[1]Urvashi Sangwan

2Dr. Rajender Singh Chhillar

Efficient Cyber Security Framework for IoT Using Machine Learning Algorithms

Abstract: - Through the network's infrastructure, the IoT can impart perception, recognition, and remote control to inanimate objects. Due to IoT's characteristics, it's possible to integrate the real world into a digital system, which improves precision, productivity, and bottom line. While traditional internet infrastructure comprises of highly capable computers and servers, IoT gadgets have limited processing power and storage space. The authentication and key agreement system SecureAuthKey is very lightweight. The suggested technique is meant to solve the security and privacy problems that plague modern constraint-based CPS programmes. A lightweight approach for authenticating cyber-physical objects is one of the expected outcomes. Adaptation and autonomy in cyber-physical systems need not be compromised by a lack of trust or security in users' data or the devices themselves, according to a new type of security algorithm. To provide flexibility and scalability, a new middleware module has been developed on the Raspberry platform to facilitate communication between CPS-based devices and services. Secure Internet of Things (IoT) evaluation methodology that may be used in a variety of contexts and is user-focused. With the suggested system, the two CPS units will be able to authenticate one another. When a network grows larger, the possibility of an attack rises. Therefore, the IoT network is much more susceptible to attacks than conventional networks. As the number of connected devices grows, so do the number of potential threats to their security. To protect the IoT ecosystem from current threats, cutting-edge technology is essential. The proposed approach must be very efficient and have low computational overheads. It creates random session keys to protect wireless transmissions. The system is protected from a variety of cyber threats. The current constraint-based CPS system requires a new lightweight security solution.

Keywords: IoT Security, CPS, Machine Learning Algorithms, Lightweight Security, Authentication, SecureAuthKey, Threat Detection, Network Security, Feature Engineering, Deep Learning

1. Introduction

Some CPS-based applications make advantage of small, diverse physical components. The current security methods work best on CPS-based, heavily-loaded systems that have stronger hardware support. Some examples of such hardware components are increased processing and memory support, as well as sufficient energy backup. As a result, conventional safeguards work well in such a setup. The problem is that not all CPS-based devices have the hardware or software to use modern authentication and encryption techniques. The existing security technique cannot be applied due to hardware constraints [1]. Existing research solutions and research projects focus on a specific security property, such as authentication, privacy, or authorisation, rather than building a universal and adaptable framework that can serve the needs of a CPS-based application. Existing research solutions for CPS devices do not follow tried and true operating system principles while creating said solutions [2]. Unfortunately, not all security measures proved effective when used in a distributed system. Most current methods place a heavy computational burden on authenticating clients' devices while ignoring other possible security flaws. Moreover, eavesdropping and other security issues are often overlooked during the system design process.

[1] Ph.D Scholar Maharshi Dayanand University, Rohtak

[2]Professor Maharshi Dayanand University, Rohtak

Corresponding author mail : usangwan@gmail.com

Second author mail : Chhillar02@gmail.com

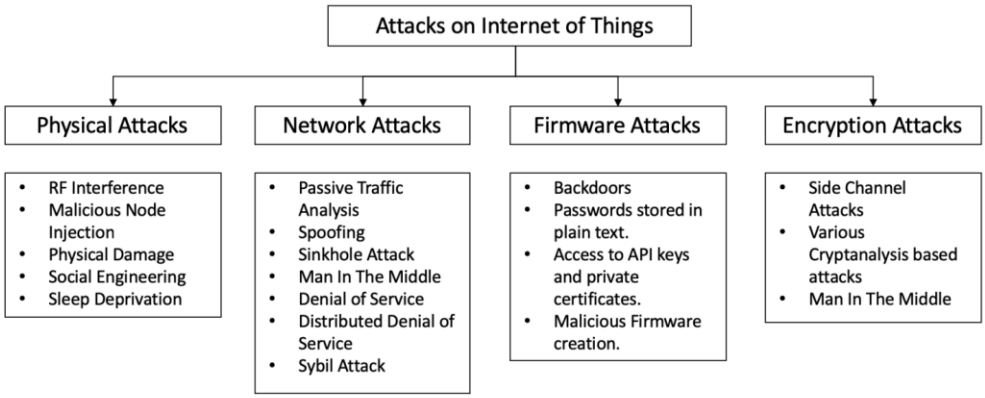

Fig 1: Types of attacks on IoT

Small CPS applications typically employ network nodes that are low in compute power, memory, and power. Furthermore, these gadgets begin communicating via the open wireless channel. As a result, users' privacy and security are put at serious danger, and hackers have a window of opportunity to launch a variety of attacks against them [3]. A secure end-to-end key agreement mechanism is crucial to ensuring the confidentiality of communications between two devices. The secret key is distributed between nodes via the key agreement procedures. However, typical cryptographic operations of the key establishment, like Rivest-Shamir-Adleman (RSA) or Elliptic Curve Cryptography (ECC), may be too resource-intensive and computationally intensive for heterogeneous CPS devices to handle. Due to the wide variety of CPS devices and the limited resources available to each, a lightweight security mechanism was needed. Additionally, it protects against any sort of cyber assault.

As has been shown in a number of studies, IoT devices can be easily breached by cybercriminals. The variety of possible assaults on IoT networks is depicted in Figure 1. Several types of assaults fall into this category, such as those involving physical means, networks, firmware, and encryption.

If you want to compromise a gadget, a physical attack is the way to do it. Such assaults have the potential to disable the gadget or alter its behaviour. An opponent needs to have physical possession of the devices in order to launch these kinds of attacks. Possible outcomes of these attacks range from mere inconvenience to actual bodily injury [4]. However, due to the diversity of the IoT, not all devices on the network may be able to benefit from these security measures. In addition, outdoor IoT devices, including low-cost sensors with minimal security measures, may see a significant increase in the computational and communication loads caused by such solutions.

2. Background

In this research, we offer a method for automatically generating the MUD profiles of IoT devices, which can subsequently be used to classify similar devices in the network and detect anomalies in their network traffic, such as those caused by compromised firmware or a hacker. The Software Defined Networking (SDN) architecture then takes this formal behavioural profile and translates it into static and dynamic flow rules that the network can enforce at runtime [5-9]. Traffic that follows these guidelines is permitted through, while all other traffic is screened for threats. The IDS's workload is greatly reduced in this manner, allowing it to scale effectively and detect threats unique to individual devices. After the MUD policy rule has been generated, we use multi-stage machine learning models to detect volumetric attacks and pinpoint aberrant microflows (5-tuple). In conclusion, we propose a methodical attempt to model (statically) and enforce (dynamically) cyber-security for large-scale IoT systems.

Various IoT security studies have been conducted over the past few years, each focusing on a different facet of IoT ecosystem security. Sensing, network, middleware, gateway, and application security were the primary attention of the authors of [13]. Researchers have examined the security of the Internet of Things in contexts as varied as the smart home, public transit, healthcare, and wireless sensor networks. In [10], the authors analysed

the security of Internet of Things (IoT) gadgets that use wireless protocols like WiFi, NFC, Bluetooth, and Zigbee. The authors of [16] conducted a survey focusing on Internet of Things network attacks.

In order to control and manage hardware in the real world, In order to be considered a cyber-physical system, it must include computational complexity, communication, detection, sensing, and programming interfaces [11-14]. Cyber-physical systems are able to observe their physical environment, make judgments (such as whether or not to flip a switch), and act accordingly (ON or OFF the switch). In this iteration, we focus on tools and procedures for keeping hackers out of our cyber-physical infrastructure. The method allows cyber-physical systems in the house to securely and privately provide authorization to linked devices via gateway devices. Cyber-physical system components used in smart home applications are typically implemented on tiny chips that also provide network connectivity. This leaves them open to attacks on their networks. Protecting the privacy and security of online interactions between users and smart home devices is a primary goal of the current system and technique.

3. Methodology

While there has been substantial progress made in the study of automated model selection and HPO, Feature Engineering (FE), an essential part of the ML pipeline, has been overlooked in many AutoML implementations. When training ML models, original feature values are not always optimal for achieving optimal prediction accuracy. Feature addition and removal may be necessary to make the system more task-appropriate [18]. Feature engineering's goal is to supply ML models with high-quality input features to use in making decisions from the data. To what extent ML can be used is limited by FE [4]. Engineering features manually is a time-consuming process that frequently necessitates expert knowledge in a certain domain. Feature engineering can be automated with the help of Automated Feature Engineering (AutoFE), which does so by automatically creating and selecting features that are pertinent to a given challenge. Compared to human feature engineering, AutoFE speeds up the creation of more precise learning models and improves their consistency [14]. FE techniques can be broken down into three distinct groups: feature creation, feature selection, and feature extraction. Extending the feature spaces requires new features, which can be generated by combining or transforming existing features. By focusing on the most important and relevant elements, feature selection helps cut down on unnecessary repetition. Using mapping functions, feature extraction might create new features, although its main objective is to reduce feature dimensionality. AutoFE is a dynamic mix of these three components.

The generate-and-select method is often used by cutting-edge AutoFE methods like AutoFeat. Using this method, a large pool of candidate features is generated, from which the most informative and useful ones are chosen. Optimizing processes like feature selection and extraction can help find the best possible model's parameters.

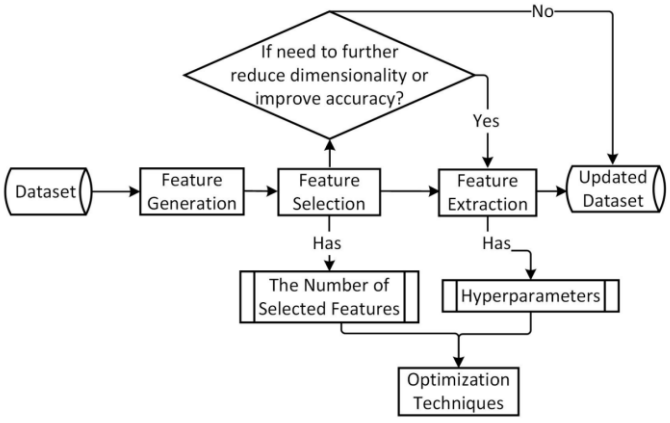

Fig 2: flowchart for the automated feature extraction process for the proposed method

Multiple points of the AV system are outfitted with the suggested IDS to ensure the safety of all internal and external communications. The IDS can be mounted on top of the CAN bus, where it can process every transmitted message and check for compromised nodes [10]. This helps to detect attacks on the CAN bus and keep it secure. The proposed IDS can also be installed inside the gateway to safeguard the additional networks [11]. Figure 2 depicts the network setup for an IDS in a car. When a node on the CAN bus receives a message from another device on the CAN bus, consisting mostly of an ID and data field, it sends the message to the IDS to determine if the signal line of the CAN bus has switched from CANH to CANL. Likewise, when a message travels from the Internet to an internal network, it is inspected by the IDS located within the gateway.

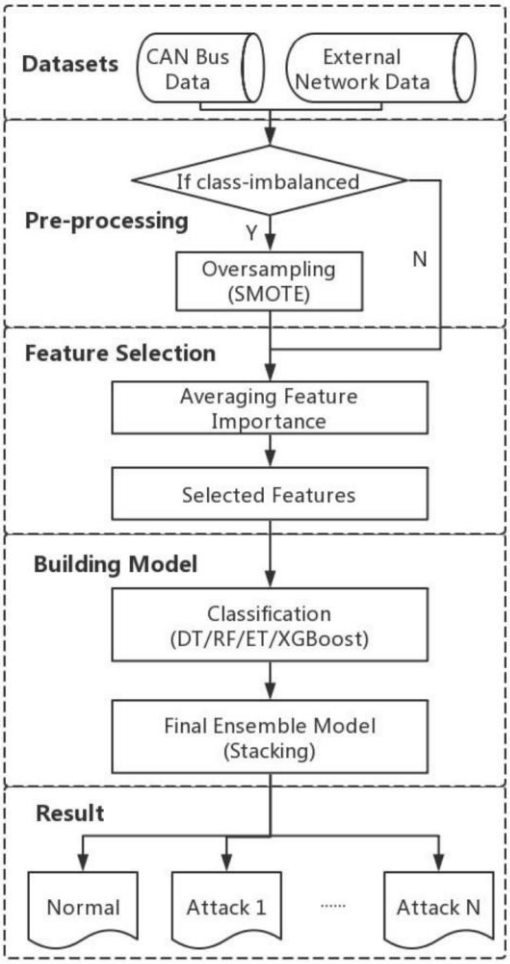

Fig 3: Proposed Attack detection process

Recently, malicious domain name detection has made use of machine learning (ML) techniques and deep learning (DL) structures. This machine learning-based classifier resides in a network and monitors DNS queries for suspicious activity. Then it attempts to locate DGAs. In addition, if the DGA classifier identifies a DGA-generated domain, it alerts the network administrator. Retrospective and real-time DGA classifiers are the two main types of extant work on constructing ML and DL based classifiers. Since retrospective detection relies on modelling many predictions across several domains, it is not practical for usage in real time. These techniques increase performance by clustering domain names using a variety of statistical tests, such as Kullback-Leibler divergence testing, as well as contextual information and HTTP headers. These procedures require a lot of processing power. In contrast, real-time detection approaches classify websites on a per-domain basis without considering any extra context. Fast though they may be, such technologies fall short of the precision of their retrospective counterparts. Better performance can be attained in both retrospective and real-time based systems, however deployment of these methods is often constrained in real-time systems. This is mostly because these techniques rely heavily on in-house data.

Based on these considerations, we designed the AmritaDGA database and the AmritaDeepDGA baseline system to detect and categorise dangerous domain names. Recurrent neural networks (RNNs), long short-term memories (LSTMs), gated recurrent units (GRUs), convolutional neural networks (CNNs), and a CNN-LSTM hybrid make up the baseline system. Each model utilised Keras character embedding to change the domain names from their original character representation to a numerical one.

4. Results

Common connections between nodes are depicted in a sequence diagram. The communication between the smart device, gateway server, and Secure CPS system is depicted in figure 4. is also indicating certain node-internal processes. Sequence diagram illustrating the execution of the SecureAuthKey algorithm between a gateway server and a smart device. The CPS Application is the user interface for managing smart devices via a gateway server.

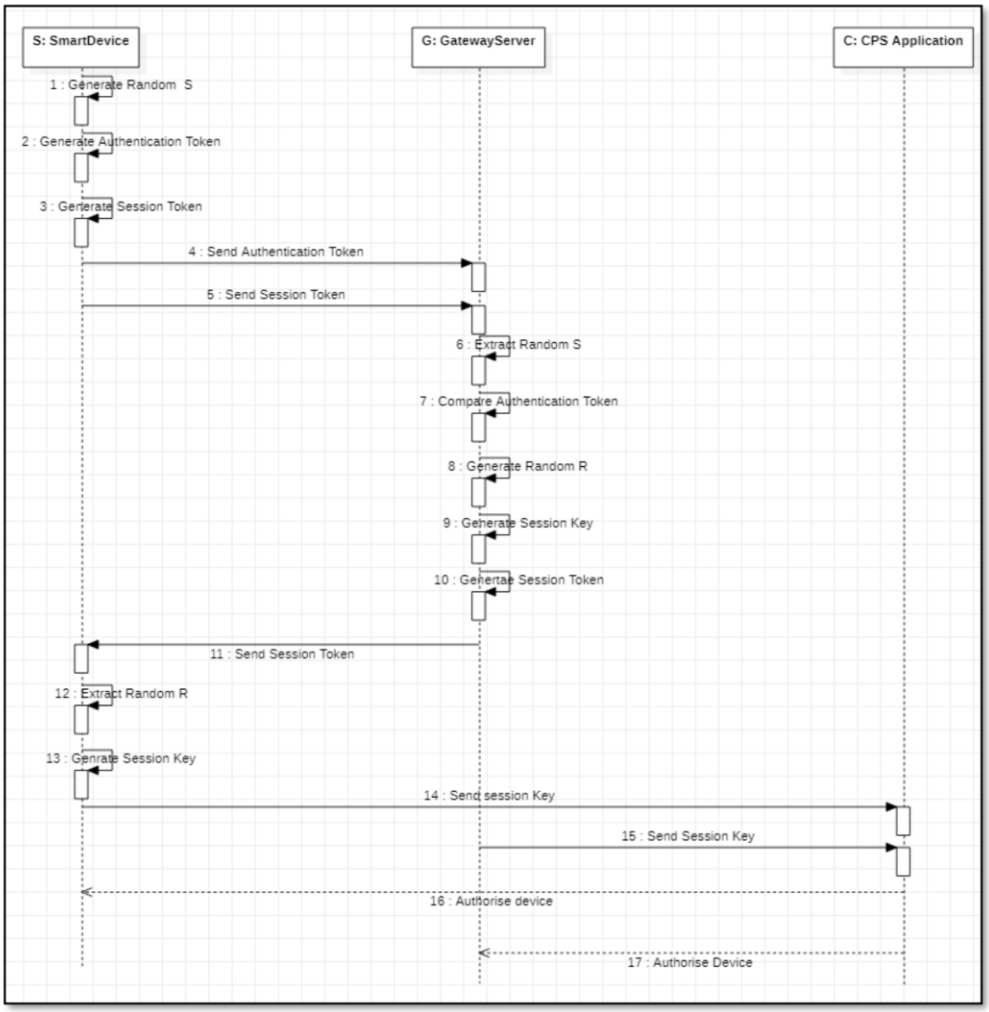

Fig 4: Sequence diagram for the cyber security attack detection method

The deployment diagram represents the tangible elements of an Object Oriented system. The hardware components needed to create and test the SecureAuthKey algorithm are seen in Figure 6.2.10. Building a wireless network that connects multiple devices requires a Wi-Fi router. Any electronic gadget that can access the internet is considered a user interface device. The user interface device can be a mobile phone or a laptop computer. The SecureAuthKey Algorithm was included into the smart home infrastructure. A Raspberry Pi can act as a gateway server because it is a microcontroller.

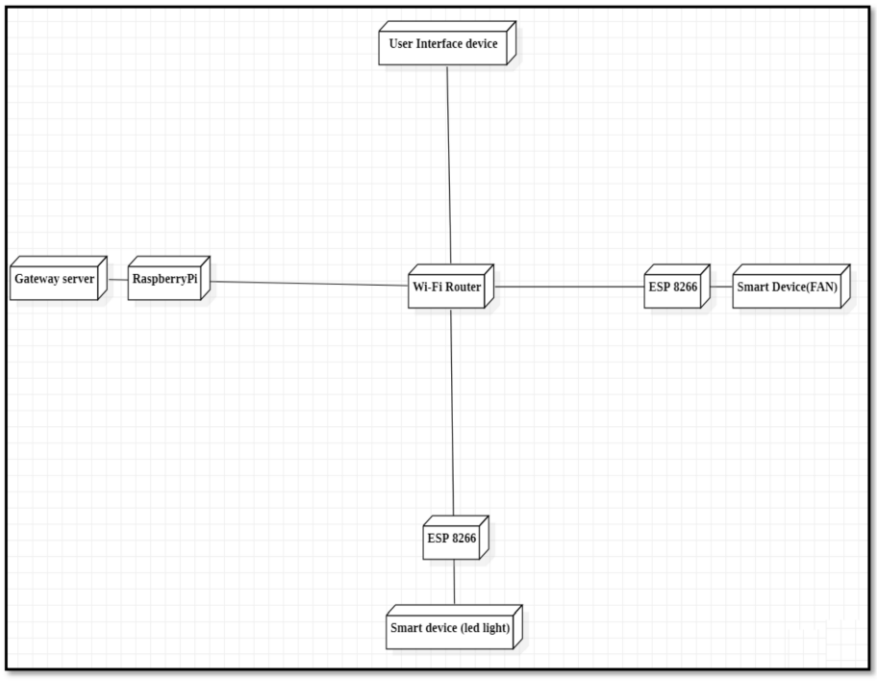

Figure 5: Use case diagram for the proposed methodology

Most DL architectures had greater than 95% train accuracy over 15 iterations on the DS1 data set. There was no improvement in train accuracy across all DL architectures after 15 iterations. Most significantly, overfitting began to negatively impact RNN performance. Last but not least, after running for 100 epochs, all DL architectures, with the exception of RNN, reached good performance. When compared to other DL architectures, LSTM train accuracy was high for the DS2 dataset. As more and more epochs passed, the LSTM's performance improved. As a result, it appears that the LSTM is the superior architecture. However, until 10 epochs, GRU's train accuracy was comparable to LSTM's; after that, GRU's performance steadily declined, and by the time it reached 100 epochs, it was quite poor. After 10 epochs, the accuracy of the CNN and CNN-LSTM architectures remained constant, whereas it dropped to 90% for the RNN. Even after 10 epochs, RNN's performance was dismal. In general, the number of training epochs needed to master a feature varies across DL architectures.

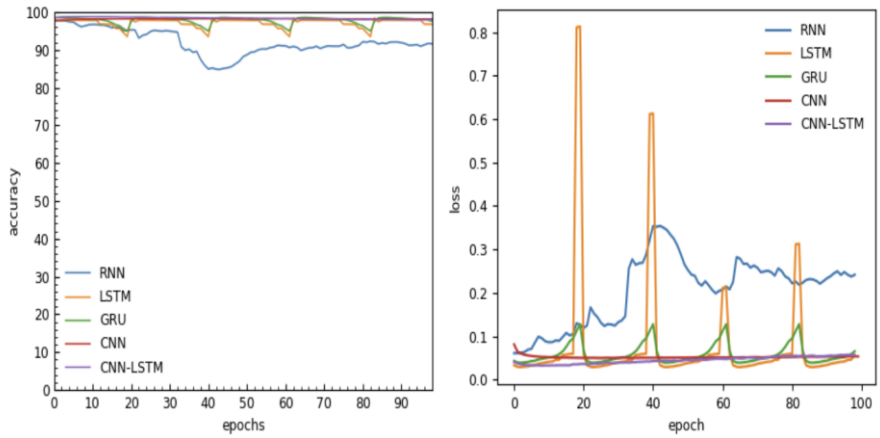

Fig 6: Accuracy and Loss visualization for the Dataset 1

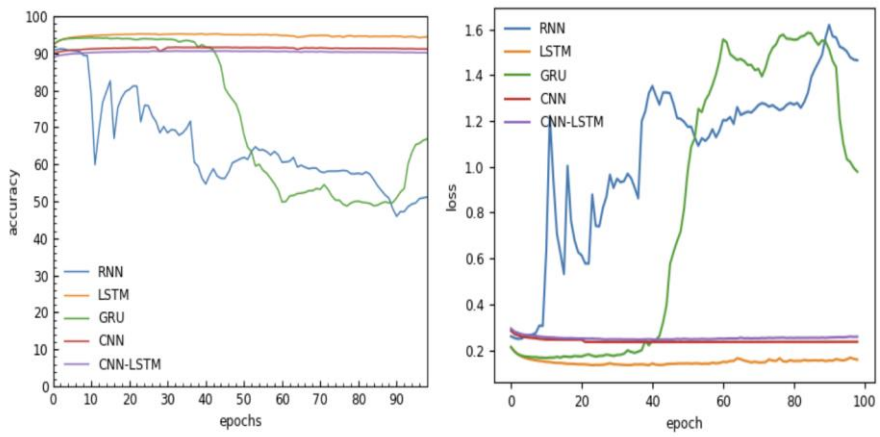

Fig 7: Accuracy and Loss visualization for the Dataset 2

5. Conclusion

Many manufacturers are leasing brand-new, vulnerable devices to the market, and they are being widely distributed. Many cyber attacks, such as those that breach privacy, gain unauthorised access, or create denial of service, have been traced back to these gadgets. First, we discussed the importance of IoT network security and the components of its design, before moving on to discuss the roles and issues faced by various stakeholders in the IoT ecosystem. Our subsequent work zeroed in on potential dangers to the security of IoT networks, as well as their repercussions and potential solutions, allowing us to pinpoint the issues that need more investigation. Machine Learning (ML) and Deep Learning (DL) algorithms are excellent for IoT data analytics for autonomous vehicles, smart cities, smart homes, electronic healthcare, and IoT security. However, certain IoT applications require human competence to build efficient ML models, restricting their usability. Thus, automated ML (AutoML) is a viable solution for generating ML models without human input. This chapter covers automated data preprocessing, feature engineering, model selection, HPO, and model updating with concept drift adaptation. We examined IoT data analytics workloads and machine learning and deep learning models.

References

[1] S. Kumar, M. K. Chaube and S. Kumar, "Secure and Sustainable Framework for Cattle Recognition Using Wireless Multimedia Networks and Machine Learning Techniques," in IEEE Transactions on Sustainable Computing, vol. 7, no. 3, pp. 696-708, 1 July-Sept. 2022, doi: 10.1109/TSUSC.2021.3123496.

[2] P. Kumar et al., "PPSF: A Privacy-Preserving and Secure Framework Using Blockchain-Based Machine-Learning for IoT-Driven Smart Cities," in IEEE Transactions on Network Science and Engineering, vol. 8, no. 3, pp. 2326-2341, 1 July-Sept. 2021, doi: 10.1109/TNSE.2021.3089435.

[3] S. Yılmaz, E. Aydogan and S. Sen, "A Transfer Learning Approach for Securing Resource-Constrained IoT Devices," in IEEE Transactions on Information Forensics and Security, vol. 16, pp. 4405-4418, 2021, doi: 10.1109/TIFS.2021.3096029.

[4] N. Chawla, A. Singh, H. Kumar, M. Kar and S. Mukhopadhyay, "Securing IoT Devices Using Dynamic Power Management: Machine Learning Approach," in IEEE Internet of Things Journal, vol. 8, no. 22, pp. 16379-16394, 15 Nov.15, 2021, doi: 10.1109/JIOT.2020.3021594.

[5] D. H. Hagos, A. Yazidi, Ø. Kure and P. E. Engelstad, "A Machine-Learning-Based Tool for Passive OS Fingerprinting With TCP Variant as a Novel Feature," in IEEE Internet of Things Journal, vol. 8, no. 5, pp. 3534-3553, 1 March1, 2021, doi: 10.1109/JIOT.2020.3024293.

[6] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali and M. Guizani, "A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security," in IEEE Communications Surveys & Tutorials, vol. 22, no. 3, pp. 1646-1685, thirdquarter 2020, doi: 10.1109/COMST.2020.2988293.

[7] S. Zafar et al., "A Systematic Review of Bio-Cyber Interface Technologies and Security Issues for Internet of Bio-Nano Things," in IEEE Access, vol. 9, pp. 93529-93566, 2021, doi: 10.1109/ACCESS.2021.3093442.

[8] W. Y. B. Lim et al., "Hierarchical Incentive Mechanism Design for Federated Machine Learning in Mobile Networks," in IEEE Internet of Things Journal, vol. 7, no. 10, pp. 9575-9588, Oct. 2020, doi: 10.1109/JIOT.2020.2985694.

[9] M. U. Aftab et al., "A Hybrid Access Control Model With Dynamic COI for Secure Localization of Satellite and IoT-Based Vehicles," in IEEE Access, vol. 8, pp. 24196-24208, 2020, doi: 10.1109/ACCESS.2020.2969715.

[10] O. Mendsaikhan, H. Hasegawa, Y. Yamaguchi and H. Shimada, "Quantifying the Significance and Relevance of Cyber-Security Text Through Textual Similarity and Cyber-Security Knowledge Graph," in IEEE Access, vol. 8, pp. 177041-177052, 2020, doi: 10.1109/ACCESS.2020.3027321.

[11] N. M. Karie, N. M. Sahri, W. Yang, C. Valli and V. R. Kebande, "A Review of Security Standards and Frameworks for IoT-Based Smart Environments," in IEEE Access, vol. 9, pp. 121975-121995, 2021, doi: 10.1109/ACCESS.2021.3109886.

[12] M. Saharkhizan, A. Azmoodeh, A. Dehghantanha, K. -K. R. Choo and R. M. Parizi, "An Ensemble of Deep Recurrent Neural Networks for Detecting IoT Cyber Attacks Using Network Traffic," in IEEE Internet of Things Journal, vol. 7, no. 9, pp. 8852-8859, Sept. 2020, doi: 10.1109/JIOT.2020.2996425.

[13] A. Kovačević, N. Putnik and O. Tošković, "Factors Related to Cyber Security Behavior," in IEEE Access, vol. 8, pp. 125140-125148, 2020, doi: 10.1109/ACCESS.2020.3007867.

[14] B. Chatterjee, D. Das, S. Maity and S. Sen, "RF-PUF: Enhancing IoT Security Through Authentication of Wireless Nodes Using In-Situ Machine Learning," in IEEE Internet of Things Journal, vol. 6, no. 1, pp. 388-398, Feb. 2019, doi: 10.1109/JIOT.2018.2849324.

[15] M. H. Cintuglu, O. A. Mohammed, K. Akkaya and A. S. Uluagac, "A Survey on Smart Grid Cyber-Physical System Testbeds," in IEEE Communications Surveys & Tutorials, vol. 19, no. 1, pp. 446-464, Firstquarter 2017, doi: 10.1109/COMST.2016.2627399.

[16] R. Kozik, M. Choraś and W. Hołubowicz, "Packets tokenization methods for web layer cyber security," in Logic Journal of the IGPL, vol. 25, no. 1, pp. 103-113, Feb. 2017, doi: 10.1093/jigpal/jzw044.