**LITERATURE SURVEY**

**1)** Detecting malicious websites with low-interaction honeyclients

**AUTHORS:**  A. Ikinci, T. Holz, and F. Freiling. Monkey-spider:

: Client-side attacks are on the rise: malicious websites that exploit vulnerabilities in the visitor’s browser are posing a serious threat to client security, compromising innocent users who visit these sites without having a patched web browser. Currently, there is neither a freely available comprehensive database of threats on the Web nor sufficient freely available tools to build such a database. In this work, we introduce the Monkey-Spider project [Mon]. Utilizing it as a client honeypot, we portray the challenge in such an approach and evaluate our system as a high-speed, Internetscale analysis tool to build a database of threats found in the wild. Furthermore, we evaluate the system by analyzing different crawls performed during a period of three months and present the lessons learned.

**2)**A guided approach to finding malicious web pages

**AUTHORS:** L. Invernizzi, S. Benvenuti, M. Cova, P. M. Comparetti, C. Kruegel, and G. Vigna. Evilseed

Malicious web pages that use drive-by download attacks or social engineering techniques to install unwanted software on a user's computer have become the main avenue for the propagation of malicious code. To search for malicious web pages, the first step is typically to use a crawler to collect URLs that are live on the Internet. Then, fast prefiltering techniques are employed to reduce the amount of pages that need to be examined by more precise, but slower, analysis tools (such as honey clients). While effective, these techniques require a substantial amount of resources. A key reason is that the crawler encounters many pages on the web that are benign, that is, the "toxicity" of the stream of URLs being analyzed is low. In this paper, we present EVILSEED, an approach to search the web more efficiently for pages that are likely malicious. EVILSEED starts from an initial seed of known, malicious web pages. Using this seed, our system automatically generates search engines queries to identify other malicious pages that are similar or related to the ones in the initial seed. By doing so, EVILSEED leverages the crawling infrastructure of search engines to retrieve URLs that are much more likely to be malicious than a random page on the web. In other words EVILSEED increases the "toxicity" of the input URL stream. Also, we envision that the features that EVILSEED presents could be directly applied by search engines in their prefilters. We have implemented our approach, and we evaluated it on a large-scale dataset. The results show that EVILSEED is able to identify malicious web pages more efficiently when compared to crawler-based approaches.

**3)** Blog identification and splog detection.

**AUTHORS:** P. Kolari, T. Finin, and A. Joshi. Svms for the blogosphere:

Weblogs, or blogs have become an important new way to publish information, engage in discussions and form communities. The increasing popularity of blogs has given rise to search and analysis engines focusing on the 'blogosphere'. A key requirement of such systems is to identify blogs as they crawl the Web. While this ensures that only blogs are indexed, blog search engines are also often overwhelmed by spam blogs (splogs). Splogs not only incur computational overheads but also reduce user satisfaction. In this paper we first describe our experiments on blog identification using Support Vector Machines (SVM). We compare results of using different feature sets and introduce new features for blog identification. We then report preliminary results on splog detection and identify future work.

**4)** Phishdef: Url names say it all.

**AUTHORS:** A. Le, A. Markopoulou, and M. Faloutsos.

Phishing is an increasingly sophisticated method to steal personal user information using sites that pretend to be legitimate. In this paper, we take the following steps to identify phishing URLs. First, we carefully select lexical features of the URLs that are resistant to obfuscation techniques used by attackers. Second, we evaluate the classification accuracy when using only lexical features, both automatically and hand-selected, vs. when using additional features. We show that lexical features are sufficient for all practical purposes. Third, we thoroughly compare several classification algorithms, and we propose to use an online method (AROW) that is able to overcome noisy training data. Based on the insights gained from our analysis, we propose PhishDef, a phishing detection system that uses only URL names and combines the above three elements. PhishDef is a highly accurate method (when compared to state-of-the-art approaches over real datasets), lightweight (thus appropriate for online and client-side deployment), proactive (based on online classification rather than blacklists), and resilient to training data inaccuracies (thus enabling the use of large noisy training data).

**5)** The core of the matter: Analyzing malicious traffic in cellular carriers.

**AUTHORS:** C. Lever, M. Antonakakis, B. Reaves, P. Traynor, and W. Lee

Much of the attention surrounding mobile malware has focused on the in-depth analysis of malicious applications. While bringing the community valuable information about the methods used and data targeted by malware writers, such work has not yet been able to quantify the prevalence with which mobile devices are actually infected. In this paper, we present the first such attempt through a study of the hosting infrastructure used by mobile applications. Using DNS traffic collected over the course of three months from a major US cellular provider as well as a major US noncellular Internet service provider, we identify the DNS domains looked up by mobile applications, and analyze information related to the Internet hosts pointed to by these domains. We make several important observations. The mobile malware found by the research community thus far appears in a minuscule number of devices in the network: 3,492 out of over 380 million (less than 0.0009%) observed during the course of our analysis. This result lends credence to the argument that, while not perfect, mobile application markets are currently providing adequate security for the majority of mobile device users. Second, we find that users of iOS devices are virtually identically as likely to communicate with known low reputation domains as the owners of other mobile platforms, calling into question the conventional wisdom of one platform demonstrably providing greater security than another. Finally, we observe two malware campaigns from the upper levels of the DNS hierarchy and analyze the lifetimes and network properties of these threats. We also note that one of these campaigns ceases to operate long before the malware associated with it is discovered suggesting that network-based countermeasures may be useful in the identification and mitigation of future threats