Adversarial Attack Against Network Intrusion Detection Systems with Deep Learning

MBOW MARIAMA*[1, a)], HIROSHI KOIDE[2, b)], KOUICHI SAKURAI[1, c)]

Abstract: Many machine learning (ML) and deep learning (DL) techniques have been adopted as a new approach for Network Intrusion Detection System (NIDS) due to their ability to learn underlying threat patterns/features. Recent researches revealed ML/DL-based models are susceptible to adversarial attacks. Several adversarial techniques have emerged lately from the deep learning research, largely in the area of computer vision where minor modifications are performed on images that cause a classifier to produce incorrect predictions. However, in other fields, such as intrusion detection, the exploration of such methods is still growing. The intention of this research is to study the nature of the adversarial problem in NIDS. We focus on the attack perspective, which includes techniques to generate adversarial examples capable of evading a variety of deep learning models. More specifically, we explore the robustness of RNN-IDS and CNN-IDS against adversarial attack.

Keyword: AI security, cyber security, Intrusion Detection System, Adversarial Attack

1. Introduction

An intrusion detection system (IDS) which is an important cyber security technique, plays an essential role in defending computer networks against attacks [1].The detection method used in intrusion detection system (IDS) are generally classified to signature-based and anomaly-based detection system. Due to the inability to detect novel attacks, signature-based network intrusion detection system (NIDS) that was traditionally used to detect malicious traffic is starting to be replaced by anomaly-based NIDS which creates a model of normal behaviors of the system and detects deviation from this model. Among the various approaches for anomaly-based, artificial intelligence field have gained increasing attention due to the advantage of machine learning (ML) algorithms in detecting zero-day-attacks [2].

For a long period, the sole focus of IDS researchers using ML techniques was improving the performance of NIDS (true positive rate, accuracy etc.). Still in this same objective deep learning (DL) techniques have been recently widely used to reduce false rate and improve accuracy detection. Nowadays, the security part of these models cannot be ignored; many of them have been shown to be vulnerable to adversarial attack. A common challenge of these algorithm is generalization or robustness [3].

We define adversarial attacks when attacker intentionally inputs adversarial examples to machine learning /deep learning models in order to fool or cause the model to make a mistake. several adversarial examples have emerged lately from the deep learning research, largely in the area of computer vision. Recently researchers begin to realize that adversarial examples may widely exist in various application scenarios including security applications. These adversarial attacks deserve an important attention in the domain of IDS, since with the growth of machine learning applied to this area, adversaries can attempt to circumvent detection systems. Adversarial attacks can be mainly classified as poisoning and evasion attacks. In this work we will explore the evasion attacks.

Different works have applied adversarial machine learning to intrusion scenarios. Although the methods proposed have shown effectiveness in compromising a ML-based IDS model , most of them are done in shallow ML model. However, it is observed in these last years that DL-based NIDS methodologies are preferred over the ML methodologies due to their efficiency in learning from large datasets in raw form [4]. Our purpose is to assess whether an adversarial attack against IDS can pose a larger threat to network security if DL-based IDS are used.

Our objective is to study the adversarial attacks against deep learning model applied for NIDS: adversary examples will be generated and evaluated in grey-box model which mean we will assume the attacker has no knowledge of the target classifier. Specifically, we evaluate the robustness of RNN and CNN. We will perform an evasion attack using an adversarial machine learning technique known as the Jacobian-based Saliency Map Attack (JSMA) [5] perturbation method to generate adversarial examples and apply these algorithms to the NSL-KDD data sets. In this paper we present the first step of our work which is creating the RNN based IDS model.

The paper is organized as follow: Section 2 discuss the related work. Section 3 describes the methodology Section 4 discuss the experimental results and analysis. Section 5 summarizes the paper and discuss the future work. We conclude our work in Section 6 with a review of the Background needed.

2. Related work

In this section we discuss different works that have applied adversarial machine learning to intrusion detection scenarios using evasion attack. Model evasion attack is often done via

1 Department of Informatics Graduate School of Information Science and Electrical Engineering, Kyushu University
2 Cyber Security Center, Kyushu University
a) mbow.mariama.076@s.kyushu-u.ac.jp
b) koide@cc.kyushu-u.ac.jp
c) sakurai@inf.kyushu-u.ac.jp
* The author is financially supported by the Ministry of Education, Culture, Sports, Science and Technology (MEXT).

ⓒ2021 Information Processing Society of Japan

gradient descent over the discrimination function of the classifier [6].

Maria Rigaki et al. tested the effectiveness of adversarial attacks in an intrusion detection scenario assuming no knowledge of the target classifier [7]. They performed the tests on the NSL-KDD dataset. FGSM and JSMA were used to generate adversarial sample, and 5 models used to perform classification: decision tree, random forest, linear SVM, voting ensembles of the previous three classifiers and a multi-layer perceptron neural network (MLP). The results on JSMA showed that all classifiers accuracy was affected, with linear SVM being the most affected and the most resilient classifier was random forest. The authors made an important remark on the percentage of features modified by the attacks: FGSM modifies 100% of the features on every sample, while JSMA only modified on average 6% of the features. This makes JSMA the more realistic attack.

To test the vulnerability of ML based IDS, Elie Alhajjar et al. [3] investigated the effects of creating perturbations using techniques from evolutionary computation: Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) and deep learning (GAN) in a white box model. Their methods achieved high misclassification rates against 11 shallow machine learning classifiers: Support Vector Machine (SVM), Decision Tree (DT), Naive Bayes (NB), K-Nearest Neighbors (KNN), Random Forest (RF), Multi-layer Perceptron (MLP), Gradient Boosting (GB), Logistic Regression (LR), Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), and Bagging (BAG). With DT and SVM the most vulnerable.

JSMA has also been used against Multilayer Perceptron (MLP) based IDS [6] in a white box scenario. Their results show that neural network-based IDS is susceptible to model evasion attack, and attackers can use this technique to evade intrusion detection systems effectively.

In [8] the authors test the adversarial examples in deep neural network (DNN) in the black-box model. Their methods demonstrate that adversary can generate effective adversarial examples against DNN classifier trained for NIDS even when the internal information of the target model is isolated from the adversary. A survey of relevant work can be found in [9] and [10].

3. Proposed Work

A. Datasets:

The NSL-KDD dataset [11]generated in 2009 is widely used in intrusion detection experiments.In our experiment we use the NSL-KDD as our dataset. Although this data set is outdated, it is still being used as a benchmark for building IDS and adversarial IDS as well. Moreover, the purpose of our study is the robustness of classifiers and not making claims about prediction capabilities and generalization [7]. NSL-KDD dataset is the duplicate removed and reduced size version of KDD99 datasets. The dataset covers the $KDDTrain^{+}$ dataset as the training set and $KDDTest^{+}$ as test set. It contains 41 features and one class label. Attacks in the dataset are categorized into four attack types: DoS (Denial of Service attacks), R2L (Root to Local attacks), U2R (User to Root attack), and Probe (Probing attacks).

B. Data pre-processing

In this part, we do pre- processing using common techniques such as one-hot encoding and Min-Max Scaling, which results in a final data set with 122 features. A detailed pre-processing was given in [7]:

- All categorical variables were transformed to numerical using One-hot encoding.
- Normalization of all features using Min-Max Scaler was performed.Every feature is mapped to the [0,1] range in order to avoid having features with very large values

$$xi = \frac{xi - Min}{Max - Min}$$

- The problem was transformed to a 5-class classification one by changing the attack label from 39 distinct attack categories to four (" DoS"," U2R"," R2L"," Probe") and" normal".

	Normal	**Dos**	**Probe**	**R2L**	**u2r**	**Total**
KDD Train^{+}	67343	45927	11656	995	52	125973
KDDt Test^{+}	9711	7460	2885	2421	67	22544

Table1: 5-class classification in NSL-KDD

C. Building the ML Model for IDS

We now describe the construction of the target machine learning model for network-based intrusion detection system (IDS) and its performance in detecting attack traffic. In fig 1 our proposed baseline model.

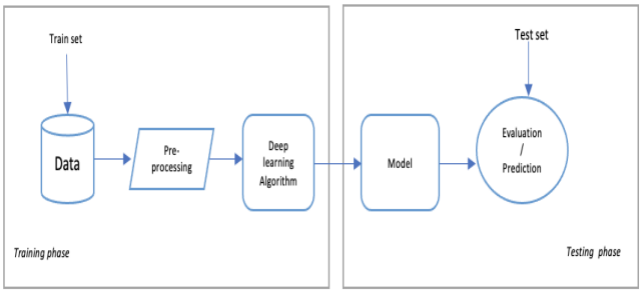

Fig 1: Proposed baseline architecture

RNN-IDS- Many RNN models have been recently used in IDS and their proposed model show better performance when compared to other classifier [12] [13] [14]. RNN-based IDS was proposed by Yin et al [12],they studied the performance of the model in binary classification and multiclass classification in NSL-KDD dataset. A comparison of the performance of RNN-IDS with other machine learning methods was performed. The proposed model performed well when compared to ML algorithms. Our baseline model will be based on these approaches.

There are three layers in our models. First, simple RNN layer which has output shape (None,122) and 15128 parameters (weights) in this layer. Second, dense layer has output shape (None,80) and 9840 parameters. Lastly, activation layer has output shape (None,5). We use a Rectified Linear Unit (ReLU) activation function, , in the input layer as well as hidden layer of the RNN .As activation function, we use SoftMax.To compile the model, we use an Adam optimization algorithm, and categorical_crossentropy loss function. We train the model with 100 epochs with a batch size of 512. Figure 2 describe the model design.

ⓒ2021 Information Processing Society of Japan

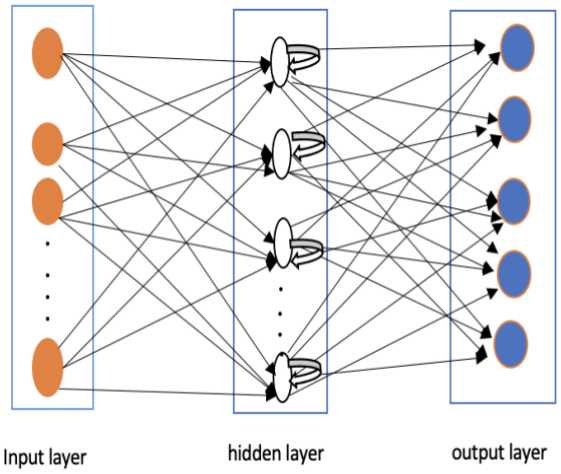

Fig. 2. Proposed model (A simple RNN network with one hidden layer)

4. Experiment & Results

A. Metrics

The most important performance indicator (Accuracy, AC) of intrusion detection [12] is used to measure the performance of our model. The Accuracy is the percentage of the number of records classified correctly versus total the records. In addition, we use the Confusion Matrix, a largely used metric for supervised learning. The confusion matrix, is a specific table layout that allows visualization of the performance of an algorithm. Table 2 describe the confusion matrix.

•Accuracy= $\frac{True\ Positive+True\ Negative}{True\ Positive+True\ Negative+False\ Positive+False\ Negative}$

	Predicted: YES	Predicted: NO
Actual: YES	True Positive	False Negative
Actual: NO	False Positive	True Negative

Table2: confusion matrix

B. Environment

We have used one of the most popular deep learning frameworks – Tensorflow 2 [15] , and implement deep learning by using Keras on the top of Tensorflow to build deep learning model. We have installed Tensorflow in Anaconda [16]. The experiment is performed on a personal macbook pro, which has a configuration of 2 GHz Quad-Core Intel Core i5, 16 GB memory and does not use GPU acceleration.

C. Experiment Result

We evaluate our Recurrent Neural Network(RNN)trained over the NSL-KDD datasets. The experiments show that our model RNN works with an accuracy of 99% in the KDDTrain and 74% in KDDTest when the learning rate is 0.01. Based on the result in the confusion matrix, table3, our model shows lower detection rates for minority attack classes like U2R and R2L. However, the lower detection rate of RNN for minority attacks has also been mentioned in [4]. Another work would be to investigate this problem when using RNN for IDS. Hence, our future work with this model will be to improve the accuracy detection rate by adding more hidden layers or working with the variant of RNN such as LSTM, GRU etc.

Predicted / Actual	**Normal**	**Dos**	**Probe**	**R2L**	**U2R**
Normal	**9445**	62	197	7	0
Dos	1718	**5633**	65	44	0
Probe	928	215	**1235**	43	0
R2L	2436	38	31	**380**	0
U2R	60	1	1	5	**0**

Table3: Confusion Matrix of our model

5. Future Work

In this paper we have presented the first step of our work which is creating the RNN to detect and classify benign and attack traffic using the network-based intrusion detection system (IDS) dataset: NSLKDD. The model is created with 3 layers: a simple RNN with 122 units, 1 hidden layer with 80 units and output layer with 5 units. Experimental results have shown an accuracy of 99% in the train set and 74% in the test set . However, this accuracy result isn't better than [14] [12].It should also be noted that no attempt has been made to tune the RNN classifier for the sake of optimizing its performance under any metric. Also, the purpose of our work is to test the performance of those model under adversarial attack. Due to the limited time, we did not make more comparisons to tune the RNN model for improving the performance and continue the work. The future work is to improve our RNN based IDS model, implement the CNN-IDS and perform the evasion attack on these RNN and CNN models. To implement our attack, we will select the Jacobian-based Saliency Map Attack (JSMA) method in a grey-box setting, where we will assume the adversary has no knowledge over the model used to perform prediction.

6. Background

A. Deep learning

Deep Learning(DL) is the subset of Machine Learning(ML) which includes many hidden layers to get the characteristics of the deep network. Machine learning algorithms have been classified into two type: traditional machine learning models also called shallow models and Deep learning model. The deep learning techniques are said to be more efficient than the shallow machine learning due to their ability to learn the important features from large datasets and their computationally efficient training algorithm.

Many deep learning algorithms have been proposed and successfully applied to several domains. The typical models include Deep Neural Network(DNN)Deep Belief Network (DBN), Autoencoder (AE), Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) .While AE and DBN are unsupervised learning models, CNN and RNN are supervised learning models.A taxonomy of machine learning algorithms has been proposed in [17], it summarized the common

ⓒ2021 Information Processing Society of Japan

machine learning algorithms used in IDSs.The figure is reproduced in fig3. In this paper, we consider using supervised learning models, which are CNN, RNN. An introduction of each model is described below.

The essential part of the neural network is a neuron with an activation function (σ), a set of weights (W) and a set of biases (b) [18]. Regarding these parameters, transformation is defined by:

$$h = \sigma\,(w^T x + b)$$

Where x is the inputs of neurons, w is the weighs, b is bias, T is matrix transpose and σ is activation function. One of the most used Neural Network topology is the Multilayer Perceptron (MLP) Network [6]. A MLP network is usually constructed with three or more layers, that is, one input layer, one or more hidden layer, and one output layer. A neural network with multiple hidden layers is usually called a deep neural network [18].

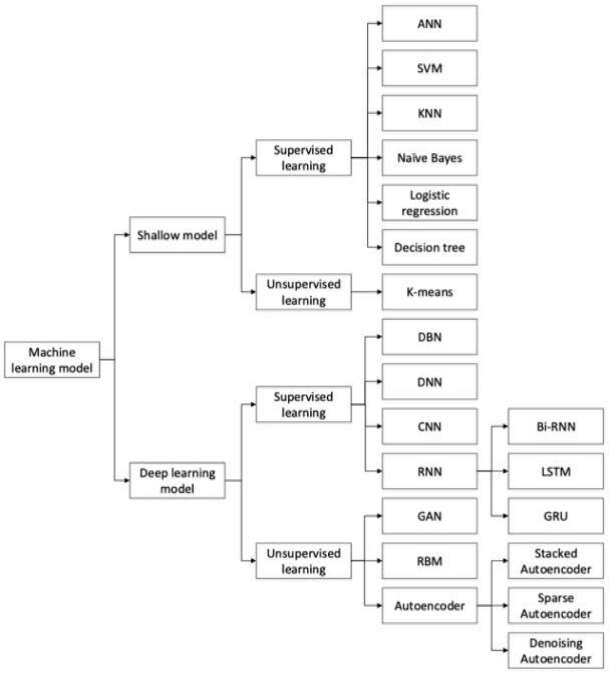

Fig 3: taxonomy of machine learning algorithms (in [17])

- **Convolutional Neural Network (CNN)**

CNN is a special family of neural networks that contain convolutional layers, where its goal is to learn suitable feature representations of the input data. It is an advanced model used in many fields such as image classification, text recognition, object tracking, speech recognition, posture estimation, natural language processing, visual saliency detection, and human action recognition [18]. A convolutional neural network consists of an input layer, the stack of convolutional and pooling layers for feature extraction, and finally a fully connected layer and a softmax classifier in the classification layer. Its difference with MLP is the weight sharing and pooling. CNN is widely successful in the computer vision field. For the IDS, they are used for the supervised feature extraction and classification purposes.CNN-IDS has been proposed in [14] [18].In fig 4 an example of CNN-IDS proposed in [18]

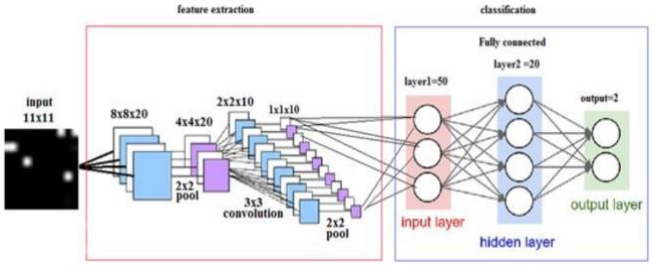

Fig 4: CNN-IDS. Proposed structure include convolution, pooling and fully connected layers for IDS (in [18]).

- **Recurrent Neural Network (RNN)**

Recurrent Neural Networks (RNN) extends the capabilities of the traditional feed-forward neural network and is designed to model the sequence data of neural networks. It has the property of reusing information already given. RNN is made of input, hidden, and output units, where the hidden units are considered to be the memory elements. It contains a looped connection in the hidden layer, which implies that we use the previous hidden state along with the input to predict the output [19].In fig 5 an example of RNN. To make a decision, each RNN unit relies on its current input and the output of the previous input. Due to growing computational resources, RNNs have played an important role in the fields of computer vision, natural language processing (NLP), semantic understanding, speech recognition, language modelling, translation, picture description, and human action recognition.

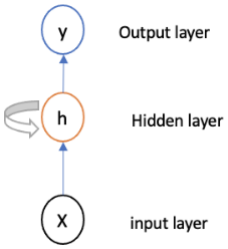

Fig 5: A Recurrent Neural Network (RNN)

Many IDS based RNN or CNN have been proposed where they performed well compared to machine learning method. A literature can be found in [20].

B. Adversarial Machine Learning

Adversarial machine learning is the field that studies the robustness of machine learning classifiers by a class of attacks which aim to fool or deteriorate the performance of classifiers. Adversarial attack viewed from the attacker strategy can be mainly classified as poisoning attack, when the attack occurs during the training phase: the attacker aims to influence the training data to cause the model to under-perform, or evasion attack, if the attack occurs during the testing phase: the attacker manipulates the data to cause the model to make incorrect predictions.

A Qualitative taxonomy was presented by Huang and al. [21]. It categorized the attacks based on three properties: Influence (causative or exploratory), security violation (integrity, availability, privacy), specificity (targeted, indiscriminate)

ⓒ2021 Information Processing Society of Japan

A major component of an adversarial attack is the attack strategies. Creating these adversarial sample consist of solving an optimization problem to determine the minimum perturbation which maximizes the loss for the neural network. Several techniques have been proposed with a trade-off on performance, complexity, computational efficiency [7].These models include the Fast Gradient Sign Method (FGSM) [22], the Jacobian-based saliency map attack (JSMA) [5], Evolutionary algorithms [23], Deepfool [24], Generative adversarial networks(GAN) [25] etc. Based on the adversary knowledge, we outline three application scenarios [10]:

- White Box: The attacker has complete knowledge of the target classification model ,the training data, model parameters and other useful information
- Gray Box: The adversary has an incomplete knowledge of the target model and knows the features considered by the model and its type.
- Black Box: The adversary is totally unaware target model.

C. JSMA

The Jacobian based Saliency Map Attack (JSMA) is and adversarial sample Generation proposed by Papernot et al. JSMA uses feature selection, with the aim of minimizing the number of features modified (L0 distance metric) while causing misclassification [9].The JSMA generates adversarial samples based on the Saliency Map method [5], which gives an indication of which features will have more effect on the misclassification if they are perturbed. Its technique allows an adversary who has knowledge of the network architecture to leverage the adversarial saliency map to identify features of the input that most significantly impact output classification [5]. JSMA has been identified to be a realistic attack due to its ability to modify small range of features while performing a successfully attack [7] [9].

References

[1] A. Ghorbani, L. Wei and T. Mahbod, Network Intrusion Detection and Prevention, New York Dordrecht Heidelberg London: Springer, 2009, pp. viii-ix.

[2] S. Dule, O. L. Nandi, A. K. Charles and C. Tucker, "Generative Adversarial Attacks Against Intrusion Detection Systems Using Active Learning," *In ACM Workshop on Wireless Security and Machine Learning (WiseML '20),* no. https://doi.org/10.1145/3395352.3402618, p. 6, July 13, 2020.

[3] E. Alhajjar, P. Maxwell and N. D. Bastian, "Adversarial Machine Learning in Network Intrusion Detection Systems," *arXiv:2004.11898v1 [cs.CR],* 2020.

[4] Z. Ahmad, A. S. Khan, C. W. Shiang, J. Abdullah and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Transactions on Emerging Telecommunications Technologies published by John Wiley & Sons Ltd.,* 2020.

[5] N. Papernot, S. Jha, M. Fredrikson, B. C. Z and A. Swami, "The Limitations of Deep Learning in Adversarial Settings," *the 1st IEEE European Symposium on Security & Privacy, IEEE 2016,* 2016.

[6] A. Ayub, W. Johnson, A. T. Douglas and S. Ambareen, "Model Evasion Attack on Intrusion Detection Systems using Adversarial Machine Learning," *54th Annual Conference on Information Sciences and Systems (CISS),* 2020.

[7] R. Maria and E. Ahmed, "Adversarial Deep Learning Against Intrusion Detection Classifiers," *ST-152 Workshop on Intelligent Autonomous Agents for Cyber Defence and Resilience,* 2017.

[8] Y. Kaichen, L. Jianqing, Z. Chi and F. Yuguang, "Adversarial Examples Against the Deep Learning Based Network Intrusion Detection Systems," *MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM),* 2018.

[9] N. MARTINS, C. MAGALHÃES, T. CRUZ and A. P. HENRIQUES, "Adversarial Machine Learning Applied to Intrusion and Malware Scenarios: A Systematic Review," *IEE Access,* vol. 8, 2020.

[10] I. Olakunle, A.-K. Rana, M. Ashraf and O. S. M., "The Threat of Adversarial Attacks Against Machine Learning in Network Security: A Survey," *arXiv:1911.02621v2 [cs.CR],* 2020.

[11] "Canadian Institute for Cybersecurity," NSL-KDD dataset, 2009, [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html.

[12] Y. CHUANLONG, Z. YUEFEI, F. JINLONG and H. XINZHENG, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," *IEEE,* vol. 5, 2017.

[13] M. Sheikhan and Z. Jadidi, "Intrusion detection using reduced-size RNN based on feature grouping," *Neural Computing and Applications,* 2012.

[14] N. Chockwanich and V. Visoottiviseth, "Intrusion Detection by Deep Learning with TensorFlow," *21st International Conference on Advanced Communication Technology (ICACT),* 2019.

[15] "Tensorflow," Google, [Online]. Available: https://www.tensorflow.org/about.

[16] "Anaconda," [Online]. Available: https://www.anaconda.com/.

[17] L. Hongyu and L. Bo, "Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey," *Applied Sciences MDPI,* 2019.

[18] L. Mohammadpour, T. Ling, C. Liew and C. Chun, "A Convolutional Neural Network for Network Intrusion Detection System," *Proceedings of the APAN – Research Workshop 2018 ISBN 978-4-9905448-8-1,* 2018.

[19] S. Ravichandiran, Hands-On Deep Learning Algorithms with Python, Birmingham, UK: Packt Publishing Ltd., July 2019.

[20] A. Zeeshan, S. K. Adnan, W. S. Cheah, A. Johari and A. Farhan, " Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Transactions on Emerging Telecommunications Technologies ,* 2020.

[21] H. Ling, D. J. Anthony, N. Blaine, I. P. R. Benjamin and T. J. D, "Adversarial Machine Learning," *in: Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence,* pp. 43-58, 2011.

[22] J. G. Ian, S. Jonathon and S. Christian, "EXPLAINING AND HARNESSING ADVERSARIAL EXAMPLES," *International Conference on Learning Representations (ICLR);arXiv:1412.6572v3 [stat.ML],* 2015.

[23] A. Nguyen, J. Yosinski and J. Clune, "Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images," *In Computer Vision and Pattern Recognition (CVPR '15), IEEE, 2015;arXiv:1412.1897v4 [cs.CV],* 2015.

[24] M.-D. Seyed-Mohsen, A. Fawzi and P. Frossard, "“Deep-Fool: A Simple and Accurate Method to Fool Deep Neural Networks”;doi: 10.1109/cvpr.2016.282," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016).,* 2016.

[25] I. Goodfellow, "NIPS 2016 Tutorial: Generative Adversarial Networks," *https://arxiv.org/abs/1701.00160,* 2017.

Acknowledgments

This research is supported by the Ministry of Education, Culture, Sports, Science and Technology (MEXT).
Part of this research is also supported by Hitachi Systems and JST SICORP

ⓒ2021 Information Processing Society of Japan