An Effective Machine Learning-Based File Malware Detection—A Survey

Ashwin A. Kumar, G. P. Anoosh, M. S. Abhishek and C. Shraddha

Abstract The objective of this paper is to enable computers to learn on their own, identify malicious activities, increase scanner efficiency and sensitivity. The machine learning algorithm enables the identification of patterns in observed data, the development of models that explains the world and the prediction of things without explicitly preprogrammed rules and models. There have been huge research interests in the cybersecurity industry as well as in universities in the subjects of how to effectively block malicious documentation without a sign of slowing down. The main aim of the paper is to investigate the efficiency of large files and increase sensitivity in malware detection.

Keywords Malware · Machine learning · Scanner · Vulnerabilities

1 Introduction

Machine learning (ML) plays a key role in a wide range of serious applications, such as data mining, the processing of natural languages, image recognition, and skilled systems. ML provides likely solutions in all these and more domains and is set to support our future development.

Cybersecurity is a set of technology and approaches designed to save attack, unofficial access, change or destruction of computer systems, networks, applications, and facts. Network security systems and computer (host) security structures encompass cyber protection systems. At a minimum, everyone has a firewall, antivirus software and intrusion detection system (IDS). IDS helps to become aware of, manage and decide the unauthorized use, duplication, change, and demolition of the statistics machine [1]. Violations of security consist of external invasions (assaults from outside the company) and inner invasions (attacks inside the agency). However, the surprisingly complex nature of many real global issues often way that it's miles

A. A. Kumar (✉) · G. P. Anoosh · M. S. Abhishek · C. Shraddha
Department of Computer Science and Engineering, Vidyavardhaka College of Engineering,
Mysuru, Karnataka, India

V. Bindhu et al. (eds.), *International Conference on Communication, Computing and Electronics Systems*, Lecture Notes in Electrical Engineering 637,
https://doi.org/10.1007/978-981-15-2612-1_34

unreasonable, if not possible, to find out particular algorithms to be able to resolve them results easily at all times.

It is possible to use a weak document in order to carry out malware payload, which can be embedded or downloaded from a document file, to spread malware across documents. In most cases, JavaScript supports the document to enable entire weaknesses and then execute code of choice for the attacker. This includes ambiguity and storage management techniques such as buffer overflow, ROP, and encrypted heap shellcode [2].

Many methods for chunk-based documented attacks ranging from passive detection, for example, to dynamic analyzes by means of sandboxing technologies have been developed. One of the advantages of signature-based detection is that the sensing of known malware is fairly low. On the other hand, since signature-based detection uses byte commands to match a particular malware, zero-day attacks or malware deviations are not dynamic. This presents major challenges for AV scanners that rely heavily on signature detection. Another solution is to perform analysis and behavior-based malware detection.

This utilizes sandbox software to add an additional identification layer. This monitors enforcement shown in a text folder when it is accessed in a determined context rather than using byte instructions, and if a certain comportement is detected, a warning action will be made. Even with highly complicated content like JavaScript to attack, this increases the detection rate. It should be remembered that the sandbox-based technology operates only if an identified file conducts malicious actions in a real world. They also regularly find ways to avoid sandboxing systems with so-called anti-sandbox approaches. If an atmosphere for sandboxes is observed, for instance, then good behavior and sleep mode can be seen. There are also other boundaries. Some sandboxing tools address only detailed PDF types of attacks, for example, MDScan for JavaScript [3], Nozzle for heap spray [4], or record the lively compliance of a system and require manual analysis to make an unveiling decision, as in the case of CWSandbox [5].

Printers are part of every corporate and personal network these days, so that every network has a good survival chance. In many networks, network printers and multifunction printers (MFP), throughout general, have arisen as cyber-attackers because they are meant to serve several (wired and wireless) interfaces and direct many protocols in order to support a huge base of domestic workers and ad hoc traffic. The viability of cooperation with all manner of printers [6–10], which could be supplied by the relevant section of the study, has been accentuated previous research and guides. Protocol assaults include: Denial-of-service assaults, privilege escalation, the leakage of print jobs, or system filings and even code execution on a printer itself [13]. Printing protocol abuses include select malicious motions an attacker might perform on a target printer.

2 Literature Review

In Zhang [11], the MLPDF model uses a back-propagating algorithm with a stochastic descent search to update a model to effectively PDF-based malware detection using a machine learning-based approach, the neural network model of multi-layer perceptron (MLP) to pinpoint PDF-based malware, called MLPDF. The datasets used are brain and malicious PDF papers. The findings of the study show an impressive MLPDF approach that is well above all eight well-known commercial antivirus scanners evaluated.

Chen et al. [12] have used the technology of IoT, WSN, and cloud technology to give information to the farmers in their phone. They have introduced nodes, which are a set of different sensors, combined to form a single unit to measure various physical and environmental factors. The details are then stored in the cloud and compiled. The datasets are Drebin and MaMaDroid (5879 malware samples) and are analyzed using the R programming and the Cloud MongoDB. The result is the rate of detection of malware decreased in Mama Droid from 96 to 1% and in Drebin from 97 to 1%, with only a little distortion caused by our method of manipulation for example.

Hecht and Sagi [13] included behavioral systems used in infogain, gain ratio and correlation (Pearson) methods to analyze and detect network-printer attacks to achieve the best results in print-protocol traffic detection. Base data is mild and deceptive in terms of experimental observations, whereby the proposed architecture identifies printable protocol attacks efficiently, offering a marginal fall-positive rate of 99.9% accuracy.

Liu et al. [14] describe the detection of adversarial examples based on steganalysis where the author presents adversarial examples that can be detected effectively with the steganalysis detector. Attack methods are based on network gradient calculations like fast gradient sign method (FGSM), fast gradient value (FGV), and Jacobian-based saliency map attack (JSMA). In comparison, some techniques, such as L-BFGS, Deepfool, and Carlini and Wagner (C&W) assault, are focused on solving optimization problems. Dataset is an ImageNet10-class.

Clements and Lao [15] the fast gradient sign method (FGSM) generate in the direction of the sign of the cost functions gradient to produce an adversarial input with very slight perturbation. The Jacobian-based saliency maps attack (JSMA) algorithm uses the gradients of the learned function, rather than the cost function, to produce a saliency map of the input. Datasets are MNIST and CIFAR10. Experimental results show that the proposed algorithms achieve 100% stealthiness for both datasets under all adversarial scenarios.

Sohi et al. [16] have used network intrusion detection systems focus on signature-based intrusion detection methods exhibiting a lower level of false positives, compared to the anomaly based detector. Synthetic datasets generated using overlay methodology, where four different scenarios are taken into account. The number of alarms raised by the Bro running against the same pool of mixed data can indicate how much improvement can be achieved by applying our method.

Alkasassbeh and Almseidin [17] describe machine learning methods for network intrusion detection of the use of the KDD99 dataset in which the detection price became 88 percent of widespread attacks, whether recognized or unknown. The basic benefit of this research is the minimum amount of academic information that wants to produce top traffic category outcomes.

Zhang and Su [18] blanketed Machine Learning Attack and Defense on Voltage Over-scaling Lightweight Authentication results show that ANN, RNN, and CMA-ES can clone the mission reaction behavior of VOS-based fully authentication with predictive accuracy of up to 99.65%, while predictive accuracy is much lower than 51.2% after deploying our proposed ML resilient method.

Cai [19] proposed a preliminary study on Android Malware Detection's sustainability. Datasets include the first collection of 1221 harmless applications (oldBen) in each device class by installing the top 50 popular apps. By uploading the top 100 popular apps in each device class, the latest benevolent dataset (newBen) was collected. DroidSpan achieved F1 accuracy of 91% (versus MamaDroid's 75%). This procedure has shown that not only does DroidSpan effectively spot malware, but it also maintains high accuracy of detection for four years (93% F1 measurement) (81% F1 five years).

3 Discussion

Comparing how PDF and other commercial scanners do with larger information will be useful as part of the future research, particularly including more recent PDF documents to the dataset. To address such mechanisms, it will raise awareness of protective mechanisms against such assaults and attack changes. No previous research has focused on detecting attacks by learning and testing supervised ML classifiers on traditional (non-3-d) printer protocols.

Detector cannot have very good performance when it is not trained and tested on the same adversarial method. So we will try to explore methods for training one detector against different kinds of adversarial attacks. These techniques such as detection using side-channel information suffer from reduced sensitivity toward small Trojans. Attempts to improve the ability of NIDS systems to defend against them by extending their signature databases and generating a more realistic and close to the real-world ground truth to test a NID. The model has a limited amount of time to examine whether the input image is natural or not.

Individuals put up a digital or tough replica image in the passive face recognition to register their identity in a destiny identification gadget. This approach is smaller in magnitudes than non-malicious ones for properly modeling and forecasting the destiny values of time malicious activities. This is a common problem in the detection of anomalies. The MLP classifier has the lowest end result for the Brute Force attack, which implies that MLP can not interpret Brute Force assault details among all the different information. KDD database has 41 attributes and all of them have been registered, but as part of destiny research, additional classifiers and the role selection to see the most relevant characteristics could be investigated.

4 Conclusion

Machine learning has been developed as a new computer system capability. Machine learning will make our future stronger than any other innovation this century. Rapid progress in information storage and the strength of computer processing have dramatically changed the game over the last few years. The facts are very large, the time taken to calculate is improved, and this is where machine learning takes place to help people with large information in a minimum of time. In this paper, we examined the technique of malware detection based entirely on the behavior of documents that distributed primitive access. Our findings show that files are regularly distinguished for use beyond a few years. This location explains the negative aspects of the survey papers and also the high degree of accuracy and resilience to various obstruction systems.

References

1. Mukkamala, S., Sung, A., Abraham, A.: Cyber security challenges: designing efficient intrusion detection systems and antivirus tools. Enhancing Comput. Secur. Smart Technol. 125–163 (2005)
2. Zhang, J., Rabaiotti, J.: The PDF exploit: same crime, different face. https://www.symantec.com/connect/blogs/pdf-exploit-same-crime-different-face/. Last accessed 18 March 2018
3. Tzermias, Z., Sykiotakis, G., Polychronakis, M., Markatos, E.P.: Combining static and dynamic analysis for the detection of malicious documents. In: Proceedings of the Fourth European Workshop on System Security. EUROSEC'11, pp. 1–6 (2011)
4. Ratanaworabhan, P., Livshits, B., Zorn, B.: NOZZLE: a defense against heap spraying code injection attacks. In: Proceedings of the 18th Conference on USENIX Security Symposium. SSYM'09. Berkeley, CA USA (2009)
5. Willems, G., Holz, T., Freiling, F.: Toward automated dynamic malware analysis using CWSandbox. IEEE Secur. Priv. **5**, 32–39 (2007)
6. Kim, F.X.: Phenoelit: attacking networked embedded devices. Presented in Black Hat USA 2002. http://www.blackhat.com/presentations/bh-usa-02/bh-us02-phenoelit-network.pdf. Last accessed 18 March 2018
7. Adrian, C.: Hacking network printers. http://www.irongeek.com/i.php?page=security/networkprinterhacking. Last accessed 18 March 2018 (2017)
8. Sibert, W.O.: Malicious data and computer security. In: Proceedings of the 19th National Information Systems Security Conference (1996)
9. Muller, J., Mladenov, V., Somorovsky, J., Schwenk, J.: SoK: exploiting network printers. In: 2017 IEEE Symposium on Security and Privacy, pp. 213–230 (2017)
10. Cui, A., Costello, M., Stolfo, S.J.: When firmware modifications attack: a case study of embedded exploitation. Ndss (2013)
11. Zhang, J.: MLPdf: an effective machine learning based approach for PDF malware detection, pp. 1–6 (2018)

12. Chen, X., Li, C., Wang, D., Wen, S., Zhang, J., Nepal, S., Xiang, Y., Ren, K.: Android HIV: a study of repackaging malware for evading machine-learning detection (2018)
13. Hecht, A., Sagi, A.: PIDS: a behavioral framework for analysis and detection of network printer attacks, pp. 1–20 (2018)
14. Liu, J., Zhang, W., Zhang, Y., Hou, D., Liu, Y., Zha, H., Yu, N.: Detection based defense against adversarial examples from the steganalysis point to view (2018)
15. Clements, J., Lao, Y.: Hardware trojan attacks on neural networks (2018)
16. Sohi, S.M., Ganji, F., Seifert, J.-P.: Recurrent neural networks for enhancement of signature-based network intrusion detection systems (2018)
17. Alkasassbeh, M., Almseidin, M.: Machine learning methods for network intrusion detection and intrusion prevention systems. Pro. Quest Diss. Theses. **106** (2018)
18. Zhang, J., Su, H.: Machine learning attack and defense on voltage over-scaling-based lightweight authentication, pp. 1–12 (2018)
19. Cai, H.: a preliminary study on the sustainability of android malware detection. arXiv Comput. Sci. (2018)