International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 4 | ISSN : 2456-3307

Identifying the Attack and Malicious Links in Net

M. SatishKumar, M. Eswar, P. Hemalatha
Ass. Professor Department of computer Applications SVCET, Chittoor, Andhra Pradesh, India

ABSTRACT

Fraudulent behaviours in Google Play, the most popular Android app market, fuel search rank abuse and malware proliferation. Malicious URLs are wide wont to mount varied cyber attacks together with spamming, phishing and mal- ware. Detection of malicious URLs and identification of threat varieties square measure vital to thwart these attacks. Knowing the kind of a threat allows estimation of severity of the attack and helps adopt a good countermeasure. Existing ways usually sight malicious URLs of one attack kind. During this paper, we have a tendency to propose technique using machine learning to sight malicious URLs of all the popular attack varieties and determine the character of attack a malicious uniform resource locator tries to launch. Our method uses a range of discriminative options together with textual properties, link structures, webpage contents, DNS information, and network traffic. Several of those features square measure novel and extremely effective.

Keywords: Malicious URL, attacks, cyber crimes.

I. INTRODUCTION

The commercial success of android app markets such as Google Play and also the incentive model they provide to popular apps, create them appealing targets for fraudulent and malicious behaviors. Whereas the planet Wide net has become a killer application on the net, it's conjointly brought in an large risk of cyber attacks. Adversaries have used the net as a vehicle to deliver malicious attacks like phishing, spamming, and malware infection. As an example, phishing generally involves causation an email apparently from a trustworthy supply to trick individuals to click a uniform resource locator (Uniform Resource Locator) contained within the email that links to a counterfeit webpage. To address Web-based attacks, a good effort has been directed towards detection of malicious URLs. A common step is to use a blacklist of malicious URLs, which may be made from varied sources, significantly human feedbacks that are extremely correct however time-consuming. Blacklisting incurs no false positives, yet is effective just for better-known malicious URLs. It cannot notice unknown malicious URLs. The terribly nature of exact match in blacklisting renders it simple to be evaded. This weakness of blacklisting has been self-addressed by anomaly-based notice on strategies designed to detect unknown malicious URLs. In these strategies, a classification model supported discriminative rules or options are built with either information a priori or through machine learning. Choice of discriminative rules or options plays a vital role for the performance of a detector. A main endeavor in malicious uniform resource locator detection has focused on choosing extremely effective discriminative features. Existing strategies were designed to notice malicious URLs of one attack sort, like spamming, phishing, or malware.

In existing system we are finding the DOS attack and thus informing to the administrator if there is a chance for attack. But here we are not finding which type of attack for which type of website.

In this paper, we propose a way using machine learning to notice malicious URLs of all the popular attack sorts as well as phishing, spamming and malware infection, and determine the attack sorts malicious URLs attempt to launch. We've got adopted an oversized set of discriminative options related to matter patterns, link structures, content composition, DNS data, and network traffic. Several of those options square measure novel and extremely effective. As delineate later in our experimental studies, link quality and bound lexical and DNS options are extremely discriminative in not only detecting malicious URLs however conjointly distinguishing attack sorts. Additionally, our method is powerful against better-known evasion techniques such as redirection, link manipulation, and fast-flux hosting. Identification of attack sorts is helpful since the knowledge of the character of a possible threat permits us to take a correct reaction still as a pertinent and effective step against the threat. As an example, we could handily ignore spamming however should respond immediately to malware infection.

II. ALGORITHMS

Decision tree:

Decision tree builds classification or regression models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node (e.g., Outlook) has two or more branches (e.g., Sunny, Overcast and Rainy). Leaf node (e.g., Play) represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.

In data mining, decision trees can be described also as the combination of mathematical and computational techniques to aid the description, categorization and generalization of a given set of data.

Data comes in records of the form:
The dependent variable, Y, is the target variable that we are trying to understand, classify or generalize. The vector **x** is composed of the features, x_1, x_2, x_3 etc., that are used for that task.

Decision tree learning is the construction of a decision tree from class-labeled training tuples. A decision tree is a flow-chart-like structure, where each internal (non-leaf) node denotes a test on an attribute, each branch represents the outcome of a test, and each leaf (or terminal) node holds a class label. The topmost node in a tree is the root node.

A decision tree is a graph that uses a branching method to demonstrate every possible outcome of a decision. Decision trees can be drawn by hand or created with a graphics program or specialized software. Casually, decision trees are useful for focusing discussion when a group must make a decision. Programmatically, they can be used to assign time or other values to possible outcomes so that decisions can be automated. Decision tree software is used in data mining to simplify complex strategic challenges and evaluate the cost-effectiveness of research and business decisions. Variables in a decision tree are usually represented by circles. They can be used either to drive informal discussion or to map out an algorithm that predicts the best choice mathematically. A decision tree typically starts with a single node, which branches into possible outcomes. Each of those outcomes leads to additional nodes, which branch off into other possibilities. This gives it a treelike shape. There are three different types of nodes: chance nodes, decision nodes, and end nodes. A chance node, represented by a circle, shows the probabilities of certain results. A decision node, represented by a square, shows a decision to be made, and an end node shows the final outcome of a decision path.

III. CONCLUSION

The Web has become an efficient channel to deliver various attacks such as spamming, phishing and malware. To thwart these attacks, we have presented a machine learning method to both detect malicious URLs and identify attack types. Decision tree algorithm is used for identifying the attacks.

IV. REFERENCES

[1]. CHENETTE, S. The ultimate deobfuscator. http: //securitylabs.websense.com/content/Blogs/3198.aspx, 2008.

[2]. CHUNG, Y.-J., TOYODA, M., AND KITSUREGAWA, M. Identifying spam link generators for monitoring emerging web spam. In WICOW: Proceedings of the 4th workshop on Information credibility(2010).

[3]. CISCO IRONPORT. IronPortWeb Reputation: Protect and defend against URL-based threat. http://www.ironport.com.

[4]. CORTES, C., AND VAPNIK, V. Support vector networks. Machine Learning (1995), 273–297.

[5]. CURL LIBRARY. Free and easy-to-use client-side url transfer library. http://curl.haxx.se/, 1997.

[6]. DMOZ. Netscape open directory project. http://www. dmoz.org.

[7]. DNS-BH. Malware prevention through domain blocking. http://www.malwaredomains.com.

[8]. FETTE, I., SADEH, N., AND TOMASIC, A. Learning to detect phishing emails. In WWW: Proceedings of the international conference on World Wide Web (2007).

[9]. GARERA, S., PROVOS, N., CHEW, M., AND RUBIN, A. D. A framework for detection and measurement of phishing attacks. In WORM: Proceedings of the Workshop on Rapid Malcode (2007).

[10]. GEOIP API, MAXMIND. Open source APIs and database for geological information. http://www.maxmind.com.

[11]. GYO NGYI, Z., AND GARCIA-MOLINA, H. Link spam alliances. In VLDB: Proceedings of the international conference on Very Large Data Bases (2005).

[12]. GYONGYI, Z., AND GARCIA-MOLINA, H. Web spam taxonomy, 2005

[13]. HOU, Y.-T., CHANG, Y., CHEN, T., LAIH, C.-S., AND CHEN, C.-M. Malicious web content detection by machine learning. Expert Systems with Applications (2010), 55–60.

[14]. JWSPAMSPY. E-mail spam filter for MicrosoftWindows. http: //www.jwspamspy.net.

[15]. LEE, K., CAVERLEE, J., AND WEBB, S. Uncovering social spammers: social honeypots + machine learning. In ACM SIGIR: Proceeding of the international conference on Research and development in Information Retrieval (2010).

[16]. MA, J., SAUL, L. K., SAVAGE, S., AND VOELKER, G. M. Beyond blacklists: learning to detect malicious web sites from suspicious URLs. In KDD: Proceedings of the international conference on Knowledge Discovery and Data mining (2009).

[17]. MA, J., SAUL, L. K., SAVAGE, S., AND VOELKER, G. M. Identifying suspicious URLs: an application of large-scale online learning. In ICML: Proceedings of the International Conference on Machine Learning (2009).

[18]. MCAFEE SITEADVISOR. Service for reporting the safety of web sites. http://www.siteadvisor.com/.

[19]. MCGRATH, D. K., AND GUPTA, M. Behind phishing: An examination of phisher modi operandi. In LEET: Proceedings of the USENIX Workshop on Large-Scale Exploits and Emergent Threats (2008).

[20]. MOORE, T., CLAYTON, R., AND STERN, H. Temporal correlations between spam and phishing websites. In LEET: Proceedings of the USENIX Workshop on Large-Scale Exploits and Emergent Threats (2009).

Author's Profile:

M. SatishKumar M.C.A., M.Tech., M.Phil., working as an Assoc.professor in Sri Venkateswara college of engineering &technology, Chittoor, A.P.

M. Eswar received the PG degree from SriVenkateswara college of engineering & Technology, Chittoor, A.P.

P. Hemalatha received the PG degree from Sri Venkateswara college of engineering & Technology, Chittoor, A.P.