.

These methods are nothing but a technique to consume a feed of web pages and generate queries to search engine. The results returned by the search engine are then forwarded to an analysis infrastructure. The purpose of a method is to find candidate pages (URLs) that are based on the pages contained in the seed.

Following are five methods involved in first methods:

1. Links Method
2. Content Dorks Method
3. Search Engine Optimization Method
4. Domain Registration Method
5. DNS Queries Method

The working of all these methods is different and explained as below;

2.1.1 Links Method:-

A given input link as any suspicious URL analyzed by the links method consists of all the URLs of known suspicious link. This method has the linking structure rather than directly building the web graph by having access to the raw crawling data of the search engine.

2.1.2 CONTENT DORKS Method:-

Content dorks method searches the suspicious URLs by using known malicious keywords. These malicious keywords are taken from Google Hacking Database. All content dorks are submitted as queries to the search engine (Google). Then retrieved the URLs (links) from the results and later will submit them to the analyzer.

2.1.3 SEARCH ENGIENE OPTIMIZATION Method:-

The objective of search engine optimization (SEO) method is to identify a cloaking technique as well as links which show benign page and its corresponding cloaked page which is set up by attackers. This Cloaking is a search engine optimization (SEO) technique in which the content presented to the search engine spider is different from that presented to the user's browser. This is done by delivering content based on the IP addresses or the User-Agent HTTP header of the user requesting the page.

2.1.4 DOMAIN REGISTRATION Method:-

This method identifies suspicious sequences of domain registrations. These domains are then used to create URLs that are scheduled for analysis. The URL creation consists of taking the closest known malicious registration and replacing the domain with the suspicious domain that we have just flagged.

2.1.5 DNS QUERIES Method:-

This method traces the path of DNS requests to locate pages that lead to the domains. This recursive DNS refers to the process of having the DNS server itself to make queries to other DNS servers on behalf of the client who made the original request. It identifies the domain names of the web pages that are likely to lead to malicious pages.

2.2 Developing An Effective Analyzer For Classifying The Web Pages:

To analyze web pages, this method is based on URLs classification which is done by discovering the lexical and host-based properties of malicious URLs. To classify URLs, concept is based only on the relationship between URLs and the lexical and host-based features that characterized them. Finally machine learning classifier is used for analysis like Naive Bayes.

- **Features:**

2.2.1 Lexical features:

URLs of malicious sites tend to “look different” to users who see them. Hence URL parsed to retrieve hostname and path name for classification purpose. In this method, the detection model maintains two lists of URLs: a list of benign URLs and a list of malicious URLs.

Following are the properties included in lexical feature:

1. URL parsed to retrieve hostname and the path.
2. Delimiters used like (strings delimited by ‘.’, ‘/’, ‘?’, ‘.’, ‘=’, ‘-’ and ‘_’).

This method also include following properties which are real valued features:

I. Length of URL
II. Length of hyphen
III. Length of domain
IV. Domain name extension
V. Length of max-length in domain name
VI. Length of directory

After this, comparison takes place by calculating above properties of malicious as well as benign URLs with each other. It gives result as input URL is malicious or benign.

Naïve bayes theorem:

- Simple probabilistic classifiers based on applying Bayes theorem.
- Describes the probability of an event, based on conditions that might be related to the event.

2.2.2 Host-based features:

The following are properties of the hosts that identify by the hostname part of the URL. Some of these features are following:

1. Check for blacklist IP addresses.
2. Date of registration, update and expiration.
3. Check value of the time-to-live (TTL) for the DNS records associated with the hostname.
4. Geographical location of the IP address belongs.
5. Check for who is registrar and registrant.

This approach gives appropriate result for classifying URLs as either malicious or benign based on both lexical and host-based features.

REFERENCES:-

[1] Luca Invernizzi, Santa Barbara,Stefano Benvenuti ,Paolo Milani, Comparetti Lastline, Vienna,“EVILSEED: A Guided Approach to Finding Malicious Web Pages,” in IEEE Symposium on Security and Privacy 20-23 May 2012, pp 428 – 442.

[2] M. A. Rajab, L. Ballard, P. Mavrommatis, N. Provos, and X. Zhao, “The Nocebo Effect on the Web: An Analysis of Fake Anti-Virus Distribution,” in USENIX Workshop on Large-Scale Exploits and Emergent Threats, 2010.

[3] S. Ford, M. Cova, C. Kruegel, and G. Vigna, “Analyzing and Detecting Malicious Flash Advertisements,” in Annual Computer Security Applications Conference (ACSAC), 2009.

[4] N.Provos, D. McNamee, P. Mavrommatis, K. Wang, and N. Modadugu, “The Ghost in the Browser: Analysis of Webbased Malware,” in USENIX Workshop on Hot Topics in Understanding Botnet, Vol.142, 2007, pp.122-136.

[5] J. John, F. Yu, Y. Xie, M. Abadi, and A. Krishnamurthy, “Searching the Searchers with SearchAudit,” in USENIX Security Symposium, 2010.

[6] T. Moore and R. Clayton, “Evil Searching: Compromise and Re-compromise of Internet Hosts for Phishing,” in International Conference on Financial Cryptography and Data Security, 2008.