This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MCE.2021.3099634, IEEE Consumer Electronics Magazine

A Trusted Hybrid Learning Approach to Secure Edge Computing

Hichem Sedjelmaci[1], *Member, IEEE*, Sidi Mohammed Senouci[2], *Member, IEEE*, Nirwan Ansari[3], *Fellow, IEEE*, Abdelwahab Boualouache[4] , *Member, IEEE*
[1]Orange Labs, 44 Avenue de la République, 92320 Châtillon, France
[2]DRIVE EA1859, Univ. Bourgogne Franche Comté, F58000, 49 rue mademoiselle Bourgeois, 58000, Nevers, France
[3]Advanced Networking Lab., New Jersey Institute of Technology, Newark, NJ 07102, USA
[4] SnT- Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg

***Abstract*— Securing edge computing has drawn much attention due to the vital role of edge computing in Fifth Generation (5G) wireless networks. Artificial Intelligence (AI) has been adopted to protect networks against attackers targeting the connected edge devices or the wireless channel. However, the proposed detection mechanisms could generate a high false detection rate, especially against unknown attacks defined as zero-day threats. Thereby, we propose and conceive a new hybrid learning security framework that combines the expertise of security experts and the strength of machine learning to protect the edge computing network from known and unknown attacks, while minimizing the false detection rate. Moreover, to further decrease the number of false detections, a cyber security mechanism based on a Stackelberg game is used by the hybrid learning security engine (activated at each edge server) to assess the detection decisions provided by the neighboring security engines.**

INTRODUCTION

The Fifth Generation (5G) cellular network is empowering a new era of communication and computing in provisioning a variety of services such as efficient and reliable electricity distribution in a smart grid, augmented reality, intelligent transportation, industry 4.0 and telemedicine [1]. Mobile Edge Computing (MEC) is a main enabler for 5G since it provides computing capabilities within the proximity of mobile users (Internet of Things – IoT devices, smart meters, autonomous vehicles, drones, mobile phones, etc.) [2], with the aim to provide low latency and real-time access to network information for various emerging applications and services. Securing the MEC network is a fundamental issue because a variety of cyber-attacks could target its communication and computing capabilities for the attractive information processed at MEC servers. Recently, a new category of cyber-threats against the edge network was defined by cyber security expert [3],[4]. In these threats, as shown in Figure 1, the intruders aim to jam the communication between the servers and IoT devices, alter the sensitive data and hack the classification/detection decisions provisioned by the machine learning algorithms. Thereby, it is critical to secure the MEC network while the distributed edge nodes should be monitored and protected from the most complex and advanced attacks. In this article, we first review current defense mechanisms used to secure the distributed edge devices, by highlighting the static and dynamic malicious behaviors incorporated in these defense mechanisms. Then, we propose a new hybrid learning security framework that combines the experience and knowledge of a cyber security expert to improve the accuracy of attacks classification provided by machine learning. The cyber security expert feeds the detection framework periodically with attack signatures to improve the training process of the machine learning algorithm, thus enhancing the attack detection provided by the learning algorithm during the detection/classification process. The detection framework is based on a Generative Adversarial Network (GAN) approach with two discriminators: the Attack Detection Discriminator (ATDD) and the Anomaly Detection Discriminator (ANDD). These discriminators run deep learning algorithms and cooperate between each other to increase the attack detection rate while detecting unknown/zero-day attacks. As shown in Figure 1**,** at each edge server, a hybrid learning security engine based on the GAN approach is activated to monitor the network against attacks. However, the hybrid learning security engines embedded in the edge servers pose some risks since these security engines, which handle relevant security information, may potentially be hacked and infected by attackers. Thereby, to overcome this security issue, we propose a trusted security approach by incorporating a Stackelberg game to evaluate the detection decisions provided by the hybrid learning security engines and subsequently determine the false decisions generated by the infected ones.

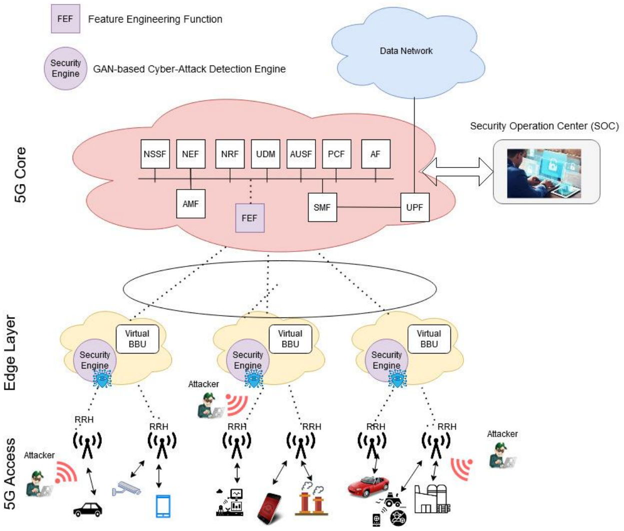


Figure 1. Cyber-attacks targeting the edge devices.

SECURITY AT THE EDGE: STATE OF THE ART

Depending on the technique used to detect attacks targeting the edge, we can classify the current techniques of detecting attacks in the edge network into two classes based on the attack behaviors.

Detecting attacks with static malicious behaviors

This detection technique relies on security rules for detecting attacks, where each attack's behavior exhibits certain signatures defined by the security expert. This technique detects only the attack that exhibits a static malicious behavior; however, the cyber-attacks that pose new threats and change their malicious behaviors frequently cannot be detected by this technique.

Baidya and Hewett [5] focused on securing the Software Defined Networking (SDN) based edge computing against network attacks. They addressed the security breaches caused by the flow rule attack, where the attacker targets a network switch by injecting false flow rules of routing. To mitigate the occurrence of this network attack, they developed a lightweight detection technique based on a set of attack signatures. The detection technique is activated at each SDN's edge controller to monitor and detect the flow rule attack.

Hassan *et al.* [6] proposed a distributed Snort-based Intrusion Detection System (IDS) to monitor and protect the cloud computing network and control system from Denial of Service (DoS) attacks. IDSs are activated at each server and collaborate with each other in a distributed manner to detect the distributed DoS attacks. They proposed a set of static rules related to the number of packets sent and dropped that should be conformed by the monitored nodes; when a node does not follow the rules, its behavior constitutes a DoS attack.

Cai *et al.* [7] developed a distributed signature-based attack detection to protect the metering infrastructure against false data injection attacks. They modeled the behavior of the attacks targeting the smart grid metering system with a set of signatures. The distributed IDSs based on attack signatures are activated at edge servers to detect the false data injection, while a reputation system is used to evaluate the trust level of the smart grid collector systems.

Detecting attacks with dynamic malicious behaviors

In this kind of detection, a machine learning algorithm is used to monitor the dynamic misbehaviors executed by the cyber-attack with the goal to detect attacks that have never been detected before by the signature-based detection technique, such as the zero-day threats. In the training phase, the algorithm aims to determine the distinguishable attack features and patterns with the purpose to detect new misbehaviors of cyber-attacks during the detection/classification phase.

Chen *et al.* [8] developed an accurate attack detection framework based on a deep belief neural network to protect the transportation system empowered by MEC from internal and external attacks. The unsupervised learning algorithm aims to determine the new attack features of threats on the MEC servers in order to improve the detection rate. The attack models used during the experimentation phase correspond to the network attacks that change their misbehaviors frequently and dynamically.

Samy et *al.* [9] proposed a robust defense system based on a deep learning algorithm to protect the IoT devices from several types of zero-day attacks. The defense system is activated at each edge server to monitor the behavior of the distrusted IoT devices. According to their experimental results, their defense system exhibits a low reaction time and high accuracy detection. However, they did not propose a security strategy to protect the defense system against the internal and external attacks.

Subramaniam *et al.* [10] studied different deep learning algorithms deployed at the edge to secure a MEC network. According to their investigation, the automated learning attack features used by the deep learning algorithms could improve the detection rate significantly as compared to other machine learning algorithms such as random forest and one-class Support Vector Machine (SVM). However, they did not evaluate the performance of the algorithms with real data.

Wang *et al*. [11] analyzed the performance of the current IDSs based on machine learning algorithms deployed at the Internet of Things (IoT) edge. The performance of machine learning is evaluated in terms of three main metrics: detection accuracy, memory storage, and complexity. The purpose of this study is to embed the selected machine learning algorithm(s) with the IDS at the real edge server to secure the IoT network.

Table 1 summarizes the pros and cons of the edge security frameworks [5]-[10] based on the following security and network metrics: detection and false positive rates, computational overhead and memory storage. The detection and false positive rates correspond to the detection accuracy of security frameworks against the known and unknown attacks. The computational overhead represents the required cost of the security framework to achieve a high level of security, while the memory storage metric corresponds to the number of attack signatures and features required to detect the attacks accurately.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MCE.2021.3099634, IEEE Consumer Electronics Magazine

Table 1. Comparison among edge security frameworks

Edge security frameworks	Detection rate	False positive rate	Computational overhead	Memory storage
Baidya *et al.* [5]	Low	Low	Low	Medium
Hassan *et al.* [6]	Medium	Medium	Medium	High
Cai *et al* [7]	Medium	Medium	Medium	High
Chen *et al.* [8]	High	Medium	High	Low
Samy et *al.* [9]	High	Meduim	High	Medium
Subramaniam *et al.* [10]	High	Medium	High	Medium

Among the AI detection techniques that could be leveraged in the cyber security context for achieving high detection accuracy, i.e., high attack detection with a low false positive rate, the hybrid learning technique [3] seems promising. This technique combines the rules (signatures) defined by the cyber security expert and attack models obtained from the machine learning algorithm to increase the detection accuracy. In this context, there is a need to better understand the human-machine interaction in addressing the accuracy of attack detection with consideration of the network constraints such overhead, latency and memory storage.

Another security issue that has not been addressed in current works of edge computing security is the trustworthiness of the deployed security engines. In fact, a security engine that is activated at the edge could be infected by attacks, thus resulting in a false detection against the monitored target, i.e., categorizing a legitimate target as an attacker and vice versa. Therefore, it is mandatory to evaluate the trust level of a security engine against the detected attack.

A Trusted Hybrid Learning Framework to Secure Edge Computing

According to the meticulous investigation done on current security systems applied in MEC and this is among the first trusted hybrid learning framework to detect the most advanced attacks targeting the edge network. We design a security framework that defines engines embedded in each edge server to protect the MEC network from attacks. The main components of our trusted hybrid learning framework are detailed in the following.

Hybrid learning security framework

As illustrated in Figure 2, the proposed framework is equipped with two main security engines: feature engineering function and GAN-based Cyber-Attack Detection Engine.

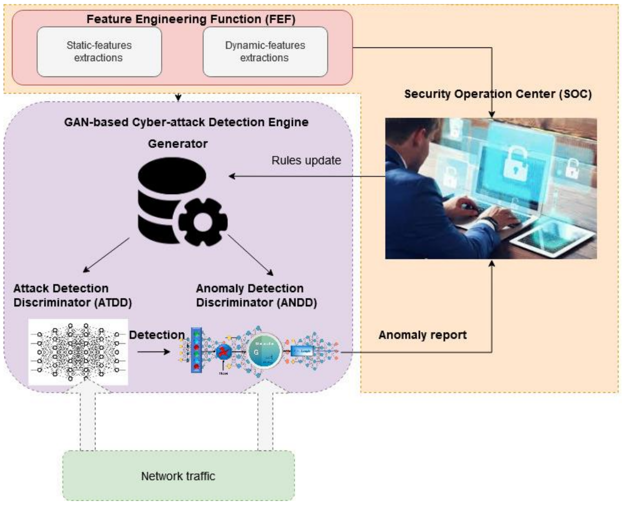

Feature Engineering Function (FEF)
Static-features extractions
Dynamic-features extractions
Security Operation Center (SOC)
GAN-based Cyber-attack Detection Engine
Generator
Rules update
Attack Detection Discriminator (ATDD)
Anomaly Detection Discriminator (ANDD)
Detection
Anomaly report
Network traffic

Figure 2. Hybrid learning security framework

Feature Engineering Function (FEF): As shown in Figure 1, this software function should be deployed in the 5G core system. It focuses on determining the attractive feature vectors to be utilized by the attack detection engine. Feature determination relies on static and dynamic approaches. In a static approach, the cyber security expert feeds the feature engineering engine with feature vectors related to some identified attacks. Here, the security expert analyzes the security alerts generated by the centralized security monitoring system, Security Operation Center (SOC), and then defines a set of feature vectors related to the new attack pattern, i.e., zero-day threats. In a dynamic approach, the feature vectors are updated dynamically, i.e., without intervention of the security expert. Here, an unsupervised machine learning technique based on game theory is used by the FEF engine to determine the potential future attack behaviors and then identify the new features related to each zero-day attack. The security game is a non-cooperative game between the FEF located at the 5G core level and an attacker (located within the neighborhood of the security engine). Here, the action of FEF is to monitor the suspected targets (located within the same neighborhoods of the security engines) and the action of an attacker is to launch an attack against the neighborhood target. The purpose of this security game is to determine the state (Nash Equilibrium) at which the FEF monitors the target instigated by the attacker at the same time. In this case, the FEF identifies the attack features prior to the attack. Readers are referred to [12] for more details about the dynamic determination of features by using game theory.

GAN-based Cyber-Attack Detection Engine: This engine is mainly based on a hybrid learning approach to detect the attacks instigated within the MEC network as shown in Figure 1. The hybrid approach in the context of cyber security relies on a combination between the knowledge and expertise of the cyber security expert and the strength of machine learning detection and classification against attacks. As explained in the introduction section, the positive aspect of the hybrid learning approach is to achieve a high zero-day attack detection rate by using a robust machine learning algorithm, while the false positive rate is reduced with the help of the cyber security expert intervention. The cyber-attack detection engine is based on the Generative Adversarial Network (GAN) approach to monitor the feature vectors delivered by the feature engineering function and to detect/predict attacks. Our GAN-based Cyber-attack detection engine consists of a generator and two discriminator systems: the attack detection discriminator (ATDD) and the anomaly detection discriminator (ANDD). GAN-based Cyber-attack detection engine is a closed loop system. The generator is based on a rule-based detection technique. The rule-based detection technique, executed at the generator system, corresponds to a set of attack signatures defined by the security expert, which are updated over time and depend on the number of suspected behaviors that are identified by the experts as possible attacks. An example of an attack signature could be the number of packets sent and dropped that are above certain thresholds to detect the Denial of Service (DoS) attack, where the packets sent and dropped are the relevant features. A set of attack signatures are stored in the signatures database of the generator.

In this cyber security context, the generator and discriminators cooperate between each other to increase the attack detection rate and reduce considerably the number of false positives. This cooperative detection is illustrated in Figure 2. The generator sends to the discriminators a list of attacks to be detected (with the related signatures) and then each discriminator extracts its required information. Specifically, in the pre-processing phase, ATDD extracts the values of features from the signatures and associates them with their corresponding labels while ANDD extracts only the values of features from the signatures without associating them with their labels. Afterward, discriminators execute the training process by using these feature values as inputs. ATDD runs a supervised multi-class deep learning algorithm [13] that builds the normal and attack patterns during the training process and then classifies the new incoming data as an attack or normal behavior during the classification/detection process according to the patterns determined in the training process. The normal and attack patterns obtained during the training process correspond to clusters of vectors related to each pattern. ANDD uses as inputs the outputs of ATDD and runs a deep convolutional generative adversarial network [14]. This discriminator also analyzes the network traffic to detect anomalies. If an anomaly is detected, ANDD first checks if it is also detected by ATDD. In the case of this anomaly only detected by ANDD, an alert message is sent to the security expert in order to update the attack signature data base, i.e., adding new attack signatures. The alert message includes attack features along with the type of attack. Note that the cyber security expert feeds the generator periodically with new attack signatures in order to improve the training process and hence to increase the attack detection rate. In addition, the security expert investigates the detection accuracy of ATDD against zero-day attacks with the goal to decrease the false positive rate. Note that zero-day attacks, detected by the discriminator, correspond to the attacks that are not defined in the signature database of the generator. We believe that the number of interventions of the security expert in our framework keeps decreasing over time, since most of unknown/zero-day attacks will be identified, and more advanced attacks will take time to appear.

Stackelberg trust game

The hybrid learning security engines based on the GAN approach embedded in MEC servers cooperate with each other by exchanging their detection decisions (i.e., attacks with the related signatures and features) in order to detect accurately the new category of attacks, i.e., zero-day attacks. However, a hybrid learning security engine could be infected by the attacks and the infected security engine could convey false detection decisions to its neighboring engines; therefore, the accuracy detection of zero-day attacks is impacted, i.e., the increase and decrease of the false positive and detection rates, respectively. Thereby, to overcome this security issue, each hybrid learning security engine should monitor the trustworthiness of the detection decisions provided by its neighboring security engines with whom it collaborates. The interaction between security engines is modeled as a Stackelberg security game, where the hybrid learning security engine is the leader player and its neighboring security engines are the follower players. In the security game, we assume that almost all follower players are selfish and aim to impact the attack detection of the leader player by providing false detection decisions. In this non-cooperative game, the leader player aims to maximize its utility with consideration of the best strategies undertaken by the follower players and vice-versa. In the Stackelberg game, the expected utility functions of the non-cooperative players depend mainly on the number of malicious hybrid-learning security engines that the leader player detects and the number of false detections that the leader player generates against the malicious follower players.

In the proposed Stackelberg security game, the leader player launches its optimal strategy for detecting the malicious hybrid learning security engine, by considering the best response of the follower's strategy. Furthermore, the malicious security engine executes its optimal strategy for providing a false detection decision, by considering the best response of the leader's strategy. Note that the best responses of the players' strategies are the total number of malicious hybrid-learning security engines that are detected accurately by the leader player and false detection decisions provided by the follower player without being detected by the security engines. Therefore, the optimal strategies of the non-cooperative players are determined by estimating the optimal state of the players defined as a Stackelberg Equilibrium (SE), which corresponds to the state when the follower and leader players respectively attack and monitor (by executing the hybrid learning security framework) the same target, i.e., the MEC server where the hybrid learning security engine is activated. We conclude that when SE is reached, the follower player executes a malicious behavior against the leader player by providing a false detection decision. In this case, the leader player categorizes the follower player as an attacker that injects a false detection during the decision-making process.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/MCE.2021.3099634, IEEE Consumer Electronics Magazine

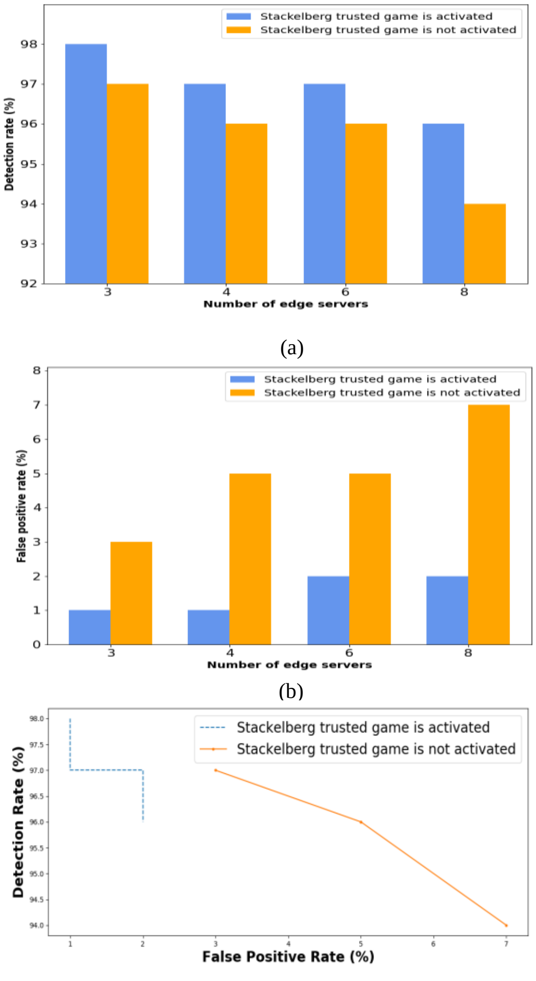
PERFORMANCE EVALUATION

Experimental setup and security metric

We use the dataset [15] to conduct the performance evaluation of the proposed trusted hybrid learning framework. This dataset has one type of normal traffic and 9 types of attack traffics. 175,341 and 82,332 are respectively the numbers of records in the training and testing/detection phases, where 49 features are defined. In our experiments, the deep learning algorithm of the employed GAN is composed of 5 hidden layers with 20 neurons in each hidden layer; the learning rate is equal to 0.2 and the number of iterations is equal to 150. In our training and detection phases, we use 4 types of attacks, namely, fuzzing, DoS, reconnaissance and Worms. Note that, in the testing/detection phase, the behaviors of these 4 types of attacks are not the same as those used during the training process. In our experiment, these 4 types of attacks in addition to the attacks that generate false detection decisions (executed by malicious hybrid learning security engines) correspond to the attack model that is used by the cyber attackers for executing distributed botnets and distributed DoS against the IoT edge network. In the security analysis, the security defense metric is computed to evaluate the number of cyber-attacks that are detected by the trusted hybrid learning framework and the number of false detections generated by the framework. The security defense rate is computed as the attack detection rate minus the false positive rate. To ensure the feasibility of the proposed security learning framework, the attack detection rate should be above 50%. The attack detection rate corresponds to the ratio of the sum of the number of new attacks that the deep learning algorithm has detected and the number of known attacks that the rule-based detection has detected to the total number of attacks launched against the edge network. The false positive rate depends on the number of false detection that the hybrid learning security framework generates, specifically against the zero-day threats.

Experimental results

As shown in Figure 3, we analyze the detection and false positive rates of the proposed hybrid learning framework with respect to whether the Stackelberg trust game is activated or not. Here, we vary the number of hybrid-learning security engines that are activated at each edge server from 3 to 8, while approximately 1/3 of these distributed security engines are malicious, infected by the attackers. From Figure 3(a), it is apparent that the hybrid learning framework exhibits a high detection rate when the network attacks attempt to target the edge severs and in the worst case (i.e., approximately 1/3 of the eight edge servers are infected by the attackers) the detection rate is almost equal to 96%. However, from Figure 3(b), the false positive rate of the trusted hybrid learning framework is low as compared to the security framework that does not activate the Stackelberg trust game. This is attributed to the distributed security game that aims to analyze the decisions provided by the hybrid learning security engines and hence to identify the security engines that provide fake decisions in order to deceive the judgment of the cooperative trusted hybrid learning security engines, i.e., detecting the normal node as an attacker and vice versa.



(a)

(b)

(c)

Figure 3. Performance of the Stackelberg trust game: (a) detection rate, (b) false positive rate, and (c)ROC curve.

As shown in Figure 4, we compare the performance of our trusted hybrid learning framework with current security framework [8] when applied to MEC networks. As explained earlier, the IDSs in [8] run the deep belief neural network algorithm to monitor and protect MEC from attacks. In this analysis, we assume that our hybrid learning framework and the security framework [8] are not infected. In the experimental study, we vary the amount of malicious traffic instigated by the attackers from 15% to 40% of the total traffic. As illustrated in Figure 4, the security defense rate of the proposed hybrid learning framework is high as compared to the current IDSs based on the AI algorithm. This result is attributed to two main reasons. First, the hybrid learning approach leverages the rule-based detection (defined by the security expert) to feed the learning algorithm with new and relevant training data, hence improving the attack detection rate. To reduce the false positive rate, specifically against zero-day threats, the cyber security expert interacts with the decisions of machine learning with the goal to correct the false detection and hence reduce over time the number of false positives that the algorithm generates. Second, the Stackelberg trust game reduces further the false detection rate generated by the hybrid learning framework. The security approach based on Stackelberg games is used to analyze the decisions against the detected attacks provided by the machine learning and rule-based detection techniques.

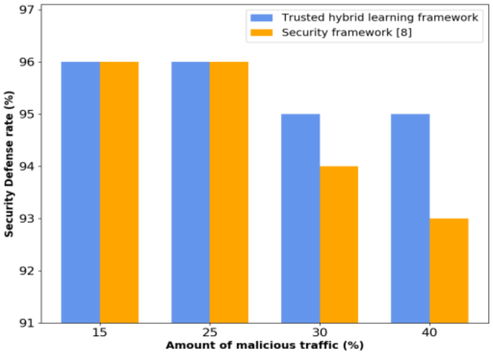


Figure 4. Security defense rate of the security frameworks

Security of trusted hybrid learning framework

The main purpose of our hybrid learning framework is to protect the MEC network from attacks that target the edge devices, such as IoT devices and edge servers. Among the network attacks that our framework can prevent, we consider, as examples, the distributed DoS and distributed botnet attacks that aim, for instance, to drop the relevant data and send a huge amount of unwanted data in order to degrade the edge quality of service. Furthermore, a security game based on the Stackelberg approach is proposed to prevent the malicious hybrid learning security engine on generating false detection decisions within its neighborhood. The main security objectives of our framework are to ensure data integrity and to mitigate external and internal attacks.

Conclusion

In this article, we have proposed a new hybrid learning framework to secure the edge computing network from the most advanced attacks, i.e., zero-day threats. The feature determination and attack detection leverage merits of AI techniques (e.g., rule-based on feature extraction and attack detection, and the machine learning algorithms in distinguishing features and detecting attacks). However, security engines based on machine learning algorithms could be hacked by attackers, e.g., AI-attacks, which aim to alter the training vectors of the algorithms and hence lead the security engines to yield false attack detections, thus increasing the false detection rate. Therefore, to overcome this issue, we have proposed, developed and demonstrated a trusted security game approach to examine the detection decision provided by distributed security engines.

References

[1] 3GPP TS 23.501 v15.1.0, System Architecture for the 5G System; Stage 2; 03/2018.

[2] X. Sun, N. Ansari, "EdgeIoT: Mobile Edge Computing for the Internet of Things," *IEEE Communications Magazine*, vol.54, no.12, pp. 22-29, Dec. 2016.

[3] L. Xiao, X. Wan, C. Dai, X. Du, X. Chen, M. Guizani, "Security in Mobile Edge Caching with Reinforcement Learning", *IEEE Wireless Communications*, Vol.25, Issue.3, 2018, pp. 116-122.

[4] D. Puthal, S. P. Mohanty, S. A. Bhavake, G. Morgan, R. Ranjan, "Fog computing security challenges and future directions [energy and security]", *IEEE Consumer Electronics Magazine*, vol.8, pp.92-96, 2019.

[5] S.S. Baidya, R. Hewett, "SDN-based edge computing security: detecting and mitigating flow rule attacks", *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, Arlington Virginia, USA, 2019, pp. 364-370.

[6] Z. Hassan, Shahzeb, R. Odarchenko, S. Gnatyuk, A. Zaman, M. Shah, "Detection of Distributed Denial of Service Attacks Using Snort Rules in Cloud Computing & Remote Control Systems", *IEEE 5th International Conference on Methods and Systems of Navigation and Motion Control (MSNMC)*, Kiev, Ukraine, 2018, pp.283-288.

[7] Z. Cai, B. Qian, Y. Xiao, "Edge Computing Based Bad Metering Data Detection", *IEEE 3rd Conference on Energy Internet and Energy System Integration (EI2)*, Changsha, China, 2019, pp.693-698.

[8] Y. Chen, Y. Zhang, S. Maharjan, M. Alam, T. Wu, "Deep Learning for Secure Mobile Edge Computing in Cyber-Physical Transportation Systems", IEEE Network, Vol 33, Issue 4, 2019, pp. 36-41.

[9] A. Samy, H. Yu, H. Zhang, "Fog-Based Attack Detection Framework for Internet of Things Using Deep Learning", *IEEE Access*, Vol. 8, 2020, pp. 74571 - 74585.

[10] P. Subramaniam, M. Jeet Kaur, "Review of Security in Mobile Edge Computing with Deep Learning", *IEEE Advances in Science and Engineering Technology International Conferences (ASET)*, Dubai, United Arab Emirates, 2019.

[11] H. Wang, L. Barriga, A. Vahidi, S. Raza, "Machine Learning for Security at the IoT Edge - A Feasibility Study", *IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems Workshops (MASSW)*, Monterey, CA, USA, 2019, pp.7-12.

[12] X. Sun, Y. Liu, J. Li, J. Zhu, X. Liu, H. Chen, "Using cooperative game theory to optimize the feature selection problem", *Neurocomputing*, Vol. 97, Issue 15, 2012, pp. 86–93.

[13] E. Eziama, K. Tepe, A. Balador, K. S., Nwizege, L. M. Jaimes, "Malicious node detection in vehicular ad-hoc network using machine learning and deep learning". *In 2018 IEEE Globecom Workshops (GC Wkshps) (pp. 1-6). IEEE.*

[14] T. Schlegl, P. Seeböck,, S.M. Waldstein, U. Schmidt-Erfurth, G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery". *In International conference on information processing in medical imaging (pp. 146-157). Springer, Cham.*

[15] M. Nour, J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)", *IEEE Military Communications and Information Systems Conference (MilCIS)*, Canberra, ACT, Australia, 2015.

About the Authors

Hichem Sedjelmaci is a Senior R&D and Projects Manager at Orange Labs, Châtillon, France. Contact him at hichem.sedjelmaci@orange.com.

Sidi-Mohammed Senouci is a full professor at university of Burgundy in France. Contact him at sidi-mohammed.senouci@u-bourgogne.fr.

Nirwan Ansari is a Distinguished Professor at the New Jersey Institute of Technology (NJIT), New Jersey, USA. Contact him at nirwan.ansari@njit.edu.

Abdelwahab Boualouache is a research associate at University of Luxembourg. Contact him at abdelwahab.boualouache@uni.lu.