JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Survey of Android Malware Prediction using Artificial Intelligence Techniques

[1]Anu Malik, [2]Harshita Kamboj, [3]Sarvendra Kumar*

[1,2,3]Neelkanth Institute of Technology, Meerut, India

e-mail: *sarvendraricky@gmail.com; anumalik1707@gmail.com

Abstract : Android overlay enables one app to draw over other apps by creating an extra View layer atop the host View, which nevertheless can be exploited by malicious apps (malware) to attack users. To combat this threat, prior countermeasures concentrate on restricting the capabilities of overlays at the OS level while sacrificing overlays usability; recently, the overlay mechanism has been substantially updated to prevent a variety of attacks, which however can still be evaded by considerable adversaries. Malware remains a big threat to cyber security, calling for machine learning based malware detection. While promising, such detectors are known to be vulnerable to evasion attacks. This paper presents survey of android malware prediction using artificial intelligence techniques.

IndexTerms **– Android, Malware, Artificial Intelligence, Secuiry, Attack, Cyber.**

I. INTRODUCTION

The popularity of the Android platform in smartphones and other Internet-of-Things devices has resulted in the explosive of malware attacks against it. Malware presents a serious threat to the security of devices and the services they provided, e.g. stealing the privacy sensitive data stored in mobile devices. This work raises a stacking ensemble framework SEDMDroid to identify Android malware. Specifically, to ensure individual's diversity, it adopts random feature subspaces and bootstrapping samples techniques to generate subset, and runs Principal Component Analysis (PCA) on each subset. The accuracy is probed by keeping all the principal components and using the whole dataset to train each base learner Multi-Layer Perception (MLP). Then, Support Vector Machine (SVM) is employed as the fusion classifier to learn the implicit supplementary information from the output of the ensemble members and yield the final prediction result [1].

Figure 1: Android malware

Detecting Android malware is imperative. Among various detection schemes, permission pair based ones are promising for practical detection. However, conventional schemes cannot simultaneously meet requirements for practical use in terms of efficiency, intelligibility, and stability of detection performance. Although the latest scheme relies on differences of frequent pairs between benign apps and malware, it cannot meet the stability. This is because recent malware tends to require unnecessary permissions to imitate benign apps, which makes using the frequencies ineffective [3]. Machine learning (ML) has been widely used for malware detection on different operating systems, including Android.

To keep up with malware's evolution, the detection models usually need to be retrained periodically (e.g., every month) based on the data collected in the wild. However, this leads to poisoning attacks, specifically backdoor attacks, which subvert the learning process and create evasion tunnels for manipulated malware samples. To date, we have not found any prior research that explored this critical problem in Android malware detectors [4]. In recent years, Ransomware has been a critical threat that attacks smartphones. Ransomware is a kind of malware that blocks the mobile's system and prevents the user of the infected device from accessing their data until a ransom is paid. Worldwide, Ransomware attacks have led to serious losses for individuals and stakeholders.

However, the dramatic increase of Ransomware families makes to the process of identifying them more challenging due to their continuously evolved characteristics. Traditional malware detection methods (e.g., statistical-based prevention methods) fail to combat the evolving Ransomware since they result in a high percentage of false positives. Indeed, developing a non-classical, intelligent technique to safeguarding against Ransomware is of significant importance [6].

The availability of big data and affordable hardware have enabled the applications of deep learning on different tasks. With respect to security, several attempts have been made to transfer deep learning's application from the domain of image recognition or natural language processing into malware detection. In this study, we propose AdMat - a simple yet effective framework to characterize Android applications by treating them as images [8]. Despite being crucial to today's mobile ecosystem, app markets have meanwhile become a natural, convenient malware delivery channel as they actually "lend credibility" to malicious apps. In the past few years, machine learning (ML) techniques have been widely explored for automated, robust malware detection, but till now we have not seen an ML-based malware detection solution applied at market scales. To systematically understand the real-world challenges, we conduct a collaborative study with T-Market, a popular Android app market that offers us large-scale ground-truth data [9]. Android malware poses severe threats to users, hence raising an urgent demand for malware detection. In-cloud Android malware detection often suffers privacy leakage and communication overheads. Therefore, this article focuses on on-device Android malware detection. At present, on-device malware detectors are usually trained on servers and then transplanted to mobile devices (e.g., smartphones). In practice, on-device training is particularly important due to the demand for offline updates. Because mobile devices are limited in resource, however, on-device training is hard to implement, especially for those high-complexity malware detectors. To overcome this challenge, we design a lightweight on-device Android malware detector, based on the recently proposed broad learning method [10].

II. BACKGROUND

H. Zhu et al.,[1] show experimental results on two separate datasets collected by static analysis way to prove the effectiveness of the SEDMDroid. The first one extracts permission, sensitive API, monitoring system event and so on that are widely used in Android malwares as the features, and SEDMDroid achieves 89.07% accuracy in term of these multi-level static features. The second one, a public big dataset, extracts the sensitive data flow information as the features, and the average accuracy is 94.92%. Promising experiment results reveal that the proposed method is an effective way to identify Android malware.

A. Alzubaidi et al.,[2] In recent years, the global pervasiveness of smartphones has prompted the development of millions of free and commercially available applications. These applications allow users to perform various activities, such as communicating, gaming, and completing financial and educational tasks. These commonly used devices often store sensitive private information and, consequently, have been increasingly targeted by harmful malicious software. This paper focuses on the concepts and risks associated with malware, and reviews current approaches and mechanisms used to detect malware with respect to their methodology, associated datasets, and evaluation metrics.

H. Kato et al.,[3]. Propose Android malware detection based on a Composition Ratio (CR) of permission pairs. We define the CR as a ratio of a permission pair to all pairs in an app. We focus on the fact that the CR tends to be small in malware because of unnecessary permissions. To obtain features without using the frequencies, we construct databases about the CR. For each app, we calculate similarity scores based on the databases. Finally, eight scores are fed into machine learning (ML) based classifiers as features. By doing this, stable performance can be achieved. Since our features are just eight-dimensional, the proposed scheme takes less training time and is compatible with other ML based schemes. Furthermore, our features can quantitatively offer clear information that helps human to understand detection results. Our scheme is suitable for practical use because all the requirements can be met. By using real datasets, our results show that our scheme can detect malware with up to 97.3% accuracy. Besides, compared with an existing scheme, our scheme can reduce the feature dimensions by about 99% with maintaining comparable accuracy on recent datasets.

C. Li et al et al.,[4] motivated to study the backdoor attack against Android malware detectors. The backdoor is created and injected into the model stealthily without access to the training data and activated when an app with the trigger is presented. We demonstrate the proposed attack on four typical malware detectors that have been widely discussed in academia. Our evaluation shows that the proposed backdoor attack achieves up to 99% evasion rate over 750 malware samples. Moreover, the above successful attack is realised by a small size of triggers (only four features) and a very low data poisoning rate (0.3%).

L. Gong, Z. Li et al.,[5] To address these shortcomings, a more pragmatic approach is to enable early detection of overlay-based malware during the app market review process, so that all the capabilities of overlays can stay unchanged. For this purpose, in this paper we first conduct a large-scale comparative study of overlay characteristics in benign and malicious apps, and then implement the OverlayChecker system to automatically detect overlay-based malware for one of the worlds largest Android app stores. In particular, we have made systematic efforts in feature engineering, UI exploration, emulation architecture, and run-time

environment, thus maintaining high detection accuracy (97% precision and 97% recall) and short per-app scan time (1.7 minutes) with only two commodity servers, under an intensive workload of 10K newly submitted apps per day.

I. Almomani et al.et al.,[6] introduces a new methodology for the detection of Ransomware that is depending on an evolutionary-based machine learning approach. The binary particle swarm optimization algorithm is utilized for tuning the hyperparameters of the classification algorithm, as well as performing feature selection. The support vector machines (SVM) algorithm is used alongside the synthetic minority oversampling technique (SMOTE) for classification. The utilized dataset is collected from various sources, which consists of 10,153 Android applications, where 500 of them are Ransomware. The performance of the proposed approach SMOTE- tBPSO-SVM achieved merits over traditional machine learning algorithms by having the highest scores in terms of sensitivity, specificity, and g-mean.

F. Mercaldo and A. Santone et al.,[7] Several techniques to overcome the weaknesses of the current signature based detection approaches adopted by free and commercial anti-malware were proposed by industrial and research communities. These techniques are mainly supervised machine learning based, requiring optimal class balance to generate good predictive models. In this paper, we propose a method to infer mobile application maliciousness by detecting the belonging family, exploiting formal equivalence checking. We introduce a set of heuristics to reduce the number of mobile application comparisons and we define a metric reflecting the application maliciousness. Real-world experiments on 35 Android malware families (ranging from 2010 to 2018) confirm the effectiveness of the proposed method in mobile malware detection and family identification.

L. N. Vu and S. Jung, "AdMat et al.,[8] The novelty of our study lies in the construction of an adjacency matrix for each application. These matrices act as “input images” to the Convolutional Neural Network model, allowing it to learn to differentiate benign and malicious apps, as well as malware families. During the experiment, we found that AdMat was able to adapt to a variety of training ratios and achieve the average detection rate of 98.26% in different malware datasets. In classification tasks, it also successfully recognized over 97.00% of different malware families with limited number of training data.

L. Gong et al et al.,[9] Our study illustrates that the key to successfully developing such systems is multifold, including feature selection and encoding, feature engineering and exposure, app analysis speed and efficacy, developer and user engagement, as well as ML model evolution. Failure in any of the above aspects could lead to the “wooden barrel effect” of the whole system. This article presents our judicious design choices and first-hand deployment experiences in building a practical ML-powered malware detection system. It has been operational at T-Market, using a single commodity server to check ~12K apps every day, and has achieved an overall precision of 98.9 percent and recall of 98.1 percent with an average per-app scan time of 0.9 minutes.

W. Yuan, Y. Jiang et al.,[10] Our detector mainly uses one-shot computation for model training. Hence it can be fully or incrementally trained directly on mobile devices. As far as detection accuracy is concerned, our detector outperforms the shallow learning-based models, including support vector machine (SVM) and AdaBoost, and approaches the deep learning-based models multilayer perceptron (MLP) and convolutional neural network (CNN). Moreover, our detector is more robust to adversarial examples than the existing detectors, and its robustness can be further improved through on-device model retraining. Finally, its advantages are confirmed by extensive experiments, and its practicality is demonstrated through runtime evaluation on smartphones.

K. Liu et al.,[11] presents complements the previous reviews by surveying a wider range of aspects of the topic. This paper presents a comprehensive survey of Android malware detection approaches based on machine learning. We briefly introduce some background on Android applications, including the Android system architecture, security mechanisms, and classification of Android malware. Then, taking machine learning as the focus, we analyze and summarize the research status from key perspectives such as sample acquisition, data preprocessing, feature selection, machine learning models, algorithms, and the evaluation of detection effectiveness. Finally, we assess the future prospects for research into Android malware detection based on machine learning. This review will help academics gain a full picture of Android malware detection based on machine learning. It could then serve as a basis for subsequent researchers to start new work and help to guide research in the field more generally.

D. Li and Q. Li et al.,[12] Ensemble learning typically facilitates countermeasures, while attackers can leverage this technique to improve attack effectiveness as well. This motivates us to investigate which kind of robustness the ensemble defense or effectiveness the ensemble attack can achieve, particularly when they combat with each other. We thus propose a new attack approach, named mixture of attacks, by rendering attackers capable of multiple generative methods and multiple manipulation sets, to perturb a malware example without ruining its malicious functionality. This naturally leads to a new instantiation of adversarial training, which is further geared to enhancing the ensemble of deep neural networks. We evaluate defenses using Android malware detectors against 26 different attacks upon two practical datasets. Experimental results show that the new adversarial training significantly enhances the robustness of deep neural networks against a wide range of attacks; ensemble methods promote the robustness when base classifiers are robust enough, and yet ensemble attacks can evade the enhanced malware detectors effectively, even notably downgrading the VirusTotal service.

III. CONCLUSION

Android applications are developing rapidly across the mobile ecosystem, but Android malware is also emerging in an endless stream. Many researchers have studied the problem of Android malware detection and have put forward theories and methods from different perspectives. Existing research suggests that machine learning is an effective and promising way to detect Android malware. Notwithstanding, there exist reviews that have surveyed different issues related to Android malware detection based on machine learning. In future implement prediction model with improved accuracy using efficient machine learning classification technique.

REFERENCES

1. H. Zhu, Y. Li, R. Li, J. Li, Z. You and H. Song, "SEDMDroid: An Enhanced Stacking Ensemble Framework for Android Malware Detection," in IEEE Transactions on Network Science and Engineering, vol. 8, no. 2, pp. 984-994, 1 April-June 2021, doi: 10.1109/TNSE.2020.2996379.
2. A. Alzubaidi, "Recent Advances in Android Mobile Malware Detection: A Systematic Literature Review," in IEEE Access, vol. 9, pp. 146318-146349, 2021, doi: 10.1109/ACCESS.2021.3123187.
3. H. Kato, T. Sasaki and I. Sasase, "Android Malware Detection Based on Composition Ratio of Permission Pairs," in IEEE Access, vol. 9, pp. 130006-130019, 2021, doi: 10.1109/ACCESS.2021.3113711.
4. C. Li et al., "Backdoor Attack on Machine Learning Based Android Malware Detectors," in IEEE Transactions on Dependable and Secure Computing, doi: 10.1109/TDSC.2021.3094824.
5. L. Gong, Z. Li, H. Wang, H. Lin, X. Ma and Y. Liu, "Overlay-based Android Malware Detection at Market Scales: Systematically Adapting to the New Technological Landscape," in IEEE Transactions on Mobile Computing, doi: 10.1109/TMC.2021.3079433.
6. I. Almomani et al., "Android Ransomware Detection Based on a Hybrid Evolutionary Approach in the Context of Highly Imbalanced Data," in IEEE Access, vol. 9, pp. 57674-57691, 2021, doi: 10.1109/ACCESS.2021.3071450.
7. F. Mercaldo and A. Santone, "Formal Equivalence Checking for Mobile Malware Detection and Family Classification," in IEEE Transactions on Software Engineering, doi: 10.1109/TSE.2021.3067061.
8. L. N. Vu and S. Jung, "AdMat: A CNN-on-Matrix Approach to Android Malware Detection and Classification," in IEEE Access, vol. 9, pp. 39680-39694, 2021, doi: 10.1109/ACCESS.2021.3063748.
9. L. Gong et al., "Systematically Landing Machine Learning onto Market-Scale Mobile Malware Detection," in IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 7, pp. 1615-1628, 1 July 2021, doi: 10.1109/TPDS.2020.3046092.
10. W. Yuan, Y. Jiang, H. Li and M. Cai, "A Lightweight On-Device Detection Method for Android Malware," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 51, no. 9, pp. 5600-5611, Sept. 2021, doi: 10.1109/TSMC.2019.2958382.
11. K. Liu, S. Xu, G. Xu, M. Zhang, D. Sun and H. Liu, "A Review of Android Malware Detection Approaches Based on Machine Learning," in IEEE Access, vol. 8, pp. 124579-124607, 2020, doi: 10.1109/ACCESS.2020.3006143.
12. D. Li and Q. Li, "Adversarial Deep Ensemble: Evasion Attacks and Defenses for Malware Detection," in IEEE Transactions on Information Forensics and Security, vol. 15, pp. 3886-3900, 2020, doi: 10.1109/TIFS.2020.3003571.