Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 6, No.3, June : 2025

SMART ATTACKS FOR SMART DEVICES: EVASION TECHNIQUES AGAINST IOT MALWARE DETECTORS

J. VidhyaJanani, Assistant Professor, Department of Computer Science and Engineering, Paavai College of Engineering, Namakkal.
G. Vigneshwaran, Student**,** Department of CSE (Artificial Intelligence and Machine Learning), Paavai College of Engineering, Namakkal
B. Devika, Student**,** Department of CSE (Artificial Intelligence and Machine Learning), Paavai College of Engineering, Namakkal
G. Kanimozhi, Student**,** Department of CSE (Artificial Intelligence and Machine Learning), Paavai College of Engineering, Namakkal
R. Lavanya, Student**,** Department of CSE (Artificial Intelligence and Machine Learning), Paavai College of Engineering, Namakkal
A.Keerthana, Student**,** Department of CSE (Artificial Intelligence and Machine Learning), Paavai College of Engineering, Namakkal
M. Monisharma, Student**,** Department of CSE (Artificial Intelligence and Machine Learning), Paavai College of Engineering, Namakkal
S. Kaviya, Student**,** Department of CSE (Artificial Intelligence and Machine Learning), Paavai College of Engineering, Namakkal

Abstract
The proliferation of Internet of Things (IoT) devices has introduced new vectors for cyberattacks, making robust malware detection mechanisms crucial for securing smart environments. Recent advancements in machine learning (ML) have significantly improved the accuracy of IoT malware detection. However, these systems remain vulnerable to adversarial attacks that exploit their inherent weaknesses. This paper explores the use of the Fast Gradient Sign Method (FGSM), a popular adversarial attack technique, to evade ML-based IoT malware detectors. By generating adversarial perturbations that subtly modify malware feature representations, we demonstrate how such manipulated inputs can successfully bypass detection models without compromising malicious functionality. Through comprehensive experiments on benchmark datasets and commonly used ML classifiers, our results highlight the susceptibility of current detection systems to FGSM-based attacks. The study underscores the urgent need for robust adversarial defenses in IoT malware detection pipelines and contributes toward understanding the limitations of machine learning models in hostile environments. These models are susceptible to adversarial assaults, though, which quietly alter inputs in order to avoid discovery. In order to create adversarial instances that avoid ML-based IoT malware detectors, this study investigates the use of the Fast Gradient Sign Method (FGSM). We show that even small perturbations that are invisible to human analysts may drastically reduce the accuracy of popular classifiers like deep neural networks, decision trees, and random forests. Our tests on an actual IoT malware dataset show that FGSM can lower detection rates by as much as 93% without affecting the malware's essential functioning. These findings underline the necessity of strong adversarial defenses in ML-based cybersecurity applications and point to a serious weakness in the present IoT malware protection systems.

Keywords:
Machine Learning, Cyber attacks, Fast Gradient Sign Method, Internet of Things, Clever hands algorithm, Principal Computational analysis

I. Introduction

Smarter homes, cities, healthcare systems, and industrial processes have all been made possible by the widespread use of Internet of Things (IoT) devices. But the attack surface for bad actors has also

increased as a result of this quick adoption. IoT devices are especially susceptible to malware assaults because they frequently lack the processing power to enable conventional endpoint security measures. Machine learning (ML)-based malware detection systems have become more and more popular as a means of combating this threat because of their capacity to identify new threats and learn from trends. In order to categorize apps or firmware as dangerous or benign, these systems usually use static or behavioral characteristics that are taken from network traffic or device activity.

In order to combat malware and supplement conventional signature-based and heuristic-based detection techniques, anti-malware engines now commonly use Machine Learning (ML) techniques as part of a multilayered detection system. Static ML-based malware detectors, which are ML-based malware detectors trained using data about computer files acquired by static analysis—that is, the examination of computer programs without running them—are the subject of our study. Static ML-based malware detectors may be broadly divided into two groups: (1) end-to-end detectors and (2) feature-based detectors.In order to extract a collection of features that are used to describe the executables, feature-based detectors [1] mostly rely on domain expertise. This process takes a lot of time and necessitates a thorough understanding of the assembly code and file structure of the executable. The process of feature engineering is ongoing. Even if they work well, ML-based models are not always reliable. Critical flaws in these systems have been shown by adversarial machine learning, which demonstrates how well-designed perturbations that are invisible to humans may drastically impair the model's functionality. The Fast Gradient Sign Method (FGSM) is one of the most well-known methods for creating these adversarial situations. FGSM successfully fools the classifier with little computation by perturbing input data in the direction of the gradient of the loss function with respect to the input.

To avoid detection, malware developers constantly tweak and alter their dangerous code. As a result, in the future, new features may be needed, and outdated ones may be weaponized to avoid detection [2, 3]. Consequently, recent studies have focused on developing models that can extract features on their own, such as end-to-end or deep learning-based detectors [4, 5, 6]. Injecting material from benign instances into malicious executables is a simple yet efficient way to get around end-to-end detectors [7, 8]. The "benign" byte patterns discovered in the adversarial malware instances may therefore cause end-to-end detectors' categorization output to change from harmful to benign. Additionally, researchers have created complex assaults that optimize and insert tiny hostile payloads into malicious executables, resulting in adversarial malware samples that are hardly altered yet nevertheless manage to avoid detection. These assaults can be classified as either black-box [8, 11] or white-box [9, 10] attacks, depending on the models' access.

The following are this work's primary contributions:

• We suggest a strong protection against functionality-preserving content modification attacks that is independent of model.

• We present two chunk-based ablation techniques created especially for malware detection.

• Using the BODMAS dataset, we empirically evaluate the state-of-the-art evasion strategies on deep learning malware detection models, demonstrating that the suggested smoothing strategy is more resilient to these assaults than a baseline classifier.

This is how the remainder of the paper is structured. An outline of the functionality-preserving assaults created especially to counter deep learning-based malware detectors and the responses that have been created thus far is given in Section 2. In Section 3, two chunk-based techniques created especially for the malware detection job are introduced, together with our (de)randomized smoothing technique for protection against adversarial malware instances. The suggested defense system is assessed in Section 4 against a number of cutting-edge evasion attempts. Section 5 concludes by summarizing our final thoughts and outlining some potential research directions.

II. Literature Review

Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM),

SVM with grid search (SVMG), K-Nearest Neighbor (KNN), and Naïve Bayes (NB) are among the machine learning models that Dubey A. K. et al. studied for heart disease detection. Training and testing were conducted using the UCI Machine Learning repository's Cleveland and Statlog datasets. LR and SVM classifier models outperform the Cleveland dataset with 89% accuracy, according to the experimental results, while LR outperforms the Statlog dataset with 93% accuracy [7]. Karthick K. et al. developed an ML model for predicting the risk of heart disease using the SVM, Gaussian Naive Bayes (GNB), LR, LightGBM, XGBoost, and RF algorithms. To choose the top characteristics from the Cleveland heart disease dataset, the authors of this study used the Chi-square statistical test. The RF classifier model had the greatest classification accuracy rate of 88.5% following feature selection [8].

The use of adversarial training to strengthen the resilience of deep learning-based malware detectors against specific adversarial attacks was investigated by Lucas et al. [12]. According to their findings, state-of-the-art assaults are not deterred by data augmentation alone. However, in certain situations, adversarially training models with less complex or low-effort versions of the same assaults might make them marginally more resistant to attacks. For example, they used adversarial samples produced by three assaults to adversarially train a deep learning-based detector: (1) In-Place Replacement attack (IPR), (2) Displacement attack (Disp) [13], and (3) Padding attack [7]. that blends displacement with in-place replacement. According to their review, the original model's accuracy was 25%, but the adversarially trained model had an accuracy of 49%.

A randomized smoothing approach was suggested by Gibert et al. [14] as an alternative to strengthen end-to-end malware classifiers' resistance to adversarial malware instances. In order to make the classifier's predictions steady and resistant to minor disturbances, randomized smoothing adds random noise to the input data during both training and inference. This method strengthens the resilience of machine learning-based classifiers against adversarial assaults. They train a basic malware classifier f to classify files using an ablated version of the infection.

A replica of the original executable x with the bytes ablated according to a probability p makes up x~ of a given executable file x. However, randomized smoothing has limits when used to protect against adversarial malware samples, even if it works well against some kinds of adversarial attacks. This is because adversarial assaults in the malware domain are different from attacks or disruptions in picture or text data due to the special difficulties created by the limitations imposed by executable files. In particular, rather than altering bytes arbitrarily, attackers frequently insert an adversarial payload into certain sections of executable files when it comes to malware detection.

Making the computer learn about the issue statement is known as machine learning. By providing it with facts and knowledge via observations and real-world encounters, we enable it to learn. Gupta et al. [3] employed data from the VXheaven, Nothing, and VirusShare datasets on models and performed 10-fold cross-validation to the malware detection architecture they suggested. A variety of classification techniques, such as random forest, naive Bayes, and support vector machine, were employed. The accuracy, precision, TPR, FPR, and FNR were used to assess the performance.

Burnap et al. (2020) [2] introduced a new method that uses self-organizing feature maps to categorize files and lessen overfitting between malicious and safe files that happens during dataset training. The VirusTotal API is used to collect this information. In this work, classifiers such as Random Forest, BayesNet, MLP, and SVM are presented. With a score of 98 percent, the random forest classifier has the best accuracy. However, when applied to different datasets, it was 12% lower because to overfitting. The Self-Organizing Feature Map (SOFM) software, a classifier based on the ANN approach, was used to solve the overfitting issue, and the accuracy rose by 7%.

III. Dataset and Feature Selection

Machine learning-enabled detection systems often respond to evasion assaults by examining the attack and retraining the model using the fresh information gathered. The technology can thwart the attacker's plan the next time they employ a similar tactic. In the aforementioned situation, this system functions

well, but what happens if it is attacked? A kind of attack where the model learns to misclassify by feeding it an example? These attacks are referred to as adversarial attacks and are becoming more and more dangerous in the field of security. Only what is known as adversarial machine learning can overcome such a problem [1]. Table 1 shows an overview of the existing systems and also discusses the AI algorithms used in these systems.

Table 1: survey of existing papers

Paper	Algorithms Used	Dataset Sources	Results
[1]	Naive Bayes methodology, SVM, random forest	VX Heaven, Virus, Share, Nothin K.	The Random Forest gives the best accuracy
[2]	Random forest, BN, MLP, SVM, SOFM	Virus total AP	Performance increased by 7.24% to 25.68%
[3]	KNN, random forest	CTU13, Stratosphere IPS project	Performance increased. An accuracy of 95.5% was accomplished
[4]	Clustering algorithms	Collected from Malicious website	Expectation maximization techniques give a high accuracy
[5]	Shared nearest neighbor (SNN)	Kingsoft, ESET NOD3 2, and Anubis	98.9% accuracy of known malware and 86.7% of accuracy for unknown malware detected
[6]	Stochastic gradient, multilayer perceptron, random forest, decision tree, nearest centroid and perceptron	Making a dataset	Improved accuracy results using random forest algorithms
[7]	MLP, DT, IB1, random forest	University of California	Random Forest had high accuracy values of 99.58 percent
[8]	Random forest, IB1, DT, support	CA's (Computer Associates) VETZoo	Better accuracy and improved performance by 9% using random forest

IV. Adversarial machine learning

When even a minor opponent may alter their inputs, machine learning algorithms that were designed to presume a benign environment fail. Adversarial machines are useful in this situation [12]. The field of machine learning known as "adversarial machine learning" examines a series of attacks designed to impair classifiers' performance on certain tasks. The durability of the machine learning model is guaranteed by adversarial machine learning [16,17].

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 6, No.3, June : 2025

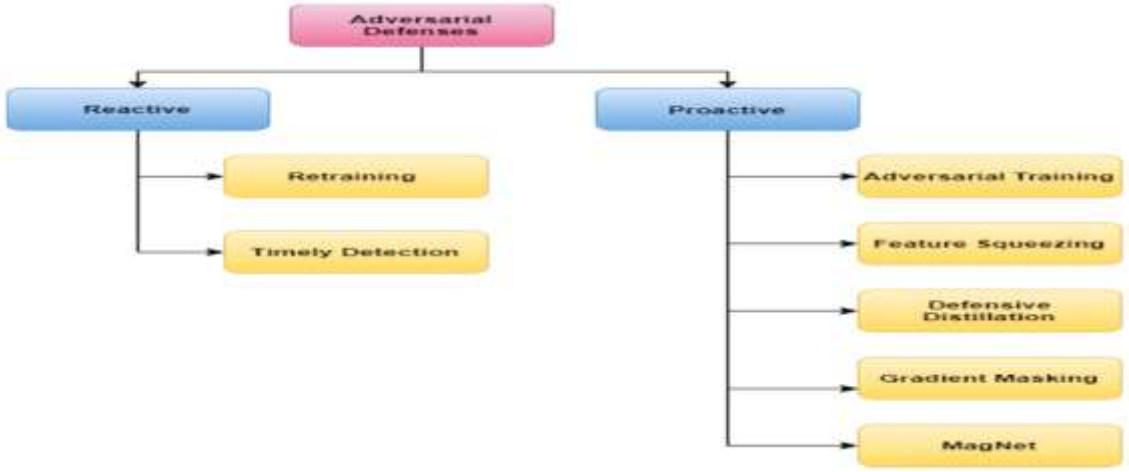

Figure 1:Diffreent types of adversarial Defences and its classification

The taxonomy of adversarial assaults and countermeasures is displayed in Figure1. With a trade-off between several attributes, such as performance, complexity, computational efficiency, and application circumstance, a variety of attack strategies and methodologies, including black box and white box assaults, have been researched and used to detect malware [15]. White box attacks occur when the attacker has complete knowledge and expertise about how a certain model operates internally, including training data, model parameters, and other pertinent classifier information. Black-box attacks, on the other hand, occur when the attacker lacks inquisitive access to the model and is unable to understand how it functions within.

Jin-Young Kim et al. [5] suggested a malware detection method based on autoencoders and GANs. A standard malware challenge dataset that was available on Kaggle was utilized. Their proposed approach outperformed previous machine learning models, including support vector machines, KNN, random forest, MLP, and naive Bayes, in detecting adversarial attacks. Joseph Clements et al. employed four adversarial attacks—FGSM, ENM, JSMA, and C&W—to identify malware using the Kitsune network IDS (NIDS) classifier on the Mirai botnet dataset.

V. IMPLEMENTATION

We initially completed the MaleVis dataset, an open-source malware dataset, as the initial stage of data collecting [5]. It includes byte pictures of one valid class and twenty-five malicious classes. This dataset was created by utilizing Sultanik's bin2png tool to convert malware binary files into three-channel RGB pictures. There are two square-sized resolutions of this dataset: 224 × 224 pixels and 300 × 300 pixels. There are 5126 RGB validation pictures and 9100 training images in the MaleVis dataset. With 350 photos apiece, every lesson in the training package is precisely balanced. However, the number of photos in the validation set varies. With 1482 photos, the valid class in the validation set is bigger. This is due to the fact that malware detection relies on separating the genuine from the malicious pictures. Machine learning includes deep learning. Artificial neural networks and deep networks of artificial neurons allow us to extract more information from incoming data. Beginning at the lowest level and working its way up, deep learning begins with the raw data. The neural network may acquire high-level characteristics as it becomes more complicated, but in image classification, for example, the lower levels would identify the edges of the item in the picture. Numerous deep learning methods exist for various types of problems. For this study, the scientists decided to use a convolutional neural network.

The procedures listed in the algorithm were employed in this study and research analysis to create the adversarial pictures for the Malevis dataset. A number of input photos were sorted into batches according to their class, taken as samples, and run through the FGSM model for a range of epsilon values. The antagonistic samples were then produced. For this project and study, the sample was generated using three different epsilon values: 0.01; 0.1; and 0.15. In order to make sure that the

perturbations are both tiny enough for the human eye to miss them and large enough to trick the trained models, epsilon values are values multiplied by the signed gradients.

The FGSM adversarial attack was then used to target the classifiers that the authors had created. For each class, adversarial samples with varying epsilon values—0.01, 0.1, and 0.15—were created and then fed into the model to see how it behaved in comparison. The outcomes of the adversarial assault are displayed in Figure 3 Below is an explanation of the attack's outcomes.

With an epsilon value of 0.1 and above, it was shown that the random forest classifier was comparatively more vulnerable to the FGSM attack. The confidence level was considerably reduced with the model's 0.01 epsilon value, but it did not misclassify for the maximum number of classes. The epsilon value of 0.01 had a 68% confidence level for the Vilsel class, which was classified with 100% correctness prior to the assault. While the Hlux class's accuracy was 83.66%, which was 100% before to the attack, the class's overall accuracy for the epsilon value 0.1 decreased by almost 16%, from 100% to 84.52%. The average accuracy reduction of the 0.15 epsilon assault was 40%, which made it lethal.

VI. Conclusion

The use of AI has enhanced the functionality of malware detection systems. Nonetheless, there are still questions regarding these categorization models' security. This study suggests a malware detection system design that makes use of adversarial training and machine learning. With the use of machine learning, deep learning, and a pretrained model, the authors have developed a malware classification system that has an accuracy of 93% for random forests, 92.3% for CNN, 93.7% for efficient nets, and 92% for VGG-16. The authors then used photos with 0.01, 0.1, and 0.15 epsilon values to conduct a FGSM attack on the EfficientNet model. The findings were successfully misclassified by the model. The system will be resilient to the FGSM adversarial assault since this model won't misclassify the outcomes when trained against these hostile samples. Adversarial training will help the system become more resilient throughout the detection process, and the machine learning model will help detect malicious files. Adversarial assaults were able to show that the model is susceptible to adversaries using the suggested system. The objective was to develop a trained model that would not falter in the face of an enemy. Future studies would use different types of assaults that are accessible, train the model to withstand those attacks, and then make it even more resilient.

Last but not least, the authors are creating a smartphone app that will enable users to input symptoms and make fast and precise predictions about cardiac disease. In order to anticipate heart illness and provide the detection result immediately, we will include the best XGBoost technology into the mobile app. Since the mobile app predicts cardiac disease based on symptoms, we will take into account and deal with the influence of "dark data" throughout its deployment. Dark data is information that is available but is either underused or not collected because of poor reporting, ignorance, or constraints in data collecting.

References

1. World Health Organization. Cardiovascular Diseases (CVDs). Available online: (2023
2. Alom, Z. et al. Early Stage Detection of Heart Failure Using Machine Learning Techniques. In Proceedings of the International Conference on Big Data, IoT, and Machine Learning, Cox’s Bazar, Bangladesh, 23–25 September (2021).
3. Gour, S., Panwar, P., Dwivedi, D. & Mali, C. A machine learning approach for heart attack prediction. In Intelligent Sustainable Systems (eds Nagar, A. K., Jat, D. S., Marín-Raventós, G. & Mishra, D. K.) 741–747 (Springer, Singapore, 2022). Nguyen T., Wang Z.A. Cardiovascular screening and early detection of heart disease in adults with chronic kidney disease. J. Nurse Pract. 2019;15:34–40. doi: 10.1016/j.nurpra.2018.08.004.
4. Pasha, S.J.; Mohamed, E.S. Novel Feature Reduction (NFR) Model with Machine Learning and Data Mining Algorithms for Effective Disease Risk Prediction. IEEE Access 2020, 8, 184087–

184108.
5. Swain, D.; Pani, S.K.; Swain, D. A Metaphoric Investigation on Prediction of Heart Disease using Machine Learning. In Proceedings of the 2018 International Conference on Advanced Computation and Telecommunication, ICACAT, Bhopal, India, 28–29 December 2018; pp. 1–6.
6. Dubey A.K., Choudhary K., Sharma R. Predicting Heart Disease Based on Influential Features with Machine Learning. Intell. Autom. Soft Comput. 2021;30:929–943. doi: 10.32604/iasc.2021.018382.
7. Karthick K., Aruna S.K., Samikannu R., Kuppusamy R., Teekaraman Y., Thelkar A.R. Implementation of a heart disease risk prediction model using machine learning. Comput. Math. Methods Med. 2022;2022:6517716. doi: 10.1155/2022/6517716.
8. Veisi H., Ghaedsharaf H.R., Ebrahimi M. Improving the Performance of Machine Learning Algorithms for Heart Disease Diagnosis by Optimizing Data and Features. Soft Comput. J. 2021;8:70–85.
9. Sarra R.R., Dinar A.M., Mohammed M.A., Abdulkareem K.H. Enhanced heart disease prediction based on machine learning and χ2 statistical optimal feature selection model. Designs. 2022;6:87. doi: 10.3390/designs6050087.
10. Singh A., Kumar R. Heart disease prediction using machine learning algorithms; Proceedings of the 2020 International Conference on Electrical and Electronics Engineering (ICE3); Gorakhpur, India. 14–15 February 2020; pp. 452–457.
11. Sahoo G.K., Kanike K., Das S.K., Singh P. Machine Learning-Based Heart Disease Prediction: A Study for Home Personalized Care; Proceedings of the 2022 IEEE 32nd International Workshop on Machine Learning for Signal Processing (MLSP); Xi'an, China. 22–25 August 2022; pp. 1–6.
12. Khdair H. Exploring Machine Learning Techniques for Coronary Heart Disease Prediction. 2021. [(accessed on 12 April 2023)].
13. Ahmad G.N., Fatima H., Abbas M., Rahman O., Alqahtani M.S. Mixed machine learning approach for efficient prediction of human heart disease by identifying the numerical and categorical features. Appl. Sci. 2022;12:7449. doi: 10.3390/app12157449.
14. Chou J.-S., Truong D.-N. A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean. Appl. Math. Comput. 2021;389:125535. doi: 10.1016/j.amc.2020.125535
15. Acharya U.R., Fujita H., Oh S.L., Raghavendra U., Tan J.H., Adam M., Gertych A., Hagiwara Y. Automated identification of shockable and non-shockable life-threatening ventricular arrhythmias using convolutional neural network. Futur. Gener. Comput. Syst. 2018;79:952–959. doi: 10.1016/j.future.2017.08.039.
16. 27.Yao Q., Wang R., Fan X., Liu J., Li Y. Multi-class arrhythmia detection from 12-lead varied-length ECG using attention-based time-incremental convolutional neural network. Inf. Fusion. 2020;53:174–182. doi: 10.1016/j.inffus.2019.06.024.