International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2020): 7.803

Detecting and Blocking of Malicious URL

Muskan V. Jaiswal[1], Anjali B. Raut[2]

[1]M.E. Scholar, Department of Computer Science and Engineering, H.V.P.M. C.O.E.T.
muskanjaiswal7697[at]gmail.com

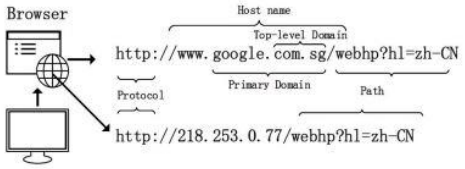
[2]H.O.D, Department of Computer Science and Engineering, H.V.P.M. C.O.E.T.
anjali_dahake[at]rediffmail.com

Abstract: ***Malicious URL, a.k.a. malicious website, is a common and serious threat to cyber security. Malicious URLs host unsolicited content (spam, phishing, drive-by downloads, etc.) and lure unsuspecting users to become victims of scams (monetary loss, theft of private information, and malware installation), and cause losses of billions of dollars every year. It is imperative to detect and act on such threats in a timely manner. Traditionally, this detection is done mostly through the usage of blacklists. However, blacklists cannot be exhaustive, and lack the ability to detect newly generated malicious URLs. To improve the generality of malicious URL detectors, machine learning techniques have been explored with increasing attention in recent years. This article aims to provide a comprehensive survey and a structural understanding of Malicious URL Detection techniques using machine learning. We present the formal formulation of Malicious URL Detection as a machine learning task, and categorize and review the contributions of literature studies that addresses different dimensions of this problem (feature representation, algorithm design, etc.). Further, this article provides a timely and comprehensive survey for a range of different audiences, not only for machine learning researchers and engineers in academia, but also for professionals and practitioners in cyber security industry, to help them understand the state of the art and facilitate their own research and practical applications. We also discuss practical issues in system design, open research challenges, and point out important directions for future research.***

Keywords: URL; malicious URL detection; feature extraction; feature selection; machine learning

1. Introduction

The advent of new communication technologies has had tremendous impact in the growth and promotion of businesses spanning across many applications including online-banking, e-commerce, and social networking. In fact, in today's age it is almost mandatory to have an online presence to run a successful venture. As a result, the importance of the World Wide Web has continuously been increasing. Unfortunately, the technological advancements come coupled with new sophisticated techniques to attack and scam users. Such attacks include rogue websites that sell counterfeit goods, financial fraud by tricking users into revealing sensitive information which eventually lead to theft of money or identity, or even installing malware in the user's system. There are a wide variety of techniques to implement such attacks, such as explicit hacking attempts, drive-by download, social engineering, phishing, watering hole, man-in-the middle, SQL injections, loss/theft of devices, denial of service, distributed denial of service, and many others. Considering the variety of attacks, potentially new attack types, and the innumerable contexts in which such attacks can appear, it is hard to design robust systems to detect cyber-security breaches. The limitations of traditional security management technologies are becoming more and more serious given this exponential growth of new security threats, rapid changes of new IT technologies, and significant shortage of security professionals. Most of these attacking techniques are realized through spreading compromised URLs (or the spreading of such URLs forms a critical part of the attacking operation [81]. URL is the abbreviation of Uniform Resource Locator, which is the global address of documents and other resources on the World Wide Web. A URL has two main components : (i) protocol identifier (indicates what protocol to use) (ii) resource name (specifies the IP address or the domain name where the resource is located). The protocol identifier and the resource name are separated by a colon and two forward slashes, eg Fig 1



Compromised URLs that are used for cyber attacks are termed as malicious URLs. In fact, it was noted that close to one-third of all websites are potentially malicious in nature [106], demonstrating rampant use of malicious URLs to perpetrate cyber-crimes. A Malicious URL or a malicious web site hosts a variety of unsolicited content in the form of spam, phishing, or drive-by download in order to launch attacks. Unsuspecting users visit such web sites and become victims of various types of scams, including monetary loss, theft of private information (identity, credit-cards, etc.), and malware installation. Popular types of attacks using malicious URLs include: Drive-by Download, Phishing and Social Engineering, and Spam [134]. Drive-by download [43] refers to the (unintentional) download of malware upon just visiting a URL. Such attacks are usually carried out by exploiting vulnerabilities in plugins or inserting malicious code through JavaScript. Phishing and Social Engineering attacks [73] trick the users into revealing private or sensitive information by pretending to be genuine web pages. Spam is the usage of unsolicited messages for the purpose of advertising or phishing. These attacks occur in large numbers and have caused billions of dollars' worth of damage, some even exploiting natural disasters [182]. Effective systems to detect such malicious URLs in a timely manner can greatly help to counter large number of and a variety of cyber-security threats. Consequently, researchers

and practitioners have worked to design effective solutions for Malicious URL Detection

2. Literature Review & Related work

After the training data is collected, the next step is to extract informative features such that they sufficiently describe the URL and at the same time, they can be interpreted mathematically by machine learning models. For example, simply using the URL string directly may not allow us to learn a good prediction model (which in some extreme cases may reduce the prediction model to a blacklist method). Thus, one would need to extract suitable features based on some principles or heuristics to obtain a good feature representation of the URL. This may include lexical features (statistical properties of the URL string, bag of words, n-gram, etc.), host-based features (WHOIS info, geo-location properties of the host, etc.), etc. These features and other features that are used in for this task will be discussed in much greater detail in this survey. These features after being extracted have to be processed into a suitable format (e.g. a numerical vector), such that they can be plugged into an off-the-shelf machine learning method for model training. The ability of these features to provide relevant information is critical to subsequent machine learning, as the underlying assumption of machine learning (classification) models is that feature representations of the malicious and benign URLs have different distributions. Therefore, the quality of feature representation of the URLs is critical to the quality of the resulting malicious URL predictive model learned by machine learning.

Using the training data with the appropriate feature representation, the next step in building the prediction model is the actual training of the model. There are plenty of classification algorithms can be directly used over the training data (Naive Bayes, Support Vector Machine, Logistic Regression, etc.). However, there are certain properties of the URL data that may make the training difficult (both in terms of scalability and learning the appropriate concept). For example, the number of URLs available for training can be in the order of millions (or even billions). As a result, the training time for traditional models may be too high to be practical. Consequently, Online Learning [77, 78], a family of scalable learning techniques have been heavily applied for this task. Another challenge is the sparsity of the bag-of-words (BoW) feature representation of the URLs. These features indicate whether a particular word (or string) appears in a URL or not - as a result every possible type of word that may appear in any URL becomes a feature. This representation may result in millions of features which would be very sparse (most features are absent most of the time, as a URL will usually have very few of the millions of possible words present in it). Accordingly, a learning method should exploit this sparsity to improve learning efficiency and efficacy. There are other challenges which are specifically found for this task, and have warranted appropriate contributions in machine learning methodology to alleviate these challenges.

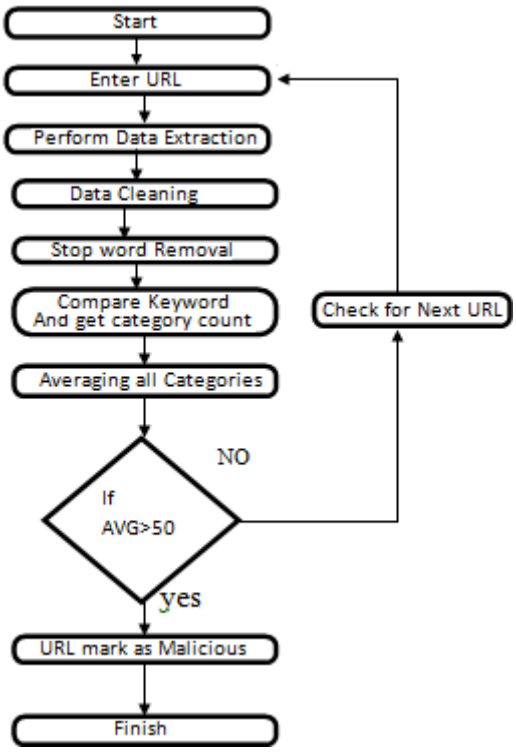
3. Problem Statement

In existing there are various types of approaches all determination done on the various kind of processes in which some part of web pages consider originating root considers various type of data downloading protocols check but the actual checking will fails lot of times so that it is necessary to define a framework which will efficient in determination of malicious urls.

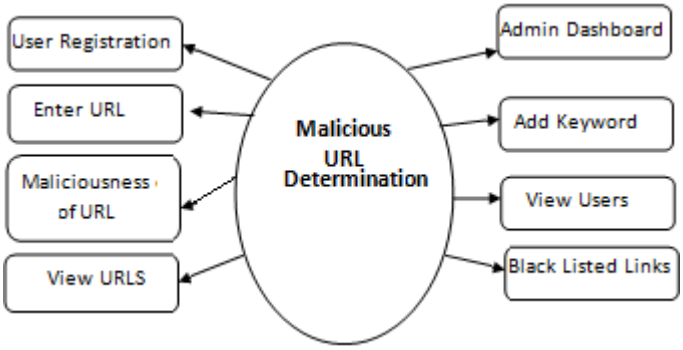
4. Proposed Work and Objectives

As per the survey seen it is necessary to develop evolutionary approach for the determination of malicious url so that we are proposed a mechanism which will work on web page data evaluation in which the content extraction will be perform and sentimental analysis will be work out. So that we proposed an evolutionary based framework for malicious url determination using content evaluation In the proposed the actual content of the web page extracted using web crawling in which the data evaluate and send to sentimental analytic tool which determine the malicious calculation. As per the results determine which can be latterly evaluated and black listed urls determine.

5. Methodology



6. Detection of Malicious URLs



7. Conclusion

In the proposed the web crawling based web page mining perform effectively which will find the malicious content more effectively as compared to other so that the proposed methodology will be highly effective as compared to previous.

References

[1] S. Sheng, B. Wardman, G. Warner, L.F. Cranor, J. Hong, C. Zhang, An empirical analysis of phishing blacklists, In: Proc. 6th Int. Conf. Email and Anti-Spam, CEAS'09, Mountain View, California, USA, 2009.

[2] S. Garera, N. Provos, M. Chew, A.D. Rubin, A framework for detection and measurement of phishing attacks. In: Proc. 5th ACM Workshop on Recurring Malcode, WORM'07, ACM, New York, NY, USA, 2007, pp. 1-8.

[3] J. Ma, L.K. Saul, S. Savage, G.M. Voelker, Beyond blacklists: Learning to detect malicious web sites from suspicious URLs, In: Proc. 15th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, Paris, France, 2009, pp. 1245-1254.

[4] J. Ma, L.K. Saul, S. Savage, G.M. Voelker, Identifying suspicious URLs: an application of large-scale online learning, In: Proc. 26th Annual Int. Conf. Machine Learning, ICML'09, Montreal, Quebec, Canada, 2009, pp. 681-688.

[5] C. Whittaker, B. Ryner, M. Nazif, Large-scale automatic classification of phishing pages, In: Proc. 17th Annual Network and Distributed System Security Symposium, NDSS'10, San Diego, CA, USA, 2010.

[6] PhishTank. Out of the net, into the tank, http://www.phishtank.com, accessed on June 18, 2010.

[7] R.B. Basnet, PyLongURL - Python library for longurl.org, software available at: http://code.google.com/p/pylongurl/, 2010.

[8] R.B. Basnet, S. Mukkamala, A.H. Sung, Detection of phishing attacks: a machine learning approach, In: Bhanu Prasad (Ed.), Studies in Fuzziness and Soft Computing, Springer, 2008, pp. 373-383.

[9] Fette, N. Sadeh, A. Tomasic, Learning to detect phishing emails, In: Proc. Int. Conf. World Wide Web, WWW'07, Banff, Alberta, Canada, 2007, pp. 649-656.

[10] Y. Zhang, J. Hong, L. Cranor, CANTINA: a content-based approach to detecting phishing web sites, In: Proc. 16th Int. Conf. World Wide Web, WWW'07, Banff, Alberta, Canada, 2007, pp. 639-648.

[11] Natural Language Toolkit (NLTK), http://www.nltk.org, accessed on July 15, 2011.