Deep Neural Network based Malicious Network Activity Detection Under Adversarial Machine Learning Attacks

Ferhat Ozgur Catak[1] [0000−0002−2434−9966] and Sule Yildirim Yayilgan[1]

[1] Department of Information Security and Communication Technology, NTNU
Norwegian University of Science and Technology, 2815 Gjøvik, Norway
{ferhat.o.catak,sule.yildirim}@ntnu.no

Abstract. Machine learning-based computational intelligence methods are used more often recently in the cybersecurity area, especially for malicious network activity detection. ML based solutions have been used and discussed by a significant number of authors in literature. Several methods, including deep learning, are used to develop models for solving this issue. So far, attackers try to generate malicious activities in a network to put down several system services or steal some information from the databases. More recent designs of security components use predictive modeling approach to detect such kind of attacks. Thus, the new target for the attackers is machine learning algorithm itself. Previous studies in cybersecurity have almost exclusively focused on attack detection in a network. Another promising line of attack detection research would be machine learning algorithm protection. There are some attacks against deep learning models in the literature, including fast-gradient sign method (FGSM) attack. This attack is the purest form of the gradient-based evading technique that is used by attackers to evade the classification model. This paper presents a new approach to protect a malicious activity detection model from the FGSM attack. Hence, we explore the power of applying adversarial training to build a robust model against FGSM attacks. Accordingly, (1) dataset enhanced with the adversarial examples; (2) deep neural network-based detection model is trained using the KDDCUP99 dataset to learn the FGSM based attack patterns. We applied this training model to the benchmark cyber security dataset.

Keywords: Cyber security · Machine learning · Adversarial attacks · Adversarial machine learning.

1 Introduction

Machine learning (ML) has been part of the cybersecurity area, especially in malicious activity detection since the 2000s, and its applications are increasing day by day [2, 12]. Predictive modeling in cybersecurity is attracting considerable interest due to its flexibility to detect the different patterns of the same attack type. For the future, ML is considered as the de facto solution for security components, especially for the distributed denial of service attacks detection [13, 5].

Intrusion detection systems (IDS) and intrusion prevention systems (IPS) are commonly used for preventing different cyber-attacks types. Early IPS/IDS components used signature-based attack detections. Thus, they are not capable of detecting changes in the attack pattern. When the attacker changes the signature of the attack, such as adding some bits to a network packet's payload, the attacker can evade its attack [11, 1, 8].

Some early studies focus on descriptive statistics to detect malicious network flow. A most known type of network attack is distributed denial of service (DDoS). In a typical DDoS attack, hackers utilize the compromised computers that hacked earlier, to generate significant network traffic to a victim system or computer. Such unusual differences could be discovered using descriptive-analytical techniques. Feinstein et al. [9] use Chi-Square statistics to classify network flow volume irregularities, is correct. The keyword is *volume* for DDoS attacks. The authors introduced time window based entropy fluctuations in a flow volume to detect malicious traffic.

Descriptive statistics based discovery schemes have relied on previously recorded data. A distinct disadvantage of this type of method is that network flux irregularities are a timely fluid target [4]. It is essential to discriminate against the set of malicious traffic precisely. On the other hand, attackers continue to develop a new type of malicious traffic. Consequently, a malicious traffic classification model needs to bypass the overfit problem to any predefined set of malicious traffic types.

Even if such overfitting problems are solved, the attackers always try to find other evading techniques for the security components. One of the most powerful evading techniques against ML-based detection method is adversarial machine learning. The adversarial machine learning has been used to describe the attacks to machine learning models, which tries to mislead models by malicious input instances. Figure 1 shows the typical adversarial machine learning attack.

A typical machine learning model basically consists of two stages as training time and decision time. Thus, the adversarial machine learning attacks occur in either training time or decision time. The techniques used by hackers for adversarial machine learning can be divided into two, according to the time of the attack:

- *Data Poisoning*: The attacker changes some labels of training input instances to mislead the output model.
- *Model Poisoning*: The hacker drives model to produce false labeling using some perturbated instance after the model is created.

The main contributions of this research are to detect network attacks using window-based training input instances according to deep neural networks under adversarial machine learning attacks for model poisoning by hackers . We performed adversarial training based model building and deep neural network

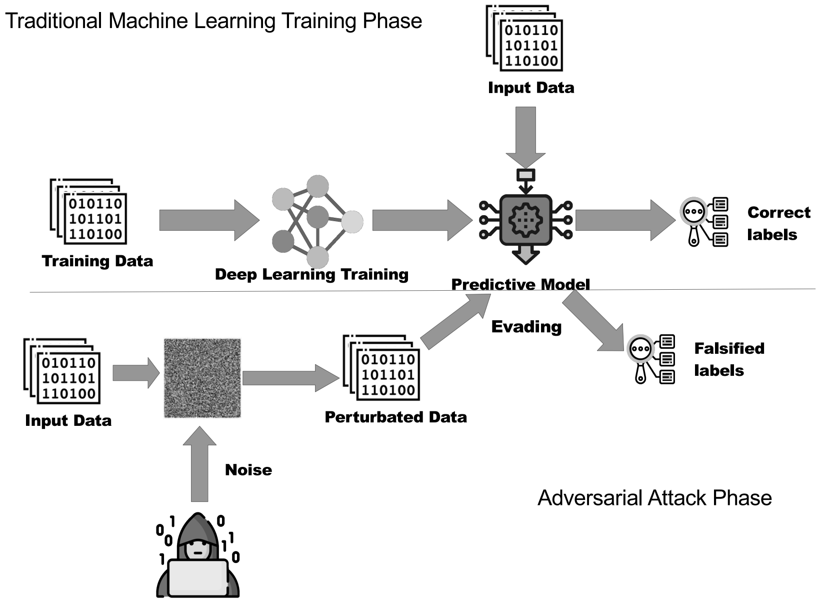


Fig. 1: A typical adversarial machine learning attack.

algorithm based classification to detect normal network behavior and malicious activities, including denial of service (DOS), probe, remote-to-local (r2l), and normal behavior. The primary purpose of the introduced design is to use a mixture model strategy [15, 19] for precise classification of malicious network flow from several network packets. The adversarial training part of the proposed model increase robustness against adversarial instances. The deep neural network model layer tries to find out the exact malicious activity class. Our model is able to respond to the model attacks by hackers who use the adversarial machine learning methods. Figure 2 illustrates the system architecture used to protect the model and to classify correctly.

Our system consists of three main parts, data enhancing, algorithm training, and classification.

The rest of the paper is organized as follows: The related work is presented in Section 2. Section 3 gives brief preliminary information. Our model evaluation and real system test results are presented in Section 4. The concluding remarks are given in Section 5.

2 Related work

In recent years, with the rise of the machine learning attacks, various researches have been submitted to build preventive actions against this kind of attacks.

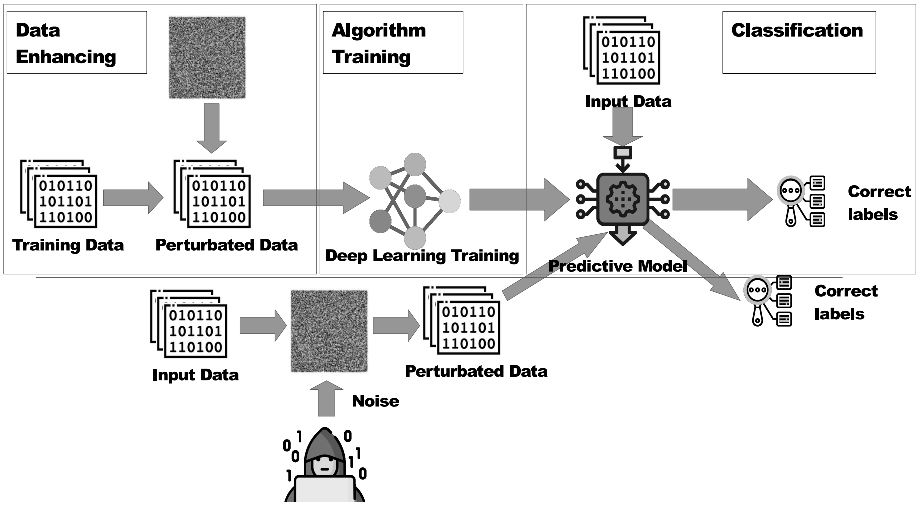


Fig. 2: General system architecture. Architecture consists of 3 parts; data enhancing, algorithm training, and classification.

Data infertility and learning resistance are suggested as countermeasures in fixing a machine learning training phase [20]. Most of the researches in these areas has been adjusted on particular adversarial attacks and usually showed the theoretical analysis of the adversarial machine learning area [14, 10].

Bo Li et al. present a binary domain and classifications. In their work, the proposal begins with mixed-integer linear programming (MILP) with constraint generation and provides instructions on top of these techniques. They further practice the Stackelberg game multi-adversary model algorithm and the other algorithm that feeds back the produced adversarial examples to the training phase, which is called as RAD (Retraining with Adversarial Examples) [16]. Contrarily, their research is individual and operates only in particular systems. It is offered as a comprehensive protection scheme. They have suggested a system that achieves healthy outcomes.

Furthermore, Xiao et al. present a technique to enhance the speed of defense training toward the rectified linear unit (ReLU) [21]. They apply weight sparsity and RELU confidence for reliable confirmation. Their methodology does not present a comprehensive proposal.

Yu et al. suggest a study that can decide the neural network's features under opposed attacks. In their study, the relationship between the input training data and malicious instances is presented. Furthermore, the relationship between the network strength and the decision surface geometry as a sign of the malicious strength of the neural network is presented. By spreading the loss surface to decision surface and other several ways, they provide adversarial robustness by decision surface. The geometry of the decision surface cannot be confirmed mostly, and there is no exact decision border between right or faulty prediction.

Robustness can be improved by creating an immeasurable model, but it can change with attack strength [25].

Mardy et al. study artificial neural networks immune with adversity and improve accuracy scales with various methods, principally with optimization, and demonstrate that there can be extra strong machine learning models [24].

Pinto et al. present a system to explain this problem with the promoted learning process. In their research, they express learning as a zero-sum, minimax objective function. They offer machine learning models that are more immune to changes that are difficult to model through the training and are strongly influenced by changes in training and test circumstances. They induce reinforced learning on machine learning models. They introduce a ”Robust Adversarial Reinforced Learning” (RARL), where they train an agent to act in the behavior of a destabilizing adversary that involves change drives to the system. Nevertheless, in their work, Robust Adversarial Reinforced Learning may overfit itself, and seldom it can miss predicting without any adversarial being in presence [22].

Carlini et al. propose a model that the self-logic and the strength of the machine learning model with a strong attack can be affected. They prove that these types of attacks can often be used to evaluate the effectiveness of potential defenses. They propose defensive distillation as a general-purpose procedure to increase robustness [6].

Harding et al. similarly investigate the effects of malicious input instances generated from targeted and non-targeted attacks in decision making. They provide that non-targeted samples are more effective than targeted samples in human perception and categorization of decisions [3].

Bai et al. present a convolutional autoencoder model with the adversarial decoders to automate the generation of adversarial samples. They produce adversary examples by a convolutional autoencoder model and use pooling computations and sampling tricks to achieve these results. After this process, an adversarial decoder automates the generation of adversarial samples. Adversarial sampling is useful, but it cannot provide adversarial robustness on its own, and sampling tricks are too specific [18].

Sahay et al. propose an FGSM attack and use an autoencoder to denoise the test data. They have also used an autoencoder to denoise the test data, which is trained with both corrupted and healthy data. Then they reduce the dimension of the denoised data. These autoencoders are specifically designed to compress data effectively and reduce dimensions. Hence, it may not be wholly generalized, and training with corrupted data requires many adjustments to get better test results [17].

I-Ting Chen et al. also provide with FGSM attack on denoising autoencoders. They analyze the attacks from the perspective that attacks can be applied stealthily. They use autoencoders to filter data before applied to the model and compare it with the model without an autoencoder filter. They use autoencoders mainly focused on the stealth aspect of these attacks and used them specifically against FGSM with specific parameters [7].

Gondim-Ribeiro et al. propose autoencoders attacks. In their work, they attack 3 types of autoencoders: Simple variational autoencoders, convolutional variational autoencoders, and DRAW (Deep Recurrent AttentiveWriter). They propose to scheme an attack on autoencoders. As they accept that ”No attack can both convincingly reconstruct the target while keeping the distortions on the input imperceptible.”. This method cannot be used to achieve robustness against adversarial attacks [23].

3 Preliminary Information

In this section, we will briefly describe adversarial machine learning, attack environments, and adversarial training that we have used in this study.

3.1 Adversarial Machine Learning

Machine learning model attacks have been utilized mostly by attackers to evade security components that protect a network. Attackers also apply model evasion attacks for phishing attacks, spams, and executing malware code in an analysis environment. There are also some advantages to hackers in misclassification and misdirection of models. Such attacks, the attacker does not change training instances. Instead, he tries to make some small perturbations in input instances in the model's decision time to make this new input instance seem safe (normal behavior). We mainly concentrate on this kind of adversarial attacks in this study. There are many attacking methods for deep learning models, and FGSM is the most straightforward and powerful attack type. We only focus on the FGSM attack, but our solution to prevent this attack can be applied to other adversarial machine learning attacks.

Fast-Gradient Sign Method (FGSM) FGSM works by utilizing the gradients of the neural network to create an adversarial example to evade the model. For an input instance $\mathbf{x}$, the FGSM utilizes the gradients ∇_x of the loss value ℓ for the input instance to build a new instance $\mathbf{x}_{adv}$ that maximizes the loss value of the classifier hypothesis h. This new instance is named the adversarial instance. We can summarize the FGSM using the following explanation:

$$\eta = \epsilon * sign(\nabla_x J(\theta, \mathbf{x}, y)) \tag{1}$$

3.2 Adversarial Training

Adversarial training is a widely recommended defense that implies generating adversarial instances using the gradient of the victim classifier, and then retraining the model with the adversarial instances and their respective labels. This technique has demonstrated to be efficient in defending models from adversarial attacks.

Let us first think a common classification problem with a training instances $X \in \mathbb{R}^{m \times n}$ of dimension d, a label space Y We assume the classifier h_θ has been trained to minimize a loss function ℓas follows:

$$\underset{\theta}{min} \frac{1}{m} \sum_{i=1}^{m} \ell(h_\theta(\mathbf{x}_i, y_i)) \tag{2}$$

Given a classifier model $h_\theta(\cdot)$ and an input instance x, whose responding output is y, an adversarial instance x^* is an input such that:

$$h_\theta(x^*) \neq y \quad \wedge \ d(x, x^*) < \epsilon \tag{3}$$

where $d(\cdot, \cdot)$ is the distance metric between two input instances original input x and adversarial version x^*. Most actual adversarial model attacks transform Equation 3 into the following optimization problem:

$$\underset{x}{\mathbf{argmax}}\, \ell\left(h_\theta(x^*), y\right) \tag{4}$$

$$s.t. d(x, x^*) < \epsilon \tag{5}$$

where ℓ is loss function between predicted output $h(\cdot)$ and correct label y.

In order to mitigate such attacks, at per training step, the conventional training procedure from Equation 2 is replaced with a `min-max` objective function to minimize the expected value of the maximum loss, as follows:

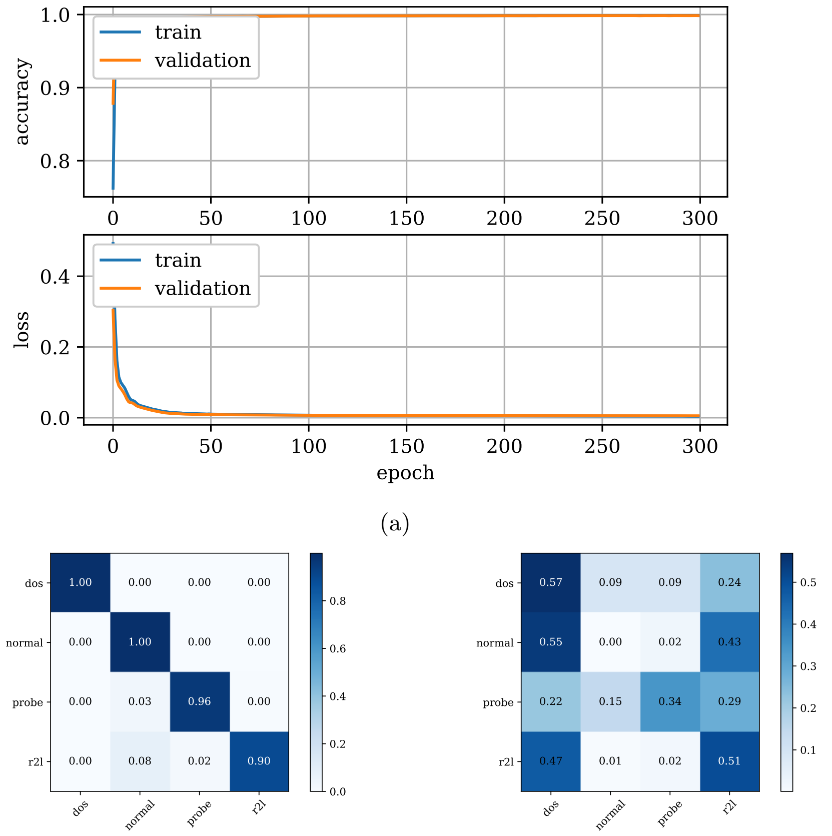
$$\underset{\theta}{min} \underset{(x,y)}{\mathbb{E}} \left(\underset{d(x,x^*)<\epsilon}{max} \ell(h(x^*), y) \right) \tag{6}$$

4 Experiments

In this section, we conduct experiments on the KDDCUP99 dataset from the publicly available data set repositories. We implemented the proposed mitigation method using Keras and TensorFlow libraries in the Python environment.

In Figure 3a, the training history of the model, which uses normal input instances, is shown. As it can be seen in history, the graph of loss and accuracy progresses smoothly. Figure 3b shows the confusion matrix of the test data set using the trained model. As can be seen from the Figure, the classification performance of the model for normal instances is quite good. Figure 3c shows the confusion matrix of adversarial samples. As can be seen from the graph, the classification performance of the model decreases considerably.

In order to show the effect of adversarial samples on the model in more detail, we have shown the results of the classification reports in Table 1-2. According to these tables, the weighted average $F1$ value of the benign test dataset is 0.997379. The weighted $F1$ value of the adversarial dataset, which was created from the same model and created from the test dataset, dramatically decreased up to 0.176636. As one can see here, a classification model created by applying only the training data is highly vulnerable to adversarial attacks.



(a)

(b)

(c)

Fig. 3: Original classifier model for KDDCUP99 dataset. (a) accuracy and loss plot with epoch, (b) test dataset confusion matrix, (c) adversarial instances confusion matrix.

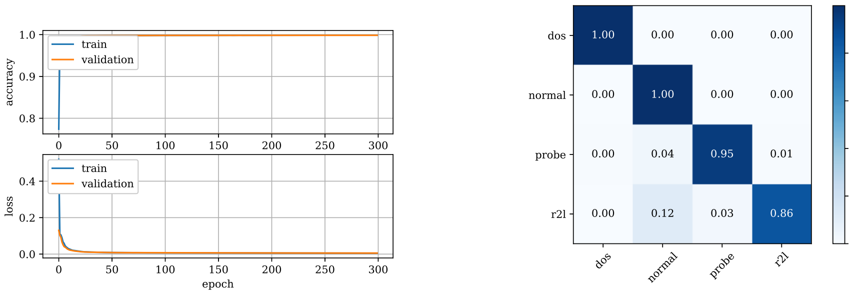
Table 1: Original model with normal instances' classification report

Classes	**precision**	**recall**	**f1-score**	**support**
dos	0.998901	0.999450	0.999175	10911
normal	0.997949	0.998119	0.998034	17546
probe	0.971239	0.962719	0.966960	456
r2l	0.920635	0.896907	0.908616	194
accuracy	0.997389	0.997389	0.997389	0.997389
macro avg	0.972181	0.964299	0.968196	29107
weighted avg	0.997372	0.997389	0.997379	29107

Table 2: Original model with adversarial instances' classification report

Classes	**precision**	**recall**	**f1-score**	support
dos	0.388392	0.569150	0.461710	10911
normal	0.014298	0.000912	0.001714	17546
probe	0.098680	0.344298	0.153395	456
r2l	0.009416	0.505155	0.018487	194
accuracy	0.222661	0.222661	0.222661	0.222661
macro avg	0.127697	0.354879	0.158827	29107
weighted avg	0.155820	0.222661	0.176636	29107

Figure 4a shows the history of the classification model trained using the adversarial training method for the train and test set. In Figure 4b, the confusion matrix of the adversarial training model is shown. As can be seen from the Figure, the classification performance of the model for adversarial instances is quite good.



(a)

(b)

Fig. 4: Adversarial trained classifier model for KDDCUP99 dataset. (a) accuracy and loss plot with epoch, (b) adversarial instances confusion matrix.

In order to show the effect of adversarial training on the model in more detail, we have shown the results of the classification report in Table 3. According to these tables, the weighted average $F1$ value of the benign test dataset is 0.996858.

According to the table, the original vulnerable model's $F1$ metric decreases up to 0.176636 with adversarial input instances, while the original $F1$ value was 0.996858. With our protection methods, the new classification model's $F1$ metric is 0.996858, almost the same as the original $F1$ metric. As can be seen, the malicious network traffic classification model shows high classification performance with only a small loss.

Table 3: Adversarial trained model with adversarial instances' classification report

Classes	**precision**	**recall**	**f1-score**	**support**
dos	0.998901	0.999542	0.999221	10911
normal	0.997493	0.997948	0.997721	17546
probe	0.975281	0.951754	0.963374	456
r2l	0.873684	0.855670	0.864583	194
accuracy	0.996874	0.996874	0.996874	0.996874
macro avg	0.961340	0.951229	0.956225	29107
weighted avg	0.996848	0.996874	0.996858	29107

5 Conclusion

In this study, we explained the methods of developing model robustness to adversarial instances during the detection of malicious network attacks. The malicious network traffic detection methods is a mature research subject in the literature, but how the model itself behaves under adversarial attack is not much researched. Attackers want to continue their malicious activities by evading network security components by applying adversarial machine learning techniques. With the increasing use of machine learning models in cybersecurity soon, there will be an increase in such attacks. In this study, we would have recommended a method to detect malicious network traffic by keeping the classification performance almost identical even under the attack of the model itself that detects network attacks. Attackers can reduce the $F1$ value of a model used without this precaution from 0.997379 to 0.176636. With our method, the $F1$ value decreases only to 0.996858, detecting malicious network traffic at very high rates.

In this study, we examined the FGSM attack. In future studies, we plan to improve our robustness method by analyzing other attack methods such as the basic iterative method and DeepFool.

References

1. Abeshu, A., Chilamkurti, N.: Deep learning: The frontier for distributed attack detection in fog-to-things computing. IEEE Communications Magazine **56**(2), 169–175 (2018)
2. Ben-Asher, N., Gonzalez, C.: Effects of cyber security knowledge on attack detection. Computers in Human Behavior **48**, 51 – 61 (2015). https://doi.org/https://doi.org/10.1016/j.chb.2015.01.039, http://www.sciencedirect.com/science/article/pii/S0747563215000539
3. Bertenthal, S.H.P.R.B.I., Gonzalez, C.: Human decisions on targeted and non-targeted adversarial sample (Aug 2018), https://mindmodeling.org/cogsci2018/papers/0103/index.html
4. Bhuyan, M.H., Bhattacharyya, D., Kalita, J.: An empirical evaluation of information metrics for low-rate and high-rate ddos attack detection. Pattern Recognition Letters **51**, 1 – 7

(2015). https://doi.org/https://doi.org/10.1016/j.patrec.2014.07.019, http://www.sciencedirect.com/science/article/pii/S016786551400244X

5. Cappers, B.C.M., van Wijk, J.J.: Snaps: Semantic network traffic analysis through projection and selection. In: 2015 IEEE Symposium on Visualization for Cyber Security (VizSec). pp. 1–8 (Oct 2015). https://doi.org/10.1109/VIZSEC.2015.7312768
6. Carlini, N., Wagner, D.A.: Towards evaluating the robustness of neural networks. CoRR **abs/1608.04644** (Aug 2016), http://arxiv.org/abs/1608.04644
7. Chen, I., Sirkeci-Mergen, B.: A comparative study of autoencoders against adversarial attacks. nt'l Conf. IP, Comp. Vision, and Pattern Recognition (Aug 2018), https://csce.ucmss.com/cr/books/2018/LFS/CSREA2018/IPC3651.pdf
8. Diro, A.A., Chilamkurti, N.: Distributed attack detection scheme using deep learning approach for internet of things. Future Generation Computer Systems **82**, 761 – 768 (2018). https://doi.org/https://doi.org/10.1016/j.future.2017.08.043, http://www.sciencedirect.com/science/article/pii/S0167739X17308488
9. Feinstein, L., Schnackenberg, D., Balupari, R., Kindred, D.: Statistical approaches to ddos attack detection and response. In: Proceedings DARPA Information Survivability Conference and Exposition. vol. 1, pp. 303–314 vol.1 (April 2003). https://doi.org/10.1109/DISCEX.2003.1194894
10. Gettings, M.I.V.G.K.M., Kinsy, M.A.: Survey of attacks and defenses on edge-deployed neural networks (Nov 2019), https://arxiv.org/abs/1911.11932
11. Han, F., Xu, L., Yu, X., Tari, Z., Feng, Y., Hu, J.: Sliding-mode observers for real-time ddos detection. In: 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA). pp. 825–830 (June 2016). https://doi.org/10.1109/ICIEA.2016.7603695
12. Jasiul, B., Szpyrka, M., Śliwa, J.: Detection and modeling of cyber attacks with petri nets. Entropy **16**(12), 6602–6623 (2014). https://doi.org/10.3390/e16126602, http://www.mdpi.com/1099-4300/16/12/6602
13. Jiang, D., Xu, Z., Zhang, P., Zhu, T.: A transform domain-based anomaly detection approach to network-wide traffic. Journal of Network and Computer Applications **40**, 292 – 306 (2014). https://doi.org/https://doi.org/10.1016/j.jnca.2013.09.014, http://www.sciencedirect.com/science/article/pii/S1084804513002038
14. Jiang, J.G.Y.Z.X.H.Y., Sun, J.: Rnn-test: Adversarial testing framework for recurrent neural network systems (Nov 2019), https://arxiv.org/abs/1911.06155
15. Latif, S., Rana, R., Qadir, J.: Adversarial machine learning and speech emotion recognition: Utilizing generative adversarial networks for robustness. CoRR **abs/1811.11402** (2018), http://arxiv.org/abs/1811.11402
16. Li, B., Vorobeychik, Y.: Evasion-robust classification on binary domains. ACM Trans. Knowl. Discov. Data **12**(4), 50:1–50:32 (Jun 2018). https://doi.org/10.1145/3186282, http://doi.acm.org/10.1145/3186282
17. Mahfuz, R.S.R., Gamal, A.E.: Combatting adversarial attacks through denoising and dimensionality reduction: A cascaded autoencoder approach. CoRR **abs/1812.03087** (Dec 2018), http://arxiv.org/abs/1812.03087
18. Quan, W.B.C., Luo, Z.: Alleviating adversarial attacks via convolutional autoencoder pp. 53–58 (Jun 2017). https://doi.org/10.1109/SNPD.2017.8022700, https://doi.org/10.1109/SNPD.2017.8022700
19. Ren, K., Zheng, T., Qin, Z., Liu, X.: Adversarial attacks and defenses in deep learning. Engineering **6**(3), 346 – 360 (2020). https://doi.org/https://doi.org/10.1016/j.eng.2019.12.012, http://www.sciencedirect.com/science/article/pii/S209580991930503X

20. Rubinstein, L.H.A.D.J.B.N.B., Tygar, J.D.: Adversarial machine learning. In: Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence. pp. 43–58. AISec '11, ACM, New York, NY, USA (Oct 2011). https://doi.org/10.1145/2046684.2046692, http://doi.acm.org/10.1145/2046684.2046692
21. Shafiullah, K.Y.X.V.T.N.M., Madry, A.: Training for faster adversarial robustness verification via inducing relu stability. CoRR **abs/1809.03008** (Sep 2018), http://arxiv.org/abs/1809.03008
22. Sukthankar, L.P.J.D.R., Gupta, A.: Robust adversarial reinforcement learning. CoRR **abs/1703.02702** (Mar 2017), http://arxiv.org/abs/1703.02702
23. Tabacof, G.G.P., Valle, E.: Adversarial attacks on variational autoencoders. CoRR **abs/1806.04646** (Jun 2018), http://arxiv.org/abs/1806.04646
24. Tsipras, A.M.A.M.L.S.D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. CoRR **abs/1706.06083** (Jun 2017), http://arxiv.org/abs/1706.06083
25. Zhao, F.Y.C.L.Y.W.L., Chen, X.: Interpreting adversarial robustness: A view from decision surface in input space. CoRR **abs/1810.00144** (Sep 2018), http://arxiv.org/abs/1810.00144