Proceedings of the 6th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/71/20241665

A review of black-box adversarial attacks and defenses in machine learning-based malware detection

Jiaxiang Chen

Hangzhou Dianzi University, Hangzhou, 310018, China

cjx12036@gmail.com

Abstract. In recent years, significant advancements have been made in cyber security, particularly through the application of machine learning (ML) methods. ML-based techniques have enhanced system security by effectively distinguishing between malicious and benign objects across various domains, including spam email detection, social media content filtering, intrusion detection systems, and malware detection. This paper focuses on the specific area of ML-based malware detection and its advantages over traditional methods, such as improved accuracy and the ability to generalize to unknown threats. Despite these advancements, ML-based malware detection systems are vulnerable to adversarial attacks, especially in black-box scenarios where the internal workings of the detection model are not accessible. This paper provides a comprehensive review of adversarial attacks on black-box malware detection systems and examines the current defense mechanisms against these attacks. Understanding the challenges and strategies in this field, this review aims to offer insights into enhancing the robustness and security of ML-based malware detection systems.

Keywords: Adversarial attacks, Malware detection, Black-Box models, Machine learning.

1. Introduction

Machine learning (ML) methods are now commonly employed by researchers and security experts to enhance system security and drive innovation. Numerous cybersecurity issues involve distinguishing between benign and malicious objects [1] detecting malware [2]. In the specific area of malware detection (MD), which involves classifying software executables as either benign or malicious, ML offers several advantages over traditional detection methods [3]. The benefits include enhanced accuracy in classification and detection, the capability to swiftly process large datasets, and the ability to recognize new or unknown threats, thus offering clearer insights into behavior patterns.

Given the significant achievements of existing ML based malware detection systems, it is imperative to evaluate their resilience against adversarial attacks. Adversarial attacks can significantly compromise the effectiveness of malware detection systems by allowing malicious software to evade detection. Also, the majority of current malware detection systems operate in a black-box manner, where the internal workings of the detection model are not accessible [4]. Therefore, we did a brief survey of the current popular black-box counter attacks and defenses. This paper provides a comprehensive review of adversarial attacks on these black-box systems, offering readers a broad understanding of the challenges and strategies in this field.

DOI: 10.54254/2755-2721/71/20241665

With this paper, people can have a general understanding of black box adversarial attacks and defenses in ML-based malware detection, and a general understanding of the current research trend and some open issue in this field.

2. Background

2.1. Machine learning in malware detection

Machine learning-based malware detection processes in a feature space, using digital representations of executables. These features enable the model to predict the classification of previously unseen samples. As shown in Figure 1, The machine learning-based malware classification process comprises five key phases:

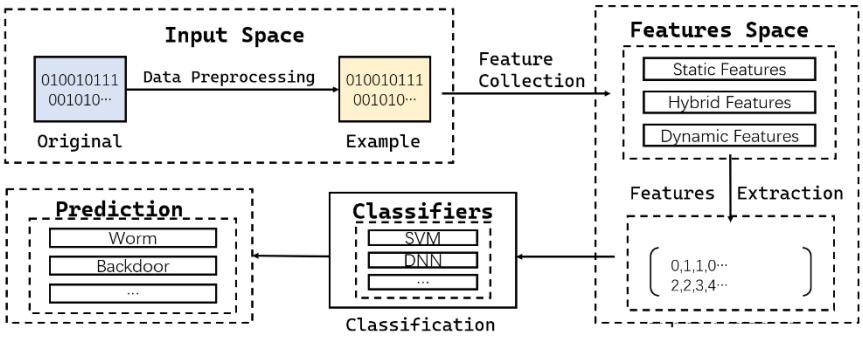

Figure 1. Process of malware detection classifier

2.1.1. Data preprocessing. Original samples are processed to fit the classifier, resulting in examples. Techniques like data augmentation and equalization are applied to strengthen malware classifiers.

2.1.2. Feature collection. This phase involves gathering low-level features, like static, hybird or dynamic features. Static features like raw bytes, operation codes, and function calls are collected, while dynamic methods execute samples to gather behavioral features like API calls, hardware-based features, and network traffic.

2.1.3. Feature extraction. To improve the classification model, high-level features are extracted in this section, to select key attributes, decrease data size, and create input features.

2.1.4. Classification. Machine learning models are employed as classifiers to predict input features. In this section, fundamental ML models as well as deep learning models are used for malware classification.

2.1.5. Decision making. Eventual decisions are made according to classifier prediction results. Binary classification methods determine the likelihood of a sample being benign or malicious, while multi-class classification methods identify different malware families.

Input features for malware classification typically fall into two categories: static and dynamic features. (1) Static Features: Extracted from the software without execution, including binary code, source code, and assembly code. (2) Dynamic Features: Obtained by executing the software in an isolated environment and analyzing its runtime behavior.

2.2. Attacker's aim

Suppose a model M can classify benign input b into 1, malign input m into 2, the attacker's aim is to generate an sample m' from m which can evade the model M. The goal can be summarized as:

$$M(m) = 0; M(b) = 1; \text{generate } m' \text{ from } m;\ M\left(m'\right) = 1 \tag{1}$$

DOI: 10.54254/2755-2721/71/20241665

Which is aiming at making malware bypass the ML-based detection system.

2.3. Threat models

Here are the three main threat models used to understand adversarial attacks and defenses in the field: the white-box models, gray-box models, and black-box models.

In the black-box model, attackers have no knowledge of the structure and parameters of the target network. However, they can interact with the deep learning (DL) algorithm by querying predictions for specific inputs. Attackers typically generate adversarial samples using a substitute classifier, trained with data-prediction pairs and other benign and adversarial samples.

In the gray-box model, attacker has knowledge of the architecture of the target model but is inaccessible to its weights. Like in the black-box model, adversaries can interact with the deep learning (DL) algorithm by querying predictions for specific inputs. The additional structural information available in the gray-box model typically leads to better attack performance compared to the black-box model.

The white-box model represents the strongest adversary, who know everything about the target model, from its architecture to its parameters. This allows the attackers to adapt their attacks and precisely generate adversarial samples for the target model.

3. Gray & Black-box adversarial attacks in ML-based malware detection

In Gray and Black box scenario, the target model only provide predicted outputs. Tactics in these scenarios must rely on indirect strategies like transferability and query-based, as they lack direct access to the model. They may approximate the target model or gather some information from the literature and the application domain. Next, we will have a brief introduce to these attack strategies.

3.1. Transferability (Substitute Model-Based) attack

In substitute model-based attack, the attacker creates a substitute model that approximates the target model. The method for building a substitute model varies with the threat model. In a black-box scenario, the attacker only observes the target model's predictions. They generate a synthetic dataset by querying the target model with input samples from both non-malicious and malicious classes and recording the outputs [5,6]. This dataset is then used to train the substitute model. In a graybox scenario, the attacker might have partial access, such as to the training data, which can be used to develop a more accurate substitute model [6]. Once a substitute model is constructed, whitebox attacks can be used to attack.

In Malware detection area, Rosenberg et al. [7] trained Gated Recurrent Unit (GRU) as the substitute model, and then performed the FGSM attack on the substitute model. Hu and Tan [8] used an RNN model which consist of a Gumbel-Softmax layer and a sequence decoder as substitude model, and trained a LST model based on this generative substitute RNN to reduce the accuracy of the target model.

Additionally, similar to Substitute Model-Based attack, GAN is also often used to train a generator which aims to generate adversarial samples to minimize the accuracy of the target model. In recent studies, GAN-based attacks have been cited and achieved significant results. Like MalGAN [9] attack framework proposed by Hu and Tan or GAPGAN [10] attack framework proposed by Yuan et al.

3.2. Query attack

Query attacks iteratively perturb an input sample toward a desired class based on feedback from the target model, functioning as gradient-free attacks in a black-box manner [11]. By observing changes in the predicted output due to small input perturbations, attackers can gauge their progress. In gray-box scenarios, increased knowledge allows attackers to guide attacks more effectively, such as by identifying which features most likely affect the decision boundary.

In malware detection, maintaining malicious functionality and using discrete feature vectors are crucial [2]. Thus, typical query attacks are less effective due to these constraints. To address this, recent research in malware detection proposes software transplantation-based techniques for query attacks [12]. This involves gradually perturbing a malware example with benign features to evade detection. This

DOI: 10.54254/2755-2721/71/20241665

method can be applied with varying levels of information about the target model. By transplanting benign features, the malware exhibits benign traits, leading to misclassification.

3.3. Universal adversarial perturbations

Recent research has found UAPs as an efficient method for creating adversarial examples. UAPs allow one perturbations set to be applied to different malware samples, generating adversarial samples with each application. If attackers know which perturbations sets are universal, they can apply them in multiple attacks, thus lowering the overall cost of the attack [13].

4. Defense mechanisms

4.1. Single-Model defenses (Model Robustness)

Although Single-Model Defenses mostly target white-box attacks, many black-box attacks are dependent on white-box attacks, so we thought it would be necessary to introduce this approach.

4.1.1. Randomization-Based defenses. Random feature nullification was proposed by Wang et al., reducing the attacker's possibility of manipulating important features to generate adversarial examples [14]. Its effectiveness doesn't easily transfer from images to malware, and it succeeds only when the attacker makes random perturbations [15].

4.1.2. Defensive distillation. Proposed by Papernot et al., this approach involves training a primary deep neural network (DNN) with definitive class labels, and then training a secondary DNN using the probabilities derived from the primary DNN's predictions and often results in a secondary DNN that generalizes better [16]. Though successful in image recognition, it is less effective in malware detection [17].

4.1.3. Adversarial training. Proposed by Szegedy et al., adversarial training enhances robustness against adversarial attacks by merging adversarial examples into the training data to enhance generalization has succeeded in ML malware detection model [17]. However, it is not inherently resistant to new types of attacks and requires ongoing updates with fresh adversarial examples [18].

4.2. Ensemble defenses

Ensemble defenses use several models in decision-making, aiming to enhance robustness, increase attack complexity, reduce variance, and improve prediction accuracy and generalization [19].

4.2.1. Voting defense. As show in Figure 2, voting defense is a static defense approach and has been applied in malware detection [20, 21]. An input is processed by multiple models, and the final prediction is determined through a voting mechanism [22].

In malware detection, Stokes et al. [15] propose an ensemble learning approach which enhanced the robustness of malware classifiers by aggregating predictions from multiple classifiers through voting, significantly reducing the success rates of white-box gradient-based attacks. Li and Li [23] introduce the adversarial deep ensemble defense method, which uses adversarial examples for retraining and applies saddle-point optimization to improve generalization, demonstrating effectiveness against multiple attack methods on Android datasets.

4.2.2. Moving target defenses(MTD). As show in Figure 2, MTDs regularly alter their configurations and randomize components to increase uncertainty of system, which makes it harder for attacks to break through defenses [24], consideres a type of ensemble defense, aims to avoid being static targets. In adversarial ML feild, MTDs can be implemented either dynamic or hybird [6], changing their configuration during prediction to prevent attackers from understanding how predictions are made.

DOI: 10.54254/2755-2721/71/20241665

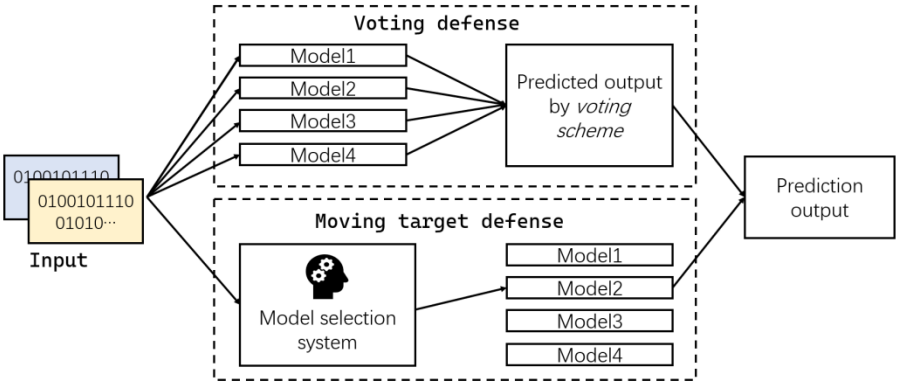

Figure 2. Comparison of Voting defense and Moving target defense

5. Discussion

5.1. Current research trends

A key area of study has been the development and evaluation of various black-box adversarial attacks, utilizing strategies like Evolution Strategies and other state-of-the-art techniques [25,26]. Additionally, he attack detection approach has recently been applied in the field of malware detection. Aqlib Rashid and Jose Such proposed Malprotect [27] stateful defense and demonstrates that MalProtect can notably lower the success rate of query attacks across different ranges and threat models.

Meanwhile, novel attacks are emerging. M. Xiong and C. Ye proposed a novel universal trigger attack method, which is introduced for API sequence-based malware detection [28]. Xiang Ling et al. proposed MalGuise [29], a practical black-box adversarial attack framework designed to evaluate the security risks of learning-based Windows malware detection systems achieve high attack success rate.

5.2. Open issues

One critical issue is the effectiveness and efficiency of current defense mechanisms. While various techniques such as signature-based detection, behavioral analysis, heuristics, machine learning, and anomaly detection have been integrated into multi-faceted security approaches, they remain insufficient in the face of zero-day exploits and advanced evasion tactics. The resource-intensive nature of dynamic analysis further complicates this landscape, exacerbated by the high prevalence of false positives and negatives, which undermine detection accuracy and reliability [30].

Robustness certification also presents a major challenge. While experimental validation of defenses is common, theoretical guarantees of their effectiveness remain scarce. The robustness metrics proposed, such as the CLEVER metric for estimating the lower bound of minimum adversarial perturbations, offer some computational feasibility, but their accuracy and reliability have been contested. As a result, there is a pressing need for more sophisticated analysis technologies and standardized risk assessment frameworks to evaluate and ensure model security throughout the AI lifecycle.

6. Conclusion

The landscape of malware detection has been significantly transformed by the integration of machine learning techniques, which have enhanced the accuracy and efficiency of identifying malicious software. However, this evolution has also introduced new vulnerabilities, particularly through adversarial attacks that exploit the weaknesses of ML models. This paper has provided a detailed review of adversarial attacks on black-box malware detection systems, highlighting the unique challenges posed by these attacks and the strategies employed to counter them.

The review underscores the importance of understanding the transferability and query-based strategies that adversaries use in black-box scenarios. It also emphasizes the need for robust defense mechanisms, such as adversarial training, ensemble methods, and model robustness techniques, to

DOI: 10.54254/2755-2721/71/20241665

mitigate the impact of these attacks. While significant progress has been made in developing these defenses, the dynamic and evolving nature of adversarial threats necessitates continuous research and innovation.

By focusing on the enhanced robustness of machine learning models and integrating them with traditional security measures, the academic and industrial communities can significantly improve the security and robustness of malware detection systems against evolving cyber threats.

References

[1] Giovanni Apruzzese, Pavel Laskov, Edgardo Montes de Oca, Wissam Mallouli, Luis Búrdalo Rapa, Athanasios Vasileios Grammatopoulos, and Fabio Di Franco. The role of machine learning in cybersecurity. Digital Threats, jul 2022. doi: 10.1145/3545574.

[2] K. Grosse, N. Papernot, P. Manoharan, M. Backes, and P. McDaniel. Adversarial perturbations againstdeep neural networks for malware classification. arXiv preprint arXiv:1606.04435, 2016.

[3] Edward Raff, Richard Zak, Gary Lopez Munoz, William Fleming, Hyrum S Anderson, Bobby Filar, Charles Nicholas, and James Holt. Automatic yara rule generation using biclustering. In Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security, 71–82, 2020.

[4] R. Uda and S. Araki. Investigation of The Latest Malware Detection Engines and Lightweight Byte n-Gram Methods with Real Custom Malware, 2024 16th International Conference on Computer and Automation Engineering (ICCAE), Melbourne, Australia, 2024, pp. 6-11, doi: 10.1109/ICCAE59995.2024.10569568.

[5] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z.B. Celik, and A. Swami. Practical black-box attacks against machine learning. Proceedings of the 2017 ACM on Asia conference on computer and communications security, pp.506–519, 2018.

[6] Aqib Rashid and Jose Such. Effectiveness of moving target defenses for adversarial attacks in ml-based malware detection. arXiv preprint arXiv:2302.00537, 2023.

[7] I. Rosenberg, A. Shabtai, L. Rokach, and Y. Elovici, Generic black box end-to-end attack against state of the art API call based malware classifiers, in Proc. RAID, Heraklion, Greece, Sep. 2018, pp. 490–510.

[8] W. Hu and Y. Tan. Black-box attacks against RNN based malware detection algorithms, in Proc. AAAI Workshops, New Orleans, LA, USA, Feb. 2018, pp. 245–251.

[9] W. Hu and Y. Tan, Generating adversarial malware examples for black-box attacks based on GAN, 2017, arXiv:1702.05983.

[10] J. Yuan, S. Zhou, L. Lin, F. Wang, and J. Cui, Black-box adversarial attacks against deep learning based malware binaries detection with GAN, in Proc. ECAI, Santiago de Compostela, Spain, Aug. 2020, pp. 2536–2542.

[11] Jianbo Chen, Michael I Jordan, and Martin J Wainwright. Hopskipjumpattack: A query-efficient decision-based attack. In 2020 IEEE Symposium on Security and Privacy (SP), 1277–1294. IEEE, 2020.

[12] Ishai Rosenberg, Asaf Shabtai, Yuval Elovici, and Lior Rokach. Query-efficient black-box attack against sequence-based malware classifiers. In Annual Computer Security Applications Conference, pp.611–626, 2020.

[13] Raphael Labaca-Castro, Luis Muñoz-González, Feargus Pendlebury, Gabi Dreo Rodosek, Fabio Pierazzi, and Lorenzo Cavallaro. Universal adversarial perturbations for malware. arXiv preprint arXiv:2102.06747, 2021.

[14] Qinglong Wang, Wenbo Guo, Kaixuan Zhang, Xinyu Xing, C Lee Giles, and Xue Liu. Random feature nullification for adversary resistant deep architecture. arXiv preprint arXiv:1610.01239, 2016.

[15] Jack W. Stokes, De Wang, Mady Marinescu, Marc Marino, and Brian Bussone. Attack and defense of dynamic analysis-based, adversarial neural malware classification models. arXiv preprint arXiv:1712.05919, 2017.

DOI: 10.54254/2755-2721/71/20241665

[16] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In 2016 IEEE Symposium on Security andPrivacy (SP), pp.582–597. IEEE, 2016.
[17] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In 2017 IEEE symposium on security and privacy (SP), pp.3–18. IEEE, 2017
[18] Ian Goodfellow. A research agenda: Dynamic models to defend against correlated attacks. arXiv preprint arXiv:1903.06293, 2019.
[19] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. Nature, 521(7553):436, 2015.
[20] Minhao Cheng, Cho-Jui Hsieh, Inderjit Dhillon, et al. Voting based ensemble improves robustness of defensive models. arXiv preprint, arXiv:2011.14031, 2020.
[21] Suleiman Y. Yerima and Sakir Sezer. Droidfusion: A novel multilevel classifier fusion approach for android malware detection. IEEE transactions on cybernetics, 49(2):453–466, 2018.
[22] Todd Jackson, Christian Wimmer, Michael Franz. Multi-variant program execution for vulnerability detection and analysis. In Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research, pp.1–4, 2010.
[23] D. Li and Q. Li, Adversarial deep ensemble: Evasion attacks and defenses for malware detection, IEEE Trans. Inf. Forensics Security, vol. 15, pp. 3886–3900, 2020.
[24] Jin-Hee Cho, Dilli P. Sharma, Hooman Alavizadeh, Seunghyun Yoon, Noam Ben-Asher, Terrence J. Moore, Dong Seong Kim,Hyuk Lim, Frederica F Nelson. Toward proactive, adaptive defense: A survey on moving target defense. IEEE Communications Surveys & Tutorials, 22(1):709–745, 2020.
[25] Shuai Zhou, Chi Liu, Dayong Ye, Tianqing Zhu, Wanlei Zhou, and Philip S. Yu. 2022. Adversarial Attacks and Defenses in Deep Learning: From a Perspective of Cybersecurity. ACM Comput. Surv. 55, 8, Article 163 (August 2023), pp.39.
[26] Huiying Li, Shawn Shan, Emily Wenger, Jiayun Zhang, Haitao Zheng, Ben Y. Zhao. (2022). Blacklight: Scalable Defense for Neural Networks against Query-Based Black-Box Attacks. In Proceedings of 31th USENIX Security Symposium (USENIX Security'2022).
[27] A. Rashid and J. Such. MalProtect: Stateful Defense Against Adversarial Query Attacks in ML-Based Malware Detection, in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 4361-4376, 2023, doi: 10.1109/TIFS.2023.3293959.
[28] Mengying Xiong, Chunyang Ye. Universal adversarial triggers for attacking against API sequence: based malware detector. Fourth International Conference on Signal Processing and Machine Learning (CONF-SPML 2024). Vol. 13077. SPIE, 2024.
[29] Ling, Xiang, et al. A Wolf in Sheep's Clothing: Practical Black-box Adversarial Attacks for Evading Learning-based Windows Malware Detection in the Wild. arXiv preprint, arXiv:2407.02886 (2024).
[30] Qiu, Hao, Leonardo Lucio Custode, and Giovanni Iacca. Black-box adversarial attacks using evolution strategies. Proceedings of the Genetic and Evolutionary Computation Conference Companion, 2021.