Literature Survey on Malicious URL Identification

Arya Babu
Department of Computer Science and Engineering.
Sree Buddha College of Engg. For Women.
Pathanamthitta, Kerala, INDIA

Abstract- We have always experienced malicious attack knowingly or unknowingly. Some of them would be visible, easily identifiable while others may not. Some commonly found attacks are Drive by download in which malicious files automatically get downloaded in the user's system. The effectiveness of an attack depends on the type of attack. Cybercrimes reported till now will always be the best references for understanding the after attacks of software attacks. Fresh measures are discovered to detect wide range of malicious attacks that spread via web and prevent users from accessing them. The malicious URL's are inspected by crawlers and steps are taken to block such sites and applications. But still as long as web exists there will always be chances for vulnerability. Malicious links are used as source to the distribution channels to propagate malware all over the Web. This paper introduces various identification associated with the URL (Uniform Resource Locator) classification to identify whether the target website is a malicious or benign.

Keywords— Drive-by-download, Evilseed, FakeAV, Keyloggers, Botnets

I. INTRODUCTION

Now a days, Internet is an essential part of the day to day life of many people. Most of the person's choice is to search information, conduct business and enjoy entertainment on the Internet. At the same time internet has become the primary platform used by miscreants to attack users. Attackers that use malicious websites to install malware programs by exploiting browser vulnerabilities. Malicious web content has become one of the most effective mechanisms for cyber criminals to distribute malicious code. Attackers frequently use drive-by-download exploits to compromise a large number of users. In a drive-by-download attack, the attacker first crafts malicious client-side scripting code that targets vulnerability in a web browser or in one of the browser plugins.

Web browsers play an important role in allowing users to easily interact with the World Wide Web by traversing, retrieving and finally presenting the related topics to them. While, one may say the Internet is a powerful resource to gain knowledge, yet the Internet has another side as well. Many people benefit from using the Internet since they can simply access huge amount of information in little time. This is one of the strongest advantages of internet. However, internet is a reflection of containing both good and bad impacts. The most important issues in using the Internet are related to the user's security. Although for user security various concepts are defined and they might have different levels of obtaining it, yet one common aspect between all is *how to provide it*, especially while they are using online services. Without security, the user might or even might not encounter a threat which can somehow result in gaining access to the user's belongings without his notice and thus, allows the attack to support his system or to simply carry out another different kind of attack that result in losing everything which possesses great value like bank accounts. One type of attack among various existing attacks is malware which is installed and spread easily.

Before using a particular URL if one could inform users that it was dangerous to visit, much of this problem could be solved. To solve these problems the security community has developed blacklisting services, appliances and search engines that provide accurate feedback. The blacklists are particularly human feedbacks that are highly accurate yet time consuming.

Blacklisting is effective only for known malicious URLs. Predictably, many malicious sites are not blacklisted either because they are too new, were never evaluated, or were evaluated incorrectly.

The rest of the paper is organized as follows. Literature survey are explained in section II. Experimental results are presented in section III. Concluding remarks are given in section IV.

II. LITERATURE SURVEY

A .The Ghost in the Browser

N. Provos et al [1] introduced an overview of the current state of malware on the web. The evaluation is based on Internet wide measurements conducted over a period of twelve months starting March 2006. The results reveal several attack strategies for turning web pages into malware infection vectors. They identified four different aspects of content control responsible for enabling browser exploitation: advertising, third-party widgets, user contributed content and web server security. Through analysis and examples, the paper show how each of these categories can be used to exploit web browsers.

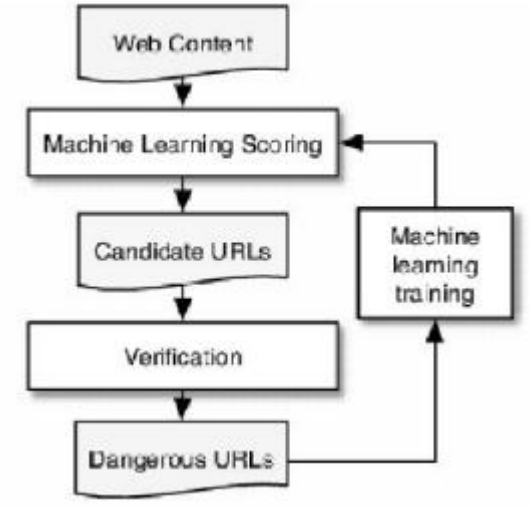

Fig 1: Data Flow in Phoney Pot

B. All your iFrames Points to Us

The fact that malicious URLs that initiate drive-by downloads are spread far and wide raises concerns regarding the safety of browsing the Web. However, to date, little is known about the specifics of this increasingly common malware distribution technique. P. Mavrommatis et al [2] attempt to fill in the gaps about this growing phenomenon by providing a comprehensive look at the problem from several perspectives. This study uses a large scale data collection infrastructure that continuously detects and monitors the behavior of websites that perpetrate drive-by downloads. In-depth analysis of over 66 million URLs (spanning a 10 month period) reveals that the scope of the problem.

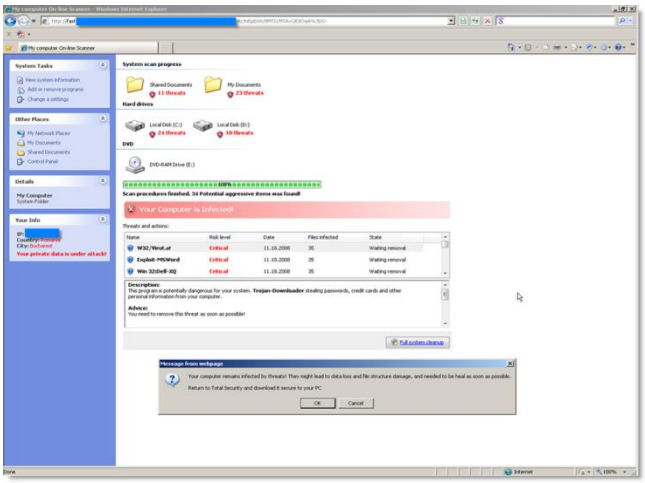

Fig 2: Example of social engineering technique

C. PhoneyC: A Virtual Client Honey pot

J Nazario et al [3] published this paper in 2009. This paper discusses about a virtual honey pot namely PhoneyC which is implemented to study the nature of malicious attack by making itself vulnerable to attack. These systems are instrumented to discover what happened and how PhoneyC mimic the behavior of a user driven network client application such as a web browser and be exploited by an attacker's content.

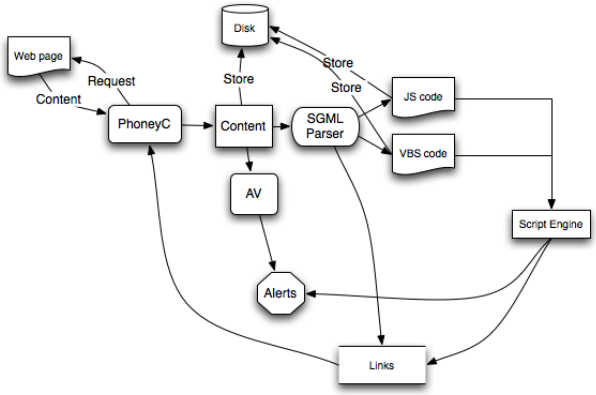

Fig 3 : Data Flow in Phoney Pot

D. The Nocebo effect on the web: An analysis of fake antivirus distribution

The paper proposed by M. A. Rajab et al [4] in the year 2010. The paper embeds the idea of an emerging malware. Fake AV attacks attempt to convince users that their computer systems are infected and offer a free download to scan for malware. Fake AVs pretend to scan computers and claim to find infected files—files which may not even exist or be compatible with the computer's OS. Users are forced to register the Fake Antivirus program for a fee in order to make the fake warnings disappear. Many users fall victim to these attacks and pay to register the Fake AV. To add insult to injury, Fake Antivirus are bundled with other malware, it remains on a victim's computer regardless of whether a payment is made. More recent Fake AV sites have evolved to use complex JavaScript to mimic the look and feel of the Windows user interface.

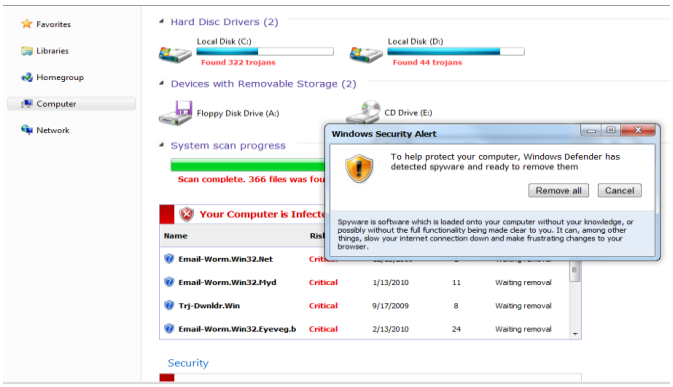

Fig 4: Fake antivirus distribution

The Fake AV detects even the operating system version running on the target machine and adjusts its interface to match. They use Google's malware detection infrastructure to discover Web sites that distribute Fake AV software. Briefly, that system uses machine learning to identify potentially malicious Web pages from Google's Web repository. Each page that is flagged by the screening process is further examined by navigating to it with an unpatched Windows virtual machine running an unpatched version of Internet Explorer. Detection algorithms use signals derived from state changes on the virtual machine, network activity, and scanning results of a group of licensed antivirus engines to decide definitively whether a page is malicious.

One of the algorithms is designed to complement the licensed AV engines to specifically detect social engineering attacks, including Fake AV attacks. They do not disclose the details of the detection algorithm, due to the highly adversarial nature of this field. This algorithm is currently used to protect hundreds of millions of Web users from Fake AV attacks and disclosing it may jeopardize this effort.The goal of this paper is to understand Fake AV distribution on internet by three levels. The first is measure the prevalence of Fake AV over time in absolute terms. Second, understand the network characteristics of domains that host Fake AV. Finally, explore the characteristics of Fake AV

E. Detection and Analysis of Drive-by-Download Attacks and Malicious JavaScript Code

The paper was proposed by M.Cova et al [5] in the year 2010. This paper presents a novel approach to the detection and analysis of malicious JavaScript code. The approach combines anomaly detection with emulation to automatically identify malicious Java- Script code and to support its analysis. They developed a system that uses a number of features and machine-learning techniques to establish the characteristics of normal JavaScript code. During detection, the system is able to identify anomalous JavaScript code by emulating its behavior and comparing it to the established profiles. To identifying malicious code, this system is able to support the analysis of obfuscated code and to generate detection signatures for signature-based systems. This system has been made publicly available and has been used by thousands of analysts.

F. BLADE : An Attack-Agnostic Approach for preventing Drive By Malware Infections

The paper was proposed by L. Lu, V. Yegneshvaran et al [6]. in the year 2010. Web-based surreptitious malware infections (i.e., drive-by downloads) have become the primary method used to deliver malicious software onto computers across the Internet. To address this threat, they present a browser independent operating system kernel extension designed to eliminate drive by malware installations. The BLADE (Block All Drive-by download Exploits) system asserts that all executable files delivered through browser downloads must result from explicit user consent and transparently redirects every unconsented browser download into a non-executable secure zone on disk.

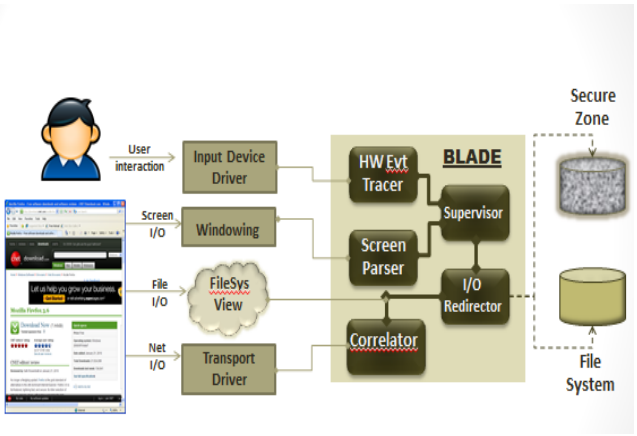

Fig 5: BLADE architecture

BLADE thwarts the ability of browser-based exploits to surreptitiously download and execute malicious content by remapping to the file system only those browser downloads to which a programmatically inferred user-consent is correlated, BLADE provides its protection without explicit knowledge of any exploits and is thus resilient against code obfuscation and zero-day threats that directly contribute to the pervasiveness of today's drive by malware. They present the design of the BLADE prototype implementation for the Microsoft Windows platform, and report results from an extensive empirical evaluation of its effectiveness on popular browsers. The evaluation includes multiple versions of IE and Firefox, against 1,934 active malicious URLs, representing a broad spectrum of web-based exploits now plaguing the Internet. BLADE successfully blocked all drive-by malware install attempts with zero false positives and a 3% worst-case performance cost.

G. EVILSEED: A Guided Approach For Finding Malicious Web page

EVILSEED focuses its searches "near" known malicious pages. [7] EVILSEED implements different techniques to extract from a page features that characterize its malicious nature pages with similar values for such features are also likely to be malicious. Then, by using the features extracted from an evil seed and by leveraging existing search engines. EVILSEED guides its search to the neighborhood around known malicious pages. The notion of "maliciousness" is used in a broad sense, and the general techniques are independent of the exact type of threat that a particular page constitutes. In the current version of EVILSEED, consider a malicious a page that, when any one visited, leads to the execution of a drive-by download exploit (possibly after redirecting the user). Consider a page to be malicious when it attempts to trick a user into installing a fake anti-virus program. In this paper, the term web page and URL are used synonymously. The actual inputs to and the "unit of analysis" of the system are URLs. In most cases, a page is uniquely identified by its corresponding URL. However, there are cases in which attackers create many URLs that all point to the same, underlying malicious page.

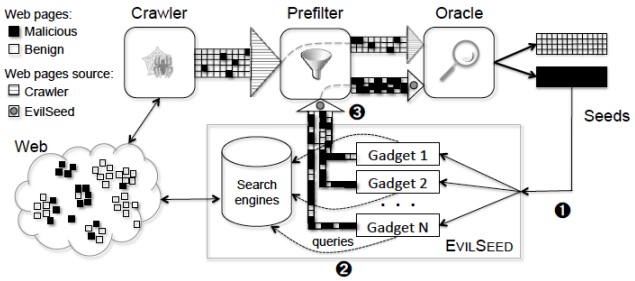

Fig 6: Evilseed Overview

Table 1: Comparison of Studied Paper

Paper Name	Published Year	Features
The Ghost In The Bowser	2007	• Identifies areas where vulnerability occurs. • Identifies the path and procedure for a malicious attack.
All your iFrames point to us	2008	• Mentions in detail about Drive By Download sattack, Identification and procedure steps
PhoneyC: A virtual client Honeypot	2009	• Implements a virtual client Honeypot and with this malicious attacks are identified
The Nocebo effect on the web: An analysis of fake antivirus distribution	2010	• **FV** Distribution is analyzed and an approach is implemented to identify such attacks.
Detection and Analysis of Drive-by-Download Attacks and Malicious JavaScript Code	2010	• Designs an approach to identify malicious attacks and vulnerable JavaScript code
BLADE : An Attack-Agnostic Approach for preventing Drive By Malware Infections	2010	• A browser independent OS extension is designed to deal block drive by downloads
EVILSEED: A Guided Approach For Finding Malicious Web Page	2012	• Starts from an initials seed of known, malicious web pages. • Identify malicious web pages more efficiently.

III.TYPES OF MALWARE

The Malware has been categorized into different types, the main and most common categories as follows:

Virus: A virus is a malicious program propagates from one program to another or from one computer to another by inserting their code into other program. Viruses attach themselves to a program such as executable file and its self-replicating capability spread the infection from one computer to another. It can cause denial of service and performance degradation.

Worm: Worms are standalone malicious software that can operate independently and don't hook itself to propagate. They exploit the security vulnerability by using computer or network resources and spread themselves via storage devices such as USB devices, communication media such as Email. It can cause network performance issues and consume large amount of memory of systems resources.

Trojan horse: this kind gives power to remote hijackers, to use your system as they wish. It may get your passwords, observe your systems or damage the system files, mask themselves by appearing to be something legitimate. Malignant piece of software that conceals itself and behaves as a legitimate program to takes unauthorized control of the computer. Trojan does not self-replicate instead downloaded through user interaction such as downloading a file from the internet. Steal password or login details. Electronic money theft Modify/delete files Monitor user activity.

Rootkit: Rootkits are the masking techniques for malware, basically designed to conceal the malicious intent of the program from the antivirus removal programs. It can be installed through a software exploit or by a Torjan.

Spyware: A software negatively affect a system by keeping track of user's activity without their consent and send back thesensitive information to creator. Spyware can be installed with other software such as freeware or dropped by Trojans. Sophisticated type of spyware captures entire network interface, digital certificate, encryption keys and other sensitive information.

Keyloggers: Serious form of Spyware secretly record keystrokes, read cookies and files on the drive to gather personal details. It can be installed by another malicious program or when a user visited a infected site. It capture sensitive information such as username, password, credit card number or online banking details.

Botnet: A botnet is remotely controlled software – collection of autonomous software robots. This kind of malware controls your systems remotely and sends spam or spyware. Botnet is usually a zombie program under common control on public and private network infrastructure. It doesn't sit around on machine waiting for the instruction from a third party instead it looks for the communication with similar instances of bots awaiting instructions. Many of botnets are zombie and wait for

command of the party who runs it. There are two types of botnet such as, simple or hierarchical simplest bot configuration is where the bots are connected to single central hub. The next configuration is hierarchical structure where bot master connects to hundreds of bots which in turn is connected to many bots.

IV.CONCLUSION

Malware is a threat to user's computer system in terms of stealing confidential information, corrupting or disabling security system. This survey paper presents various types of malware and some existing technologies used by security researchers.

REFERENCES

[1] N. Provos, D. McNamee ET AL "The Ghost in the Browser: Analysis of Web based Malware," USENIX Workshop on Hot Topics in Understanding Botnet, 2007.

[2] N. Provos, P. Mavrommatis et al, "All Your iFrames Point to Us," USENIX,2008

[3] J. Nazario, "PhoneyC: A Virtual Client Honeypot," in USENIX Workshop on Large-Scale Exploits and Emergent Threats, 2009.

[4] M. A. Rajab, L. Ballard, "The Nocebo Effect on the Web: An Analysis of Fake Anti-Virus Distribution," in USENIX Workshop on Large-Scale Exploits and Emergent Threats, 2010.

[5] M. Cova, C. Kruegel, and G. Vigna,"Detection and Analysis of Drive-by-Download Attacks and Malicious JavaScript Code," in International World Wide Web Conference (WWW), 2010.

[6] Long Lu. Vinod Yegneswaran Phillip Porrasz Wenke Leeyy College of Computing, Georgia Institute of Technology zSRI International "BLADE : An Attack-Agnostic Approach for preventing Drive By Malware Infections",2010.

[7] Luca Invernizzi .Stefano Benvenuti "Evilseed: A guided approach for finding malicious webpages," in IEEE Security Symposium, 2012 .