A Generative Adversarial Network Technique for Ransomware Behavior Prediction

Subjects: Computer Science, Information Systems
Contributor: Mazen Gazzan, Frederick T. Sheldon

The ransomware attacks threaten not only personal files but also critical infrastructure like smart grids, necessitating early detection before encryption occurs. Current methods, reliant on pre-encryption data, suffer from insufficient and rapidly outdated attack patterns, despite efforts to focus on select features. Such an approach assumes that the same features remain unchanged. This approach proves ineffective due to the polymorphic and metamorphic characteristics of ransomware, which generate unique attack patterns for each new target, particularly in the pre-encryption phase where evasiveness is prioritized.

Keywords: ransomware ; Generative Adversarial Network ; minimax loss function ; ransomware detection and prediction ; deep learning

1. Introduction

Like other cyberattacks, ransomware attacks target a variety of systems and networks, including Personal Computers (PCs), mobile devices, Wireless Sensor Networks (WSN), Vehicular Ad Hoc Networks (VANETs), and the Internet of Things (IoT) [1][2]. Several studies have been conducted to detect ransomware attacks [3][4][5]. To detect crypto-ransomware early, the data collected during the pre-encryption phase of the crypto-ransomware lifecycle, before the encryption takes place is used [6][7]. The collected data are then used to train different machine learning algorithms to classify the programs into benign and ransomware [8]. However, the lack of sufficient data during the early phases of the attack adversely affects the accuracy of the model due to insufficient attack patterns [9].

Currently, ransomware attacks have targeted many Cyber Physical Systems (CPS), causing severe disruption of critical services and infrastructure [10][11]. In 2021, the US faced two significant CPS ransomware attacks on its critical infrastructure. The Colonial Pipeline, a major fuel supplier for the East Coast, experienced a cyberattack in May, leading to fuel shortages and panic buying in various states [12]. Then, in June, JBS, the world's top meat supplier, was attacked, prompting plant shutdowns in the US and Australia. This attack utilized the Ryuk ransomware, demanding millions in ransom. Colonial Pipeline and JBS suffered significant financial losses, paying ransoms of $4.4 million and $11 million, respectively [12]. In October 2021, the Czech Republic's major power company, CEZ, was attacked with RansomExx ransomware after an intrusion via Winnti malware, causing power outages. Earlier, in December 2020, a natural gas facility was targeted using the TrickBot malware variant, prompting a response from the Cybersecurity and Infrastructure Security Agency (CISA) [10]. These attacks underline the severe consequences of ransomware on critical infrastructure, emphasizing the need for enhanced cybersecurity, and regular system updates, underscoring the significance of addressing vulnerabilities in CPS.

The insufficient attack patterns are the main obstacle that degrades the early detection accuracy of ransomware attacks. Although several studies tried to overcome data insufficiency by focusing on how to select a subset of features that represent the immature ransomware attack patterns. Such approach assumes that the significance of those features remains unchanged. This does not hold as the polymorphic and metamorphic nature of the attack makes the ransomware generate different patterns every time it receives a new target. This is especially true during the pre-encryption stage where the goal of ransomware is to be evasive. Hence, the features become quickly obsolete. GAN has the potential to overcome the data insufficiency problem by augmenting the real attack patterns with artificial, yet realistic data. However, the Minimax loss function used by GANS's generator and discriminator is unable to estimate the distance between the probability distribution of real and artificial instances in the pre-encryption data of ransomware attacks.

2. Generative Adversarial Network for Ransomware Behavior Prediction

The major obstacle in the early detection of ransomware involves obtaining adequate data in the pre-encryption stage when the attack is still being set up and has not yet been executed [1][2]. Addressing this data shortage is crucial, as a sufficient dataset is needed for precise early detection. Data augmentation is often used in machine learning solutions and presents a promising way to tackle this scarcity of data, a problem commonly faced by malware and ransomware early detection systems. To our understanding, no existing studies specifically address data augmentation in the pre-encryption stage of ransomware attacks [10]. Another challenge stems from the ever-changing nature of ransomware, which complicates the relevancy of features used for detection models [13]. For example, an attack pattern seen in one ransomware variant at a specific time might be more relevant than the same pattern displayed by a different variant at another time. This indicates that the importance of features can vary depending on the ransomware variant and the timing of the attack. Despite this, current early detection methods often operate on the assumption that the importance of these features remains constant, leading to “behavioral drift.” This drift mostly results in detection systems becoming quickly outdated and less accurate over time.

The Generative Adversarial Network (GAN) has been widely used as an important component of deep neural networks [14]. The GAN model has gained massive attention from researchers recently due to its prominent characteristics. It has two main rewards for machine learning based models: the generality and adversarial [15]. It can generate new samples that can be used to prevent overfitting and, thus, improve machine learning performance. Moreover, it can be used to generate adversarial samples that can be used to improve the discriminability of the model. GAN use alternative training to estimate the density function over a data distribution using the Minimax algorithm [16]. The Minimax game algorithm tries to minimize the maximum possible loss which results in multiple possibilities that can be used to generate new samples. In doing so, GAN projects the available simple distribution to a much more complex high-dimensional, real-world data distribution [17]. GAN trains two adversarial networks called the generator and the discriminator. The generator is trained to map noise samples to synthetic samples with the goal is to generate new adversarial samples that can mislead the discriminator. Meanwhile, the discriminator trains to distinguish the real data samples from synthesized samples that were generated using the generator. GAN creates the new samples by making small changes to the original samples so as to deceive the detection model gain benefit of the nonlinear characteristics of neural networks and thus constructs a model that produces incorrect classification results.

Due to its prominent features, many researchers have applied the GAN algorithm to improve the classification performance of machine learning algorithms. Moti and Hashemi [14] proposed a malware detection model for Internet of Things (IoT) using the Generative Adversarial Network technique and Convolutional Neural Network (CNN). CNN was used to extract high-level features while GAN was used to generate new malware samples to mitigate the limitations of availability of insufficient malware samples in IoT. Li and Zhou [18] utilized GAN to develop a malware detection model-based adversarial example for the Android platform. Their proposed model called bi-objective GAN can generate evasive adversarial-example attacks able to fool the firewall and evade detection. Lu and Li [19] used GAN to improve the classification accuracy of the malware detection by generating new samples that can mimic realistic-like malware samples as well as the realistic distribution of data. Zhang and Zhou [16], proposed an improved Monte Carlo tree search (MCTS) algorithm for generating adversarial examples of cross-site scripting (XSS) attacks. A reward value is generated by the MCTS to rank the generated adversarial examples. The GAN algorithm was used to improve the detectability of adversarial examples. A GAN-based network was proposed to improve classification performance.

The following paragraph explains how GAN works. GAN formulates the adversarial problem as follows. Let X denote the sample space, x is a benign sample, and $g(x)$>0 denotes the classification function when the result is benign. The attacker aims to generate a malware sample x^* that make $g(x^*)$>0. Thus, the aim of the attacker can be formulated as follows:

$$x^* = \arg max_x \hat{g}(x), s.t. d\left(x, x^*\right) \leq d_{max}. \quad (1)$$

The GAN reduces the loss function value V during the training of both generator $\mathscr{G}$ and discriminator $\mathscr{D}$ by solving the following optimization function:

$$\min_{\mathscr{D}} \max_{\mathscr{G}} V(\mathscr{G}, \mathscr{D}) \quad (2)$$

where

$$V(\mathscr{G}, \mathscr{D}) = E_x[\log(D(x))] + E_z[\log(1 - D(G(z)))]. \quad (3)$$

The Z denotes the samples from noise distribution. Although the existing GAN has been effectively used to improve the performance of malware detection models, it does not fully fit for ransomware early detection due to data insufficiency that makes it difficult to perceive a clear probability distribution of the data. The unclear probability distribution prevents the GAN's generator from creating artificial samples as the discriminator will discard them due to the large distance between the probability distribution of artificial data and real data. GAN According to Dumoulin and Belghazi [20] and Uehara and Sato [21], existing GAN algorithms suffer from a vanishing gradient problem which leads to instability and model collapse due to the use of predefined adversarial loss function. Haloui and Gupta [22] used the derived approximation to the Wasserstein distance to improve the original GAN gradient-based loss function. The improved GAN algorithm is called WGAN. WGAN relies on the Arjovsky k-Lipschitz continuous function which adversely reduce the capacity of the discriminator model [23]. Gulrajani and Ahmed [5] anticipated an enhanced WGAN algorithm that penalizes the norm of discriminator gradients to train the discriminator network with respect to the sample data. There are several structure GAN algorithms including fully connected GANs [6], Conditional GANs [24], Convolutional GANs [25], GANs with inference models [24], and adversarial autoencoders [26]. Most of these algorithms use the standard loss function which suffers from the vanishing gradient problem and, thus, leads to instability and model collabs especially when insufficient data is used for training the classification task. Such limitations hinder the applications of the GAN algorithm to many challenging domains in cybersecurity such as early detection of ransomware attacks.

References

1. Dumoulin, V.; Belghazi, I.; Poole, B.; Mastropietro, O.; Lamb, A.; Arjovsky, M.; Courville, A. Adversarially learned inference. arXiv 2016, arXiv:1606.00704.
2. Uehara, M.; Sato, I.; Suzuki, M.; Nakayama, K.; Matsuo, Y. Generative adversarial nets from a density ratio estimation perspective. arXiv 2016, arXiv:1610.02920.
3. Haloui, I.; Gupta, J.S.; Feuillard, V. Anomaly detection with Wasserstein GAN. arXiv 2018, arXiv:1812.02463.
4. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances in Neural Information Processing Systems 27 (NIPS 2014), Montreal, QC, Canada, 8–13 December 2014.
5. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A.C. Improved training of wasserstein gans. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
6. Barua, S.; Erfani, S.M.; Bailey, J. FCC-GAN: A fully connected and convolutional net architecture for GANs. arXiv 2019, arXiv:1905.02417.
7. Li, M.; Lin, J.; Meng, C.; Ermon, S.; Han, S.; Zhu, J.Y. Efficient spatially sparse inference for conditional gans and diffusion models. Adv. Neural Inf. Process. Syst. 2022, 35, 28858–28873.
8. Torfi, A.; Fox, E.A.; Reddy, C.K. Differentially private synthetic medical data generation using convolutional GANs. Inf. Sci. 2022, 586, 485–500.
9. Hoang, T.-N.; Kim, D. Detecting in-vehicle intrusion via semi-supervised learning-based convolutional adversarial autoencoders. Veh. Commun. 2022, 38, 100520.
10. Gazzan, M.; Sheldon, F.T. Opportunities for Early Detection and Prediction of Ransomware Attacks against Industrial Control Systems. Future Internet 2023, 15, 144.
11. Gazzan, M.; Alqahtani, A.; Sheldon, F.T. Key Factors Influencing the Rise of Current Ransomware Attacks on Industrial Control Systems. In Proceedings of the 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 27–30 January 2021.
12. Alqahtani, A.; Sheldon, F.T. A survey of crypto ransomware attack detection methodologies: An evolving outlook. Sensors 2022, 22, 1837.
13. Aboaoja, F.A.; Zainal, A.; Ghaleb, F.A.; Al-rimy, B.A.S. Toward an ensemble behavioral-based early evasive malware detection framework. In Proceedings of the 2021 International Conference on Data Science and Its Applications (ICoDSA), Virtual, 10–11 April 2021.
14. Moti, Z.; Hashemi, S.; Karimipour, H.; Dehghantanha, A.; Jahromi, A.N.; Abdi, L.; Alavi, F. Generative adversarial network to detect unseen internet of things malware. Ad. Hoc. Netw. 2021, 122, 102591.

15. Yinka-Banjo, C.; Ugot, O.-A. A review of generative adversarial networks and its application in cybersecurity. Artif. Intell. Rev. 2020, 53, 1721–1736.
16. Zhang, X.; Zhou, Y.; Pei, S.; Zhuge, J.; Chen, J. Adversarial examples detection for XSS attacks based on generative adversarial networks. IEEE Access 2020, 8, 10989–10996.
17. Wang, C.; Xu, C.; Yao, X.; Tao, D. Evolutionary generative adversarial networks. IEEE Trans. Evol. Comput. 2019, 23, 921–934.
18. Li, H.; Zhou, S.; Yuan, W.; Li, J.; Leung, H. Adversarial-example attacks toward android malware detection system. IEEE Syst. J. 2019, 14, 653–656.
19. Lu, Y.; Li, J. Generative adversarial network for improving deep learning based malware classification. In Proceedings of the 2019 Winter Simulation Conference (WSC), National Harbor, MD, USA, 8–11 December 2019.
20. Dumoulin, V.; Belghazi, I.; Poole, B.; Mastropietro, O.; Lamb, A.; Arjovsky, M.; Courville, A. Adversarially learned inference. arXiv 2016, arXiv:1606.00704.
21. Uehara, M.; Sato, I.; Suzuki, M.; Nakayama, K.; Matsuo, Y. Generative adversarial nets from a density ratio estimation perspective. arXiv 2016, arXiv:1610.02920.
22. Haloui, I.; Gupta, J.S.; Feuillard, V. Anomaly detection with Wasserstein GAN. arXiv 2018, arXiv:1812.02463
23. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances in Neural Information Processing Systems 27 (NIPS 2014), Montreal, QC, Canada, 8–13 December 2014.
24. Li, M.; Lin, J.; Meng, C.; Ermon, S.; Han, S.; Zhu, J.Y. Efficient spatially sparse inference for conditional gans and diffusion models. Adv. Neural Inf. Process. Syst. 2022, 35, 28858–28873.
25. Torfi, A.; Fox, E.A.; Reddy, C.K. Differentially private synthetic medical data generation using convolutional GANs. Inf. Sci. 2022, 586, 485–500.
26. Hoang, T.-N.; Kim, D. Detecting in-vehicle intrusion via semi-supervised learning-based convolutional adversarial autoencoders. Veh. Commun. 2022, 38, 100520.