Poster: Feasibility of Malware Visualization Techniques against Adversarial Machine Learning Attacks

Harun Oz[1], Faraz Naseem[1], Ahmet Aris[1], Abbas Acar[1], Guliz Seray Tuncay[2], and A. Selcuk Uluagac[1]
[1] Cyber-Physical Systems Security Lab., Florida International University, Florida, USA
[2] Google
Emails: {hoz001, fnase001, aaris, aacar, suluagac}@fiu.edu, gulizseray@google.com

Machine Learning (ML) has been indispensable to malware detection in recent years. Particularly, its subset - deep learning-based models can provide superior performance over traditional methods (i.e., signature-based or heuristic-based) for malware detection [1]. However, recent research has shown that the efficiency of ML-based techniques can drop drastically due to adversaries attacking these systems via adversarially crafted/perturbed inputs. Such attacks have their roots in the computer vision domain with the study of Szegedy et al. [2], and then followed by others [3]–[5]. In the malware detection domain, adversarial ML attacks to ML-based malware detectors involve adding carefully crafted perturbations to the malware samples that preserve the malicious functionality of the malware while allowing the samples to evade the target ML-based malware classifiers (i.e., modified malware samples are classified as benign). Using such attacks, researchers were able to craft adversarial malware samples and successfully evaded ML-based malware detection systems including Windows Portable Executable (PE)-based malware detectors [6]–[8], Android malware detectors [9], [10], PDF-malware classifiers [11], [12] and even cloud based proprietary anti-virus engines (e.g., Kaspersky, Eset, Sophos) [13]. These examples clearly demonstrate that it is possible for attackers to evade state-of-the-art ML-based malware classifiers not by complex concealment techniques (e.g., polymorphism, metamorphism, packing), but by simple, minute adversarial perturbations carefully crafted via adversarial ML attacks.

In order to defend ML-based malware classifiers from such attacks, researchers employed defense mechanisms such as adversarial training [2]. However, such mechanisms are computationally costly and also suffer from model poisoning and decreased detection accuracy [8]. Therefore, defending ML-based malware detection systems against adversarial ML attacks is still an open problem.

The motivation of this work is to develop a practical method to quickly and efficiently detect malware, based on the family it belongs to, in a way that is robust against adversarial ML attacks and does not require costly adversarial defense mechanisms. To achieve this, we propose the use of visualization-based malware detection. In this preliminary work, we show that converting the malware detection problem into image-based malware classification problem provides robustness against adversarial ML attacks. The underlying robustness stems from the fact that adversarial ML attacks, which are relatively easy to apply to images in the computer vision, are extremely difficult to apply to transformed images of malware samples. This is because such an operation that adds carefully crafted adversarial noise to a malware image has a very high possibility of breaking the functionality of the actual malware when the image is converted back to a malware binary.

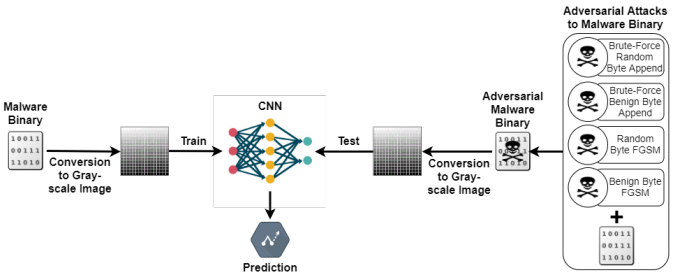

Fig. 1. An overview of our proposed approach. Malware binaries are converted into gray-scale images before being fed to the CNN for the training process. Four adversarial ML attacks are applied to malware binaries. Similar to the training process, these samples are converted to images and then fed to the model as input in order to classify them according to the malware family they belong to.

A visual depiction of our approach is shown in Figure 1. In the first stage, each malware binary undergoes pre-processing during which each binary in our dataset is converted to an array of unsigned 8-bit integers and normalized to a common size. These arrays, which represent the binaries as gray-scale images, are then used to train a Convolutional Neural Network (CNN) in the second stage. In the third stage, adversarial examples are then generated using each of the black box (i.e., brute-force random byte append and brute-force benign byte append) and white box (i.e., random byte FGSM and benign byte FGSM) attacks.

In order to test the efficacy of image-based malware classification, and compare it with state-of-the-art ML-based classification, we used the 2015 Microsoft Malware Classification Challenge dataset [14] which includes real malware samples, including obfuscated ones, from nine different malware families in the Windows Portable Executable (PE) format. As the state-of-the-art classifier, we followed the prior studies

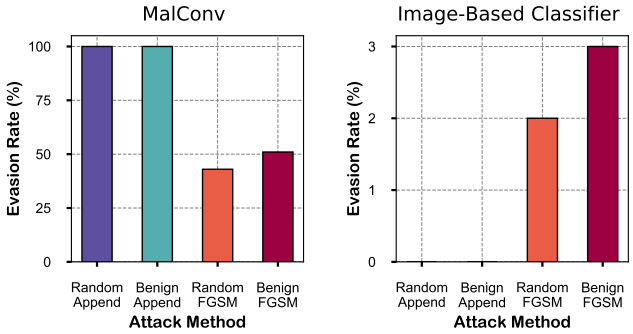

Fig. 2. A side-by-side comparison of the evasion rates of the attacks used in this study when applied to MalConv and our Image-Based Classifier. It is evident that the vast majority of adversarial examples generated using the four attacks methods failed to misclassify the image-based classifier.

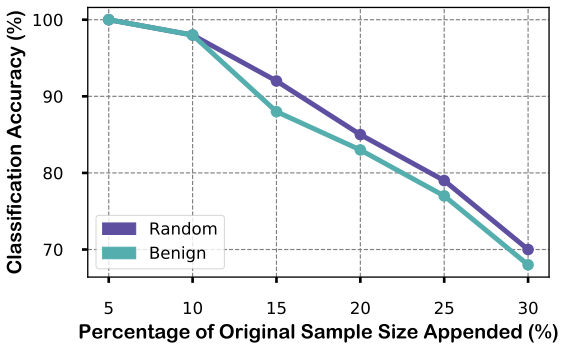

Fig. 3. The classification accuracy of the image-based classifier with respect to the amount of bytes appended to the samples as a percentage of the original sample size.

([6]–[8], [15], [16]) and trained MalConv [17], a CNN-based malware classifier that analyzes the raw bytes of PE-based malware samples. As the image-based classifier, we trained a CNN-based classifier using the gray-scale images of the malware in the dataset. To evaluate the robustness of our image-based classifier, we performed four adversarial ML attacks that preserve the functionality of malware samples to both of the classifiers.

The results of our evaluation show that the image-based malware detection approach is robust against adversarial ML attacks that can easily fool a state-of-the-art ML-based malware detector as shown in Figure 2. The evasion rate of adversarial samples dropped to 0% in certain attacks. Furthermore, our tests demonstrate that even if an adversary increases the amount of adversarial perturbations by up to 20% of the malware sample's original size, our image-based malware detector still provides a detection accuracy of above 80% as shown in Figure 3. Moreover, we analyzed the overhead incurred by implementation. The analysis indicates that the image-based malware detection technique provides a 70% decrease in training time and a three-fold reduction in RAM usage during the training process in comparison to a start-of-the-art ML-based malware classifier.

Our preliminary analysis shows that image-based classifiers are both efficient and also robust against adversarial ML attacks that preserve the functionality of the malware. For this reason, employing an image-based malware classifier does not require additional defense mechanisms, such as adversarial training; hence, it remains immune to model poisoning. To the best of our knowledge, this is the first work in the adversarial malware literature that demonstrates and analyzes the robustness of image-based classifiers against adversarial ML attacks. As future work, we will employ more adversarial ML attacks to our study such as attacks that perturb individual parts of a PE binary using the LIEF library [18]. In addition, we will incorporate more ML classifiers in addition to MalConv, and enlarge our malware dataset to include more malware binaries.

Acknowledgement

This work was partially supported by the U.S. National Science Foundation (Award: NSF-CAREER CNS-1453647). The views expressed are those of the authors only, not of the funding agencies.

References

[1] D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, "A survey of deep learning methods for cyber security," *Information*, vol. 10, no. 4, 2019.
[2] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *2nd ICLR*, 2014.
[3] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *3rd ICLR*, 2015.
[4] N. Papernot, P. D. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *IEEE European Symposium on Security and Privacy, EuroS&P, March 21-24*, 2016, pp. 372–387.
[5] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP)*, May 2017, pp. 39–57.
[6] B. Kolosnjaji, A. Demontis, B. Biggio, D. Maiorca, G. Giacinto, C. Eckert, and F. Roli, "Adversarial malware binaries: Evading deep learning for malware detection in executables," in *26th EUSIPCO*. IEEE, 2018.
[7] F. Kreuk, A. Barak, S. Aviv, M. Baruch, B. Pinkas, and J. Keshet, "Deceiving end-to-end deep learning malware detectors using adversarial examples," in *NeurIPS*, 2018.
[8] B. Chen, Z. Ren, C. Yu, I. Hussain, and J. Liu, "Adversarial examples for cnn-based malware detectors," *IEEE Access*, vol. 7, 2019.
[9] W. Yang, D. Kong, T. Xie, and C. A. Gunter, "Malware detection in adversarial settings: Exploiting feature evolutions and confusions in android apps," in *Proceedings of the 33rd ACSAC*. ACM, 2017.
[10] L. Chen, S. Hou, and Y. Ye, "Securedroid: Enhancing security of machine learning-based detection against adversarial android malware attacks," in *Proceedings of the 33rd ACSAC*. ACM, 2017.
[11] N. Rndic and P. Laskov, "Practical evasion of a learning-based classifier: A case study," in *2014 IEEE S&P*, May 2014, pp. 197–211.
[12] W. Xu, Y. Qi, and D. Evans, "Automatically evading classifiers: A case study on PDF malware classifiers," in *23rd NDSS*, 2016.
[13] R. L. Castro, C. Schmitt, and G. Dreo, "Aimed: Evolving malware with genetic programming to evade detection," in *18th IEEE TrustCom*, 2019.
[14] R. Ronen, M. Radu, C. Feuerstein, E. Yom-Tov, and M. Ahmadi, "Microsoft malware classification challenge," *CoRR*, 2018. [Online]. Available: http://arxiv.org/abs/1802.10135
[15] W. Fleshman, E. Raff, R. Zak, M. McLean, and C. Nicholas, "Static malware detection & subterfuge: Quantifying the robustness of machine learning and current anti-virus," in *Proceedings of the AAAI*, 2018.
[16] O. Suciu, S. E. Coull, and J. Johns, "Exploring adversarial examples in malware detection," in *IEEE SPW*, 2019.
[17] E. Raff, J. Barker, J. Sylvester, R. Brandon, B. Catanzaro, and C. K. Nicholas, "Malware detection by eating a whole EXE," in *The Workshops of the The Thirty-Second AAAI*, 2018.
[18] "Library to instrument executable formats," https://lief.quarkslab.com/, 2020, [Online; accessed 2-February-2020].