IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022

DOI: 10.17148/IJARCCE.2022.11642

CYBER SECURITY OF MALWARE DETECTION ON ANDROID APPS

Narmada B[1], Syed Thajudeen S[2], Suryaprakash C[3], Venkatram R[4]

Assistant Professor and Head of the Department, Department of Computer Science and Engineering, Dhirajlal Gandhi College of Technology, Salem, Tamil Nadu, India[1]

Student, Department of Computer Science and Engineering, Dhirajlal Gandhi College of Technology, Salem, Tamil Nadu, India[2]

Student, Department of Computer Science and Engineering, Dhirajlal Gandhi College of Technology, Salem, Tamil Nadu, India[3]

Student, Department of Computer Science and Engineering, Dhirajlal Gandhi College of Technology, Salem, Tamil Nadu, India[4]

Abstract: Lately, the matter of dangerous malware in devices is spreading speedily, particularly those repackaged android malware. Though understanding robot malware mistreatment dynamic analysis will give a comprehensive read, it's still subjected to high price in setting preparation and manual efforts within the investigation. Android is the most preferred openly available smart phone OS and its permission declaration access management mechanisms can't sight the behavior of malware. the matter of police investigation such malware presents distinctive challenges thanks to the restricted resources accessible and restricted privileges granted to the user however conjointly presents distinctive opportunities within the needed data hooked up to every application. In our project, a code behavior signature-based malware detection framework mistreatment associate degree SVM rule is planned, which might sight malicious code and their variants effectively in runtime and extend malware characteristics information dynamically. Experimental results show that the approach incorporates a high detection rate and low rate of false positive and false negative, the power, and performance impact on the first system can even be unheeded. Our system extracts variety of options associate degreed trains a Support Vector Machine in an offline (off-device) manner, so as to leverage the upper computing power of a server or cluster of servers.

Keywords: Support Vector Machine,Svm Classifier,Malware

I. INTRODUCTION

Machine learning (**ML**) is a field of inquiry devoted to understanding and building methods that 'learn', that is, methods that leverage data to improve performance on some set of tasks. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks. Cloud computing is comparable to grid computing, a type of computing where unused processing cycles of all computers in a network are harnesses to solve problems too intensive for any stand-alone machine. Cloud computing is an on-demand service that is obtaining mass appeal in corporate data centers. The cloud enables the data center to operate like the Internet and computing resources to be accessed and shared as virtual resources in a secure and scalable manner. Like most technologies, trends start in the enterprise and shift to adoption by small business owners.

A subset of machine learning is closely related to computational statistics, which focuses on making predictions using computers, but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning. Some implementations of machine learning use data and229neural networks in a way that mimics the working of a biological brain. In its application across business problems, machine learning is also referred to as predictive analytics.

Learning algorithms work on the basis that strategies, algorithms, and inferences that worked well in the past are likely to continue working well in the future. These inferences can be obvious, such as "since the sun rose every morning for

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022
DOI: 10.17148/IJARCCE.2022.11642

the last 10,000 days, it will probably rise tomorrow morning as well". They can be nuanced, such as "X% of families have geographically separate species with color variants, so there is a Y% chance that undiscovered black swans exist".

Machine learning programs can perform tasks without being explicitly programmed to do so. It involves computers learning from data provided so that they carry out certain tasks. For simple tasks assigned to computers, it is possible to program algorithms telling the machine how to execute all steps required to solve the problem at hand; on the computer's part, no learning is needed. For more advanced tasks, it can be challenging for a human to manually create the needed algorithms. In practice, it can turn out to be more effective to help the machine develop its own algorithm, rather than having human programmers specify every needed step.
The discipline of machine learning employs various approaches to teach computers to accomplish tasks where no fully satisfactory algorithm is available. In cases where vast numbers of potential answers exist, one approach is to label some of the correct answers as valid. This can then be used as training data for the computer to improve the algorithm(s) it uses to determine correct answers. For example, to train a system for the task of digital character recognition, the MNIST dataset of handwritten digits has often been used.

II. SYSTEM STUDY

A. Exisiting System

PDF, as one of most popular document file format, has been frequently utilized as a vector by attackers to covey malware due to its flexible file structure and the ability to embed different kinds of content. In this paper, we propose a new learning-based method to detect PDF malware using image processing and processing techniques. The PDF files are first converted to grayscale images using image visualization techniques. Then various image features representing the distinct visual characteristics of PDF malware and benign PDF files are extracted. Finally, learning algorithms are applied to create the classification models to classify a new PDF file as malicious or benign. The performance of the proposed method was evaluated using Contagio PDF malware dataset. The results show that the proposed method is a viable solution for PDF malware detection. It is also shown that the proposed method is more robust to resist reverse mimicry attacks than the state-of-art learning-based method.

Disadvantage

✓ No existing work on secure DE duplication can properly address the reliability and tag consistency problem in distributed storage systems.

✓ It leads to data redundancy and data lose.

B. PROPOSED SYSTEM

This paper discusses, portrays and focuses on an SVM-based active learning framework for smart phone malware detection, and within the mechanical man system valid the effectiveness of the strategy, tests show that the planned methodology has sensible relevancy and measurability will be complete on a range of well-liked malware observation and might detect unknown malware. Due to its less impact on system performance, potentially significant impact on the initial system capability may go unnoticed. In summary, malware applications normally use the subsequent 3 sorts of penetration techniques for installation, activation, and running on the android device: Repackaging among the rest of many is the foremost common techniques for malware developers to put in malicious applications on a mechanical man platform. These sorts of approaches commonly begin from well liked legitimate Apps and misuse them as malware. The developers commonly transfer well-liked Apps, take apart them, add their own malicious codes, so reassemble and transfer the new App to official or different markets.

C. PROBLEM DEFINITION

To create an efficient system that curbs the threat of android malware by correctly detecting and mitigating any malicious APKs via combining permissions and API calls as features to characterize malware, and use machine learning techniques to automatically extract patterns to differentiate benign and malicious Apps.

Advantage

✓ The first step in any machine learning experiment is to get some domain knowledge which help understanding the data and accomplish the experiment. In this section we will briefly highlight common types of malware, malware analysis techniques, and available malware datasets. Which will demonstrate the flow of ideas and show the importance of the experiment.

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022
DOI: 10.17148/IJARCCE.2022.11642

III.SYSTEM REQUIREMENTS

3.1 Hardware Requirements
Processor : INTEL PENTIUM IV 2.6 Ghz
RAM : 4 GB
Hard Disk Drive : 500 GB
Key Board : Standard 128 Keys
Monitor : 17" TFT MONITOR
Mouse : Logitech Serial Mouse

3.2 Software Requirements
Operating System : Windows 7 or XP
Frond end : Dot net 2012
Back end : SQL Server 2005

IV.CONCLUSION

Hence, we have successfully proposed to use permissions and API calls of Android applications to detect malware and malicious codes in Android based mobile platform. Ours is a novel approach to distinguish and detect Android malware with different intentions. It is effective, that is, it is able to distinguish variant of Android malware between distinct purposes of them. The proposed framework extracts permissions from Android applications and further combines the API calls to characterize each application as a high dimension feature vector. By applying learning methods to the collected datasets, we can derive classification models to classify Apps as benign or malware. Experiments on real world data demonstrate the good performance of the framework for malware detection.

FUTURE WORK

The research on Java code has shown a strong presence of methods for manipulation on strings, as well as for downloading them outside of Java code. Such actions are manifestations of attempts to hide the real purpose of the application, i.e. obfuscation of the code. In addition, there has been a high use of methods that give access to and launch services (including system services). There is an increased presence of the method for data transfer over the HTTP protocol compared to secure applications. However, the quality of malware detection based on Java code proved to be low. None of the algorithms did exceed 81% of correctly classified instances. There are many reasons for this: the transformation and obfuscation of the code, the mechanism of reflection, manipulation on the chains of characters make the extraction of features a difficult task. Calling the API method can be implemented in several ways, and code transformation additionally increases the difficulty.

REFERENCES

[1] Androzoo Dataset, https://androzoo.uni.lu/
[2] CIC Dataset 2020, https://www.unb.ca/cic/datasets/maldroid2020.html
[3] VirusShare.com Dataset, https://virusshare.com/
[4] SandDroid-An automatic Android application analysis system. http://sanddroid.xjtu.edu.cn:8080/#home
[5] Shabtai, A., Kanonov, U., Elovici, Y. et al. "Andromaly": a behavioral malware detection framework for android devices. J Intell Inf Syst 38, 161–190 (2012).
[6] Justin Sahs and Latifur Khan.2012.A Machine Learning Approach to Android Malware Detection. In Proceedings of the 2012 European Intelligence and Security Informatics Conference (EISIC'12). IEEE Computer Society, USA, 141–147.
[7] Zhao M., Ge F., Zhang T., Yuan Z. (2011) AntiMalDroid: An Efficient SVM-Based Malware Detection Framework for Android. In: Liu C, Chang J, Yang A (eds) In- formation Computing and Applications. ICICA2011. Communications in Computer and Information Science, vol 243. Springer, Berlin, Heidelberg.
[8] Seung-Hyun Seo, Aditi Gupta, Asmaa Mohamed Sallam, Elisa Bertino, Kangbin Yim, Detecting mobile malware threats to homeland security through static analysis, Journal of Network and Computer Applications, Volume 38, 2014, Pages 43-53, ISSN 1084-8045,
[9] Arp, Daniel & Spreitzenbarth, Michael & Hubner, Malte & Gascon, Hugo & Rieck, Konrad.(2014). DREBIN: Effective and Explainable Detection of Android Malware in Your Pocket. Symposium on Network and Distributed System Security (NDSS).

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 6, June 2022

DOI: 10.17148/IJARCCE.2022.11642

[10] D. Wu, C. Mao, T. Wei, H. Lee and K. Wu," DroidMat: Android Malware Detection through Manifest and API Calls Tracing," 2012 Seventh Asia Joint Conference on Information Security, Tokyo, 2012, pp. 62-69, DOI: 10.1109/AsiaJCIS.2012.18. https://ieeexplore.ieee.org/document/6298136/

[11] William Enck, Machigar Ongtang, and Patrick McDaniel.2009. On light weight mobile phone application certification. In Proceedings of the 16th ACM conference on Computer and communications security CCS'09). Association for Computing Machinery, New York, NY, USA, 235–245.

[12] Sanz, Borja & Santos, Igor & Laorden, Carlos & Ugarte Pedrero, Xabier & Bringas, Pablo.

[13] B.H. Robbins, (2010), "Non Parametric Tests", B.H. Robbins Scholars Series, Dept. of Biostatistics, Vanderbilt University.

[14] Bostanci, Betul, & Erkan Bostanci, (2013), "An evaluation of classification algorithms using McNemars test", Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012). Springer, India.

[15] Dietterich, Thomas G, (1998), "Approximate statistical tests for comparing supervised classification learning algorithms." Neural computation 10.7, pp: 1895-1923.

[16] La Polla, Mariantonietta, Fabio Martinelli, & Daniele Sgandurra, (2013), "A survey on security for mobile devices", IEEE communications surveys & tutorials 15.1, pp: 446-471.

[17] Tam, Kimberly, (2017), The evolution of android malware and android analysis techniques", ACM Computing Surveys (CSUR) 49.4, pp: 76

[18] Liang, Shuang, & Xiaojiang Du, (2014), "Permission-combination-based scheme for android mobile malware detection", Communications (ICC), 2014 IEEE International Conference. 53

[19] Saracino, Andrea, (2016), "Madam: Effective and efficient behavior-based android malware detection and prevention", IEEE Transactions on Dependable and Secure Computing.

[20] Linn, Cullen, & Saumya Debray, (2003), "Obfuscation of executable code to improve resistance to static disassembly", Proceedings of the 10th ACM conference on Computer and communications security, ACM.

[21] Enck, William, Machigar Ongtang, & Patrick McDaniel, (2009), "On lightweight mobile phone application certification", Proceedings of the 16th ACM conference on Computer and communications security. ACM. [22] Vidas, Timothy, & Nicolas Christin, (2014), "Evading android runtime analysis via sandbox detection." Proceedings of the 9th ACM symposium on Information, computer and communications security. ACM.

[23] Burguera, Iker, Urko Zurutuza, & Simin Nadjm-Tehrani, (2011), "Crowdroid: behavior-based malware detection system for android", Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices. ACM.

[24] Mishra, Ratha, (2016), "Study of Random Tree and Random Forest Data Mining Algorithms for Microarray Data Analysis", International Journal on Advanced Electrical and Computer Engineering vol.3 issue 4.