Incremental Research on Cyber Security metrics in Android applications by implementing the ML algorithms in Malware Classification and Detection

Dr.Sreejith Vignesh B P[1]*

[1] Associate Professor & Head – Corporate Relations, J.K.K.Nattraja College of Engineering and Technology, India

* Correspondence: authorsree@gmail.com

Abstract

Cyber attacks are prevailing to be a great headache for the technical advancements especially when dealt with mobile usage in an android application environment. For a new user, it is difficult to identify the set of harmful permissions. This could be an advantage for malware intruders to access the data or infect the mobile device by introducing malware applications. Thus the face of Cybersecurity has changed in recent times with the advent of new technologies such as the Cloud, the internet of things, mobile/wireless, and wearable technology. The technological advances in data science which help develop contemporary cybersecurity solutions are storage, computing, and behavior. In this paper, the possible investigations are done on the cyber attacks in android by adopting the various malware classification and detection techniques. Various Classifications and Detections are done on various malware prevailing in the android applications.

Keywords: Android, Handheld devices, Malware Classifications and Malware Detection Techniques

1.Introduction

Smartphones are becoming more popular devices for many people. Due to the rapid entry of smartphones, malware has been spreading widely. An Android OS-based system, being the most popular platform for mobile devices therefore it becomes easy for attackers to target them. Many attackers and hackers take advantage of the lack of security standards and limited capabilities in handheld devices. As a result, it paved the way for the attackers to steal the personal information of the users of Android mobile devices. From the year 2011, the malware attacks were increased by 155 percent across all platforms. In particular, the handheld devices (using Android OS) are the platform with the highest malware growth rate by the end of 2014. Android phones have been sold more than 60% of overall smartphones available Android possessed 82.8% of the market share in 2015 reported in the global survey of the OS smartphone market, implying that the growth of the Android is increased when compared to other OS. Today The Android platform is the fastest growing market and faces some critical risk.

The rapid growth in computer science and information technology in recent times has led to the generation of a massive amount of data. As a result, these issues to be considered include optimization, uncertainty quantification, systems theory, statistics, and types of model development. Malware poses significant threats to the cybersecurity of national infrastructures, service sectors, and ultimately the society as a whole. As a result, malware attacks have extended to mobile devices and it has become a necessity to protect ourselves from such attacks. Traditional signature-

Received: Dec 20, 2019 Revised: March 21, 2020 Accepted: May 1, 2020

DOI: 10.5281/zenodo.3786137

based and change-based malware detection methods are not able to cope with new types of malware attacks. To address this problem, in this research project, we plan to develop a practical and effective "anomaly-based" malware detection system with an emphasis on a mobile computing platform. We carry out the generation of system metrics (i.e., feature vector) and a variety of efficient machine learning techniques to curtail the malware intrusion in smart devices.

A fundamental requirement of hybrid techniques is training and testing on a large dataset, which can lead to improved accuracy. However, existing techniques for machine learning-based approaches suffer from low accuracy and high false-positive rate (FPR). Also, these techniques have been tested on small datasets. With the emergence of big data [9] and the increasing number of malware patterns [4], machine learning-based malware detection techniques must be compared and tested on a large dataset to obtain high accuracy and low FPR. In this paper, we are motivated to improve Android malware detection techniques using big data analytics. Conventional fixed algorithms (hard-wired logic on decision-making level) have become ineffective against combating dynamically evolving cyber-attacks. This is why we need innovative approaches such as applying methods of applying Machine Learning algorithms and Artificial Intelligence (AI) that provide flexibility and learning capability to software which will assist humans in fighting cyber crimes as AI offers this and various other possibilities. Numerous nature-inspired computing methods of AI (such as Computational Intelligence, Neural Networks, Intelligent Agents, Artificial Immune Systems, Machine Learning, Data Mining, Pattern Recognition, Fuzzy Logic, Heuristics, etc.) have been increasingly playing an important role in cybercrime detection and prevention. AI enables us to design autonomic computing solutions capable of adapting to their context of use, using the methods of self-management, self-tuning, self-configuration, self-diagnosis, and self-healing When it comes to the future of information security To this end, we provide a comparison of seven different ML classifiers on the dataset - Using 55 GB dataset and a 19-node Spark cluster, we compared different classifiers including Isotonic Regression, Decision Trees, Random Forest, Gradient Boosted Trees, Support Vector Machine (SVM), Logistic Regression, and Multilayer Perception. We observed that in general, tree-based techniques provide better results.

2. Background

Early detection is the most important thing to mitigate the harmful effects of malware. Throughout the years, many malware detection methods have been proposed. These can be broadly categorized into signature-based, change-based, and anomaly-based methods. 1) Signature-based methods: A signature-based intrusion method detects malware based on its signature. It first gathers data and analyzes it, and when a program or file has a similar signature to an already existing malware (which it compares to from a database) it detects it. This method is often used for detecting popular malware signatures, but it can be quite slow since it should compare the signatures from a large database, meaning it cannot be instant [24]. 2) Change-based methods: Change based detection is a method that identifies when changes occurred in the system.

It relies on probability distribution to detect the changes. These techniques include online and offline change detection techniques. 3) Anomaly-based methods: "In the anomaly-based system, a system administrator defines the baseline, or normal state of the network's traffic load, breakdown, protocol, and typical packet size. The anomaly detector monitors network segments to compare their state to the normal baseline and look for anomalies" [24]. Up until now, virtually almost all real-world deployments of malware detection (like virus scanners) are signature-based and change-based methods. Though efficient, these methods are not able to identify new types of malware (such as those carrying out zero-day attacks).

Some research has been done on anomaly-based malware detection, which is good at identifying new malware. However, so far the anomaly-based methods are not widely deployed yet because of practical issues such as efficiency/scalability, high false-positive rate, difficulty to use, etc. Our research objective is to fill in this research gap and propose a practical and effective implementation of Malware classification and malware detection method by adopting the Machine learning algorithms embedded with artificial intelligence, especially in the new and emerging area of mobile computing.

Received: Dec 20, 2019 Revised: March 21, 2020 Accepted: May 1, 2020

DOI: 10.5281/zenodo.3786137

3. Related Work

Crowdroid [5] also used anomaly-based detection, however, their methods were somewhat different. Crowdroid used two types of datasets, one they developed for testing and the other from real malware. The research concluded that detecting malware through monitoring system calls would work for emerging and new malicious software. Aung and Zaw [2] proposed a model which first uses feature selection then applies k-means, next classifies the dataset using RF and J48 and finally analyzes it.

BPS Vignesh ad Rajesh Babu[12-15] proposed a model wherein various machine learning algorithms and SVM are adopted to classify the malware and to ensure the reliability of the software by checking the permissions obtained from the end-user of android applications.

RiskRanker used a method called zero-day detection, in which it checks through applications in the Android Market to determine which may be risky. This way the system narrows down the "potential risks". There are two order modules that RiskRanker follows, the "first-order module handles non-obfuscated apps by evaluating the risks". Brenner (2010) argues that "most of the cybercrime we see today simply represents the migration of real-world crime to cyberspace which becomes the tool criminals use to commit old crimes in new ways.

ScanDroid [9], which examined the data flows of the Android application to determine if it is benign or not. One of the main aspects of these scans is checking the permissions of the app, as well as the certifications. Like RiskRanker, ScanDroid should detect malware as soon as it is installed on the system. Shabtai et al. [20] proposed a host-based malware detection system called Andromaly, that inspects the set of features such as, outgoing and incoming communication statistics.

Dini et al. [22] also include system calls feature. Haung et al. [23] used ML algorithms for the detection of malicious applications using a permission-based approach in which requested permissions were compared with the required ones. Different labeling techniques were applied based on the source of the application and methods for anti-virus classification.

4. Applications of AI Algorithms to defense against Cyber Crimes in Smart Phones

Available academic resources show that AI techniques already have numerous applications in combating cybercrimes. AISs, just like the biological immune systems which they are based on, are employed to uphold stability in a changing environment. The immune-based intrusion detection comprises the evolution of immune cytes (self-tolerance, clone, variation, etc.) and antigens detection simultaneously. An immune system produces antibodies to resist pathogens and the intrusion intensity can be estimated by variation of the antibody concentration.

Therefore, AISs play an important role in the cybersecurity research, for instance, neural networks are being applied to intrusion detection and prevention, but there are also proposals for using neural networks in "Denial of Service (DoS) detection, computer worm detection, spam detection, zombie detection, malware classification, and forensic investigations" [5]. AI techniques such as Heuristics, Data Mining, Neural Networks, and AISs, have also been applied to new-generation anti-virus technology [7]. Some IDSs use intelligent agent technology which is sometimes even combined with mobile agent technology.

Mobile intelligent agents can travel among collection points to uncover suspicious cyber activity [2]. Wang et al. (2008) stated that the future of anti-virus detection technology is in the application of Heuristic Technology which means "the knowledge and skills that use some methods to determine and intelligently analyze codes to detect the unknown virus by some rules while scanning"[7]. This section will briefly present related work and some existing applications of AI techniques to cyber defense. Barika et al. (2009) presented a detailed architecture of a distributed IDS based on artificial neural network for enhanced intrusion detection in networks

5. **Adoption of Machine Learning Algorithms**

Received: Dec 20, 2019 Revised: March 21, 2020 Accepted: May 1, 2020

DOI: 10.5281/zenodo.3786137

Machine Learning is a branch of Computer Science that lets a computer learn automatically instead of being precisely programmed [26]. It evolved from the study of pattern recognition and computational learning theory in artificial intelligence [26]. It involves the development of programs that can learn, grow, and change with their experience on data. There are various types of machine learning algorithms available, here for better reliability on the security aspects of the cyber system the Ensemble ANFIS algorithm is used as it paves way for the easy embedding of the AI Concepts. An ANFIS network has applied for classification detecting malware and good ware applications. In ANFIS first model fuzzy inference system contains the fuzzy model. (Ying, 1998) proposed by Takagi, Sugeno, and Kang to generate fuzzy rules by formulating from an input-output dataset.

There are various Machine Learning algorithms varying with the type of application. Some of them are :

a) **Artificial Neural Networks**: It a machine-learning algorithm inspired by the structure and function of biological neural networks. They are used to model the complex relationships between inputs and outputs.

b) **Deep Learning**: It consists of multiple hidden layers in an artificial neural network. This approach tries to replicate the human brain in which it processes light and sound. It is widely used in computer vision and voice recognition.

c) **Genetic Algorithms**: It is a search technique, which is based on natural selection and uses methods and mutation and crossover to generate a new genotype (a set of parameters that define a proposed solution to the problem).

d) **Linear Discriminated Analysis**: It a data classification algorithm and a generalization of Fischer's linear discriminated technique to find the linear combination of features (Class variance and between-class variance) to separate two or more classes of data. In LDA, a linear function of attributes is created and the class function giving the highest probability represents the predicted class

e) **Support Vector Machines**: These are a set of supervised learning techniques that are used for regression and data classification. Based on their learning from past samples, these predict the category or class of a new sample. SVM tries to find a decision boundary at maximum distance from any sample in the training data. Here we Consider has three inputs and one output in the fuzzy interface. Takagi and Sugeno's type rule contains if-then rules of Sreejith and Babu (2015) as follows:

If x is A y is B and z is C then q is f (*x*, *y*, *z*) (1)

where *A*, Band *C* are the fuzzy sets in the backgrounds and $q = f(x, b, z)$ is a crumbly function in the consequent. $f(x, y, z)$ is a polynomial for the input *x*, *y*, and *z*. When $f(x, y, z)$ is a constant, a zero-order Sugeno fuzzy model is formed.

Rule1: If *x* is y, *y* is *z* is then = $x + y + z$ (2)

Rule2 : If *x* and *y* is *z* is then = $x + y + z$ (3)

Rule3 : If *xy* is *z* is then = $x y + z$ (4)

In this system, each rule adds the constant term with input variables to produce an output which is linear combination input. The final output is the weighted average of each rule's output. Receiving on the final result of input entropy, the ensemble-ANFIS classifier predicts the class accurately where it belongs to. As a result, if the value is greater than or equal to one, the applications are considered to be malware else if the result is zero then it is a good ware application. The major goal of the research is to compare different ML-based classification schemes to assess their efficacy in determining malware.

We applied seven different ML techniques to identify the most appropriate classification method for our model. These include Logistic Regression, Decision Tree, Random Forest, Gradient-boosted Tree, Multilayer Perception, SVM, and Isotonic Regression. We initially used all 83 features to train our model. However, it took more than 6 hours to train the dataset. We noticed that all features were not significant in detecting the malware. To determine relevant features, we applied the Chi-Square method to obtain p-values. Attributes with p-value > 0.05 were discarded. In total, 29 features were selected out of the total 83 features. After reducing the features, the training time

Received: Dec 20, 2019 Revised: March 21, 2020 Accepted: May 1, 2020

DOI: 10.5281/zenodo.3786137

was reduced to 90 minutes. For performance comparison, we utilized five metrics, that is, Precision, Recall, F1 score, Accuracy, and false-positive rate. The process of Machine Learning is similar to Data Mining because both look through data to recognize patterns.

5. Results and Discussions

The performance of this work is measured using precision, recall, F measure, accuracy which shows that an efficient result towards the proposed protocol. These results are discussed briefly below.

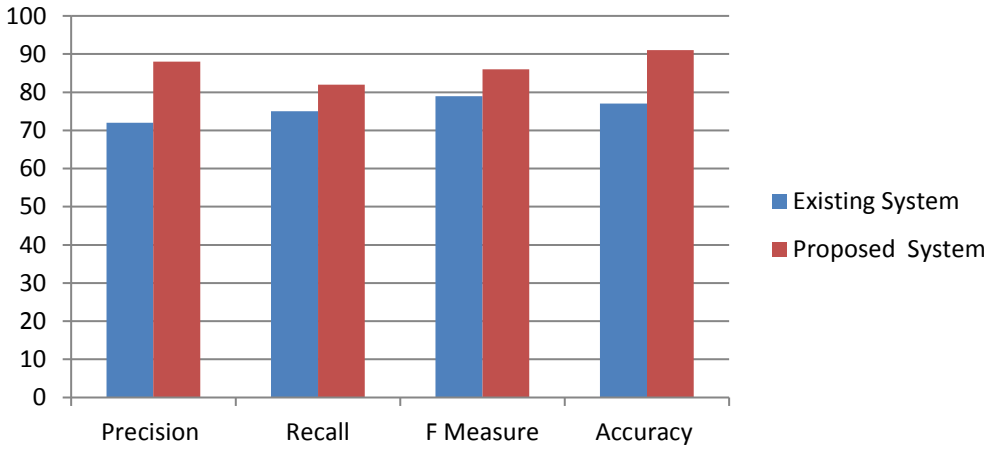

Figure 1. Comparison of Evaluation Metrics for Existing and Proposed System

The above Performance analysis chart proves that the adoption of machine learning algorithms records positive results than the existing system. So applications of Machine Learning algorithms in classifying and detecting malware proves to be the best and it gives the end-user to breathe easy in using the android applications regardless of the fear which rises out due to the cyber attacks.

6. Conclusions

Computing Techniques such as ubiquitous and mobile computing has increased the flexibility for mobile users to access the services from service providers. Most of the Android applications are commonly found freeware which enables the user to access the data in a single touch in hand. Taking this as an advantage, intruders started to create threats to the user by adding spyware in android applications. Therefore many applications that are considered as spyware injected are collected as samples and a chart is prepared to identify the malware classifications and detection approaches along with the malware avoidance techniques applied in that by the spyware writers and the demerit of the system are analyzed. Hence an approach is proposed using machine learning methods to detect the malware applications. The experimental results show that the ANFIS method detects malware application with high accuracy. The detection rate is also high compared with other techniques. This method can efficiently detect more malware applications in android smartphones.

Funding: "This research received no external funding"

Conflicts of Interest: "The authors declare no conflict of interest."

Received: Dec 20, 2019 Revised: March 21, 2020 Accepted: May 1, 2020

DOI: 10.5281/zenodo.3786137

References

1. J. Zico Kotler and Marcus A. Maloof. Learning to detect and classify malicious executables in the wild. J. Mach Learn Res., 7:2721-2744, December 2006.

2. I. Burguera, U.Z., Nadijm- Tehrani, S.: Crowroid: Behaviour –Based Malware Detection System for Android. In SPSM'11, ACM, October 2011.

3. A.Shabtai, U.Kanonov, Y.Elovici, C.Glezer, Y. Weiss: Andromaly: a behavioral malware detection framework for android devices. Journal of Intelligent Information systems 38(1) January 2011.

4. G.J. Tesauro, J.O. Kephart, and G.B. Sorkin, Neural networks for computer Virus recognition. IEEE Expert 11(4):5-6, August 1996.

5. G.Dini, F.Martinelli, A.Saracino, D.Sgandurra: MADAM:a Multilevel Anomaly Detector for Android Malware.

6. Schmidt, A.D., Peters, F., Lamour, F., Scheel, C., Camtepe, S.A, Albayrak, S.: Monitoring smartphones for anomaly detection. Mob. Netw.Appl.14(1)(2009) 92-106.

7. Mathew G. Schultz, Eleazar Eskin, Erez Zadok, and Salvatore J Stolfo. Data Mining methods for detection of new malicious executables IEEE Symposium on Security and Privacy, IEEE Computer Society.

8. Mohd Najwadi Yusoff and Aman Janatan, Optimizing Decision tree classification system using Genetic algorithm, International Journal on New Computer Architectures and their Applications 1(3): 694-713.

9. National Cyber Security Awareness Baseline Study, CyberSecurity Malaysia (October 2016)

10. Zarni Aung and Win Zaw, Detection of Android Malware Applications by using Machine Learning approaches, Proceedings of International Conference on Computer Networks and Information Technology PP:59-65.

11. Zarni Aung and Win Zaw, Permission-Based Android Malware Detection, International Journal of Scientific & Technology Research Volume 2, Issue 3, March 2013: ISSN 2277-8616.

12. SREEJITH, V. AND BABU, B.P.M.R. (2015) "Research study on various malwares its classification, detection and avoidance techniques applied in android mobile devices", International Journal of Applied Engineering Research, Vol. 10, No. 20, pp.20184–20187, ISSN: 0973-4562.

13. SREEJITH, V. AND BABU, B.P.M.R. (2016)"Certain investigations on various algorithms that is used to classify malware and goodware in android applications", ICTACT International Journal on Soft Computing, Vol. 7, No.1,pp.1344–1349.

14. SREEJITH, V. AND BABU, B.P.M.R. (2017) "Experimental research identifications on malware detection by embedding C4.5 algorithm and SVM in smart phones" Perspectivas em Ciencia da Informacao, v.22, sp.1, p303. , Nov. 2017 ISSN1413-9936

15. Vignesh, B.P.S. and Rajesh Babu, M. (2018) 'Classifying the malware application in the Android-based smart phones using ensemble-ANFIS algorithm', *Int. J. Networking and Virtual Organisations*, Vol. 19, Nos. 2/3/4, pp.257–269.

16. Jinyung Kim, Yongho Yoon, Kwangkeun Yi, Junbum Shin, and SWRD Center. 2012. ScanDal: Static analyzer for detecting privacy leaks in android applications.
MoST 12 (2012).

17. Diederik Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization.International Conference on Learning Representations (12 2014).

Received: Dec 20, 2019 Revised: March 21, 2020 Accepted: May 1, 2020

DOI: 10.5281/zenodo.3786137

18. Shuying Liang, Weibin Sun, and Ma.hew Might. 2014. Fast .ow analysis with godel hashes. In Source Code Analysis and Manipulation (SCAM), 2014 IEEE 14th
International Working Conference on. IEEE, 225–234.

19.Yepang Liu, Chang Xu, Shing-Chi Cheung, and Valerio Terragni. 2016. Understanding and detecting wake lock misuses for android applications. In Proceedings
of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering. ACM, 396–409.

20. Laurens van der Maaten and Geo.rey Hinton. 2008. Visualizing data using t-SNE. Journal of machine learning research 9, Nov (2008), 2579–2605.

21. Vinod Nair and Geo.rey E. Hinton. 2010. Recti€ed Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th International Conference on International Conference on Machine Learning (ICML'10). Omnipress, USA, 807–814. h.p://dl.acm.org/citation.cfm?id=3104322.3104425

22. Vlad Niculae. [n.d.]. A library for factorization machines and polynomial networks for classi€cation and regression in Python. Retrieved 5-22-19 from h.ps://github.com/scikit-learn-contrib/polylearn

23. Damien Octeau, Daniel Luchaup, Ma.hew Dering, Somesh Jha, and Patrick McDaniel. 2015. Composite constant propagation: Application to android inter-component communication analysis. In Proceedings of the 37th International Conference on So.ware Engineering-Volume 1. IEEE Press, 77–88.

24. Lucky Onwuzurike, Enrico Mariconti, Panagiotis Andriotis, Emiliano De Cristofaro, Gordon Ross, and Gianluca Stringhini. 2019. MaMaDroid: Detecting Android
Malware by Building Markov Chains of Behavioral Models (Extended Version). ACM Trans. Priv. Secure. 22, 2, Article 14 (April 2019), 34 pages. h.ps://doi.org/10.1145/3313391

25. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. .irion, O. Grisel, M. Blondel, P. Pre.enhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12 (2011), 2825–2830.

26. Naser Peiravian and Xingquan Zhu. 2013. Machine learning for android malware detection using permission and api calls. In Tools with Arti€cial Intelligence (ICTAI), 2013 IEEE 25th International Conference on. IEEE, 300–305.

27. Ste.en Rendle. 2010. Factorization machines. In Data Mining (ICDM), 2010 IEEE 10th International Conference on. IEEE, 995–1000.

Received: Dec 20, 2019 Revised: March 21, 2020 Accepted: May 1, 2020

DOI: 10.5281/zenodo.3786137