On the Deterioration of Learning-Based Malware Detectors for Android

Xiaoqin Fu
Washington State University
Pullman, WA
xiaoqin.fu@wsu.edu

Haipeng Cai
Washington State University
Pullman, WA
haipeng.cai@wsu.edu

***Abstract*—Classification using machine learning has been a major class of defense solutions against malware. Yet in the presence of a large and growing number of learning-based malware detection techniques for Android, malicious apps keep breaking out, with an increasing momentum, in various Android app markets. In this context, we ask the question "what is it that makes new and emerging malware slip through such a great collection of detection techniques?". Intuitively, performance deterioration of malware detectors could be a main cause—trained on older samples, they are increasingly unable to capture new malware. To understand the question, this work sets off to investigate the deterioration problem in four state-of-the-art Android malware detectors. We confirmed our hypothesis that these existing solutions do deteriorate largely and rapidly over time. We also propose a new classification approach that is built on the results of a longitudinal characterization study of Android apps with a focus on their dynamic behaviors. We evaluated this new approach against the four existing detectors and demonstrated significant advantages of our new solution. The main lesson learned is that studying app evolution provides a promising avenue for long-span malware detection.**

I. Introduction

In the past few years, numerous machine learning based approaches have been proposed to detect malicious software. For mobile apps in Android, in particular, new malware detectors are being proposed continuously in the literature [9]. However, despite the rich body of solutions proposed, new Android malware never stops breaking out in and threatening varied mobile app markets. In fact, the momentum of Android malware has been being on rise [1]. An immediate implication of this is that existing malware detectors are possibly not sufficiently effective in detecting *new* and *emerging* malware [3]. In other words, current learning-based approaches might have been deteriorating over time, which causes the degradation of their performance. A plausible reason lies in the reliance of these approaches on constant retraining (which is not always practically affordable given the typically substantial cost of training a model on a large set of samples) with new training samples (which are not always timely available). Thus, a more effective solution should not only be accurate on particular datasets but also be able to keep the accuracy for a long span without frequent retraining with changing apps.

We target malware detectors for Android that are based on supervised learning, and investigate the deterioration phenomena of these detectors using four state-of-the-art malware detectors for Android, including one state-of-the-art dynamic app classifier [2] and three state-of-the-art static approaches [8], [7], [6]. We further propose developing a new Android malware detector that is based on evolutionary characterization of apps in terms of their dynamic behaviors. We comparatively evaluated the capabilities of the proposed approach against the four state-of-the-art prior approaches as baselines, on 24,780 malware and benign apps developed across eight years from 2010 through 2017. We empirically demonstrated the rapid deterioration of existing approaches in their classification performance, and the significant advantages of the evolution-based approach we proposed.

Specifically, our results revealed that, even with a span of one year, the state-of-the-art detection performance dropped from about 90% to below 30% in terms of F1 accuracy, and the highest accuracy (F1) was mostly below 65% for various lengths of span. Our evaluation also shows that the proposed approach achieves F1 accuracy superior to (by 6–11% on average) the chosen baselines for *same-period* detection (testing apps of the same year as the training data). For classifying apps appeared one to seven years later after training (*over-time* detection), our approach also significantly outperformed all the four baseline techniques (by 7–41% on average) over all the (seven) possible spans.

These findings not only corroborated our hypothesis about the relative short span of existing learning-based malware detectors achieving competitive performance, but also suggested studying and characterizing the evolutionary characteristics of apps as a promising avenue towards long-span malware detection.

II. Our Approaches

To understand the same-period and over-time classification performance of existing malware detection solutions, we chose four state-of-the-art malware detectors for Android: *MamaDroid* [7], *DroidSieve* [8], *Afonso* [2], and *RevealDroid* [6]. *MamaDroid* uses features based on API calls in an app, extracted through a static analysis. It is the only prior work we are aware of that explicitly reported over-time (across years) detection results. However, it is possible that some other approaches have good performance in that setting albeit not having been evaluated so, which is why we selected the other three techniques for our comparative studies. Among these three, *DroidSieve* uses features computed from app resources, *Afonso* [2] classifies apps based on calls to predefined lists of APIs and system calls, and *RevealDroid* approaches app classification based on apps' usage of APIs, native code, and reflection.

TABLE I
COMPARISON OF PERFORMANCE (F1) AMONG THE FIVE MALWARE DETECTORS IN SAME-PERIOD DETECTION SETTINGS

Dataset	our classifier	*MamaDroid*	*DroidSieve*	*Afonso*	*RevealDroid*
B10+M10	93.62%	83.67%	88.22%	87.18%	85.49%
B11+M11	94.13%	97.93%	81.51%	89.78%	86.16%
B12+M12	94.23%	83.77%	85.60%	89.35%	82.77%
B13+M13	95.25%	90.60%	89.26%	91.72%	88.30%
B14+M14	92.50%	84.49%	60.20%	86.65%	83.67%
B15+M15	90.36%	84.42%	86.47%	75.14%	79.39%
B16+M16	93.39%	89.55%	89.68%	80.25%	83.89%
B17+M17	98.22%	94.84%	89.62%	95.19%	94.67%
Average	**93.75%**	**88.05%**	**82.36%**	**86.71%**	**85.02%**

Our entire study dataset includes 13,627 benign and 11,153 malicious samples, for a total of 24,780 benchmarks. These benign and malicious apps were developed across the past eight years from 2010 to 2017, according to which we divided the entire datasets into 16 yearly datasets, noted as $M10$ to $M17$ for malware and $B10$ to $B17$ for benign apps. All the 16 datasets are mutually disjoint (there were no apps shared by any two datasets).

We conducted two studies. In the same-period study, we assess the performance of each technique with training and testing apps developed in a same period of time (in the minimal unit of year). For each given mixed dataset (e.g., $B10$ and $M10$), we randomly selected a third of samples from each class (malware or benign) and reserved them as unseen/novel samples for training, and the remaining for testing. In the over-time study, we assess the performance of these techniques when they are trained on older datasets (e.g., $B12$ and $M12$) and predict labels of newer ones (e.g., $B17$ and $M17$), spanning one to seven years.

Upon our revelation of poor over-time performance of the four techniques studied, we explored a novel classification approach inspired by the an evolutionary characterization of malicious and benign Android apps that also span the same eight years, but with entirely different app samples to avoid biases in later training our classifier. For the characterization study, we used the same metrics adopted in our previous Android dynamic study [5]. These metrics have been recently shown to be effective in differentiating benign and malicious apps in same-period, but not over-time, detection settings [4]. From these previous works, we manually selected a subset of those metrics and additionally used a few more features based on method-level taint flows [3]. With these selected features, we built our classifier by simply training a random forest model. We evaluated our classifier against the four prior ones with the same two studies.

III. EMPIRICAL RESULTS

Table I lists the F1 accuracy of each of the 8 independent tests (noted in first column) achieved by our classifier versus the four baselines. As shown, our classifier achieved F1 ranging from 90% to 98% for the datasets of any of the past eight years. Also, the classification performance had very small variations. In comparison, the baselines all had relatively large variations, showing the dependence of their performance on particular datasets. As an overall average, the F1 accuracy of *DroidSpan* was 93.75%, versus 88.05% by *MamaDroid*. *Afonso* achieved slightly lower F1 than *MamaDroid* (86.71%), but higher than other two baselines.

TABLE II
COMPARISON OF AVERAGE PERFORMANCE (F1) AMONG THE FIVE MALWARE DETECTORS IN OVER-TIME DETECTION SETTINGS

Detector	Length of span between training and testing (years)							Overall
	1	2	3	4	5	6	7	
Our classifier	74.55%	65.75%	65.53%	74.37%	60.68%	71.01%	81.49%	71.43%
MamaDroid	61.85%	57.31%	51.34%	58.15%	52.95%	65.22%	80.53%	63.67%
DroidSieve	32.90%	25.41%	45.01%	35.80%	24.44%	57.60%	0.93%	29.87%
Afonso	55.91%	38.49%	47.61%	66.63%	56.52%	63.16%	27.66%	49.91%
RevealDroid	53.20%	46.53%	45.53%	42.47%	35.74%	44.53%	55.40%	45.94%

For over-time detection, Table II lists the mean F1 accuracy of each technique from all the 28 tests with one- through seven-year spans between training and testing data. The numbers indicate our classifier performed better than the four baselines at any of the seven spans. The gap was not very large for *MamaDroid* with the seven-year span (81.49% versus 80.53%). However, at all other spans, the advantages of our approach over the baselines were much more substantial. Over the seven spans, the average F1 of our approach was the highest, followed by *MamaDroid*, *Afonso*, *RevealDroid*, and *DroidSieve* in order.

IV. CONCLUSION

We studied the deterioration of learning-based malware detectors for Android with both same-period and over-time settings. Our results confirmed the significant, albeit not continuous/monotonic, deterioration of four state-of-the-art techniques, especially in the over-time settings. We also explored a novel approach informed by an evolutionary characterization of app behaviors and showed its superior performance in both settings over the four prior techniques as baselines. These findings revealed the promise of evolution-based approaches to long-span malware detection. As a next step, we plan to understand the underlying causes for the deterioration problem and develop more effective malware detection solutions accordingly.

REFERENCES

[1] Android malware accounts for 97% of all malicious mobile apps. http://www.scmagazineuk.com/updated-97-of-malicious-mobile-malware-targets-android/article/422783/, 2015.

[2] V. M. Afonso, M. F. de Amorim, A. R. A. Grégio, G. B. Junquera, and P. L. de Geus. Identifying Android malware using dynamically obtained features. *Journal of Computer Virology and Hacking Techniques*, 11(1):9–17, 2015.

[3] H. Cai and J. Jenkins. Towards sustainable android malware detection. In *Proceedings of the 40th International Conference on Software Engineering: Companion Proceeedings*, pages 350–351. ACM, 2018.

[4] H. Cai, N. Meng, B. Ryder, and D. Yao. Droidcat: Effective android malware detection and categorization via app-level profiling. *IEEE Transactions on Information Forensics and Security*, 2018.

[5] H. Cai and B. Ryder. Understanding Android application programming and security: A dynamic study. In *International Conference on Software Maintenance and Evolution (ICSME)*, pages 364–375, 2017.

[6] J. Garcia, M. Hammad, and S. Malek. Lightweight, obfuscation-resilient detection and family identification of android malware. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 26(3):11, 2018.

[7] E. Mariconti, L. Onwuzurike, P. Andriotis, E. De Cristofaro, G. Ross, and G. Stringhini. MAMADROID: Detecting android malware by building markov chains of behavioral models. In *Proceedings of Network and Distributed System Security Symposium*, 2017.

[8] G. Suarez-Tangil, S. K. Dash, M. Ahmadi, J. Kinder, G. Giacinto, and L. Cavallaro. DroidSieve: Fast and accurate classification of obfuscated android malware. In *Proceedings of ACM Conference on Data and Application Security and Privacy*, pages 309–320, 2017.

[9] D. J. Tan, T.-W. Chua, V. L. Thing, et al. Securing Android: a survey, taxonomy, and challenges. *ACM Computing Surveys*, 47(4):1–45, 2015.