Simplified review on cyber security threats detection in IoT environment using deep learning approach

Ayat T. Salim [1]
Information & Telecom Public Company,
Ministry of Communications-Iraq, Baghdad, Iraq.
ayat.tariq@coie-nahrain.edu.iq.
Ban Mohammed Khammas[2]
Department of Computer Networks Engineering, Collage of Information Engineering, Al-Nahrain University, Baghdad, Iraq,
bankhammas@coie-nahrain.edu.iq

Abstract:

Wearable technology, sensor networks, and home utilities are just a few of the businesses where the Internet of Things (IoT) is spreading quickly. With the development of the IoT, billions of gadgets are now connected to the internet and exchanging data. The proliferation of IoT devices has increased the number of IoT-based cyberattacks. In 2016 a massive denial of service (DDOS) cyber-attack was lunched utilizing infected internet of things devices a major website including Netflix and CNN was shutdown. Therefore, new ways for recognizing threats posed by hacked IoT nodes must be developed to overcome this concern. In that same context, ML and DL approaches are the best appropriate investigative control solution against IoT device-based intrusions. The point of the study is to offer a complete grasp of the IoT system-relevant technologies, standards, architecture, and the increasing dangers from corrupted IoT gadgets and an introduction to intrusion detection systems. Additionally, this research focuses on deep learning-based solutions for identifying IoT devices susceptible to cyber-attacks. The detection rate provided by deep learning algorithms shows promising results which reached 99% detection accuracy in some cases.

Keywords: IoT protocols; IoT security; machine learning; deep learning; intrusion detection system; cyber-attacks

1. Introduction

Humans in nature tend to exploit technologies to facilitate their life; why go to get groceries when your fridge can order them for you? Such a thing we call an IoT (or Internet of things) enabled device. It came into existence due to the development in communication and information technologies. In technical terms, we define IoT as a physical item (or entities) installed with circuits, algorithms, detectors, and internet access, which allows these entities to extract and share records. This action lets items be detected and controlled remotely, enabling prospects for direct integration between the physical and cyber worlds [1].

Using the data held by the devices helps us to develop expert machines and successfully manage IoT settings. Nevertheless, linking such regularly used physical objects to the web also poses worries about cyber-security risks [2]. Consequently, designing an intelligent security solution becomes necessary to protect IoT devices from external and internal threats.

1.1 Motivation

In the last few years, we have noticed a vast increment in IoT devices connected to the internet. In 2018 there were 21billion active devices, which is expected to reach 63 million by the end of 2025 [3]. These IoT-enabled devices are used to develop smart cities, education systems, intelligent health systems, e-shopping, e-banking, and more. New security challenges were introduced with this large-scale and ubiquitous system [4]. Furthermore, IoT devices are usually deployed in an unsupervised environment; this gives the attacker the possibility of physically accessing the device with malicious intent [5]. Also, we habitually connect the IoT devices using a wireless connection, increasing the possibility of an eavesdropping attack that compromises private data collected by these devices [6].

Besides these security problems, IoT devices cannot handle sophisticated security features due to their restrained power and compute capabilities. Given the heterogeneity of the IoT, new attack surfaces frequently emerge [7]. These attacks exploit existing vulnerabilities in the system, classified as missing authorization and authentication, cross-site scripting, insecure software or firmware, and lack of transport encryption and integrity validation [8].

Vendor-specific operating systems and communication protocols are common things among IoT devices. It is challenging to create a single security measure that can safeguard all types of IoT devices In addition, typical security measures like antivirus software might be difficult to implement on IoT devices due to their limited storage and processing capacity. Users are often unaware that malware and other forms of malicious software might compromise their IoT devices. This indicates that not enough safety measures have been taken. Malware attacks commonly take advantage of software or hardware vulnerabilities in IoT devices. Examples of this include the widespread Distributed Denial of service (DDoS) attacks on websites and networks caused by malware like the Mirai botnet, which took use of flaws in IoT devices to gain control of them [9]

Furthermore, malware assaults are hard to identify and react to because IoT devices are not standardized. This is because there is a wide variety of devices that might utilize a variety of techniques to report security problems and communicate with security systems. This may make it harder to identify malicious software and counterattacks. The existing lack of consistency in IoT devices may be remedied by developing a security solution that all IoT devices can adhere to. Because of this, IoT devices would be less vulnerable to malware attacks thanks to a consistent and powerful protection model [10].

1.2 literature survey

Haddad Pajouh et al., in 2018 [11] investigated the possibilities of using deep learning with Recurrent Neural Networks (RNN) to identify IoT malware. They specifically employ RNN to examine the execution operation codes of IoT applications (OpCodes). They employ a dataset of 281 malicious and 270 benign IoT applications to train their algorithms. The trained model is then tested using 100 fresh IoT malware samples —i.e., ones that had never been previously exposed to the model— across three distinct Long Short-Term Memory (LSTM) setups. The

setup with 2-layer neurons has the best Accuracy (98.18%), according to the 10-fold cross-validation investigation results, in identifying new malware samples this approaches gives really good detection accuracy on a small data set which considered to be a problem.

Azmoodeh et al., in 2018 [12] presented an approach using deep learning to identify malware on the Internet of Battlefield Things (IoBT) by analyzing the device's Operational Codes (OpCodes) sequence. To distinguish between harmful and benign applications, they convert OpCodes into a vector space and use a deep Eigenspace learning technique. Additionally, they show how resilient their suggested strategy is against junk code insertion attempts and malware detection. The method is successful in being accurate (99.68 %) here the authors reaches the desired 99% accuracy .

Seungho et al. in 2020 [13] used an opcode sequence-based convolutional recurrent neural network to identify malware. In terms of data analysis, an executable file is conceptualized as a sequence of machine codes. Initially, they covered the theoretical background of how opcode sequences may be utilized to identify malware. Next, they introduced a technique for extracting opcode sequences from executables and a deep learning-based malware-detection approach that takes the extracted sequences as input. The proposed model consists of a front-end opcode-level convolutional autoencoder that condenses a large opcode sequence into a shorter one and a back-end dynamic recurrent neural network classifier that uses the condensed sequence to make predictions. The suggested approach achieved a 96% detection rate for malware in experimental settings.

Radhakrishnan et al., in 2021 [14] Described the effectiveness of deep learning algorithms in detecting IoT malware. Particularly, their suggested solution makes use of RNN to examine IoT framework execution process codes. They utilize an IoT malware sample dataset of 271 benign and 282 malicious programs to test their methodology. Using the 104 untrained samples, they next evaluated the trained approach. The second configuration of the suggested model has a greater accuracy of 99.08 %.

2. IoT architecture

One of the most effective methods of addressing this issue is via the use of a few straightforward questions.

a) Is there any standardized architecture we can follow when designing an IoT environment? Researchers looked at all of the current IoT designs and concluded that answering this issue necessitates addressing a few additional sub-questions.

- What are the architecture layers or stacks?
- Does the meaning provided by all the layers to describe the IoT the same?
- Does the IoT nature be fully described by this architecture?

b) What do we consider essential? Are security and confidentiality among them?

We must examine the security and privacy aspects of the chosen architecture. This inquiry may also be composed of the following sub-questions.

- What problems are covered by this IoT architecture, and what are the technologies? ·
- Does this architecture require implementing security and privacy aspects?

c) Is this architecture flexible enough to support the connection of billions of heterogeneous devices through the internet?

Various authors and organizations proposed architectural models for the IoT environment, but none has been converted into a formally recognized reference model [15]. This part will go through the overall IoT architecture and how they answer the problems raised above:

- **A 3-layer architecture.** This architectural model, which was proposed in 2010, is the most prevalent and fundamental architectural model. It was made up of three layers: the "perception layer" at the bottom, the "application layer" at the top, and the "network layer" in the middle. As depicted in Figure 1, the perception layer is also known as the "device layer" because it houses all gadgets used to identify items and collect data (e.g., radiofrequency identifier, 2D barcode, etc.). Its network layer is the beating heart of the Internet of Things. It entails giving each device a unique address and securely transferring the collected data to the application layer using Wi-Fi, Bluetooth, and ZigBee protocols. Finally, the application layer oversees any IoT applications built or created utilizing the network layer's technologies [16].

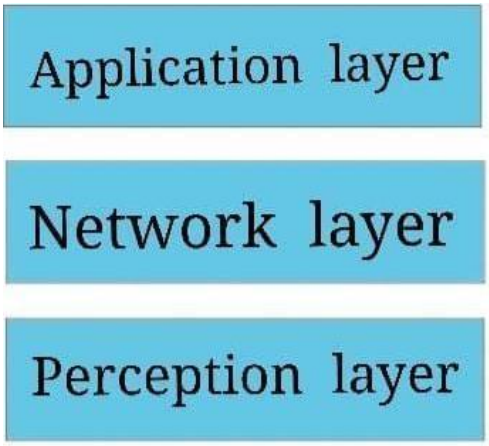


Figure 1 3-layered IoT architecture[21]

- **The International Telecommunication Union (ITU) suggested architecture.** This proposed model comprises four layers, with the most important protection and administration method. As depicted in Figure 2, according to the naming conventions, the layers are as follows: device, network, application support, service, and the application layer. [17].

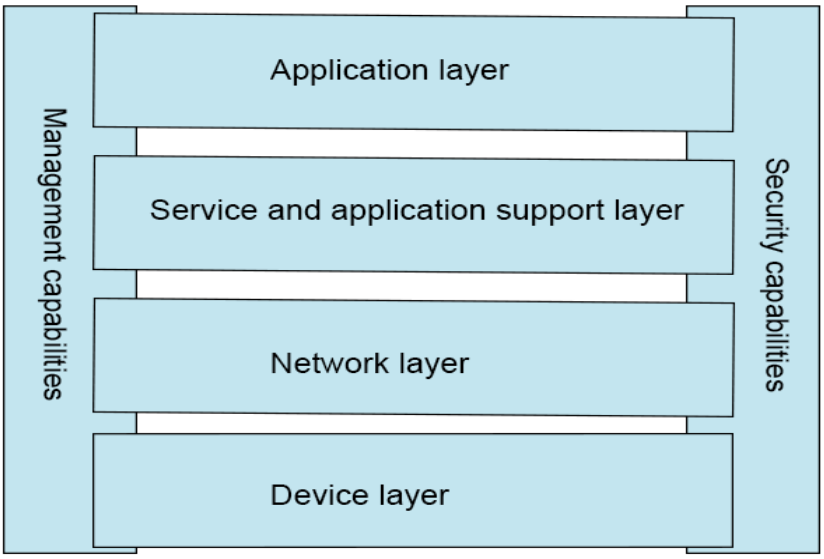


Figure 2 "ITU-T" architectural paradigm [17]

●**The European Commission approved reference architecture for the Internet of Things (IoT) (FP7)**. The European Commission is a co-initiator of the seventh framework project, which aims to develop a generic architectural reference model (ARM) that will meet the demands of business and research. The IoT- A project suggested by Martin Bauer, which presented a high-level architectural approach for creating IoT systems, is supported by this initiative. The architecture explains the organizing and designing of IoT industrial operations, IoT solutions, information, and functional views in an abstract sense. Figure 3 depicts the practical perspective [18, 19].

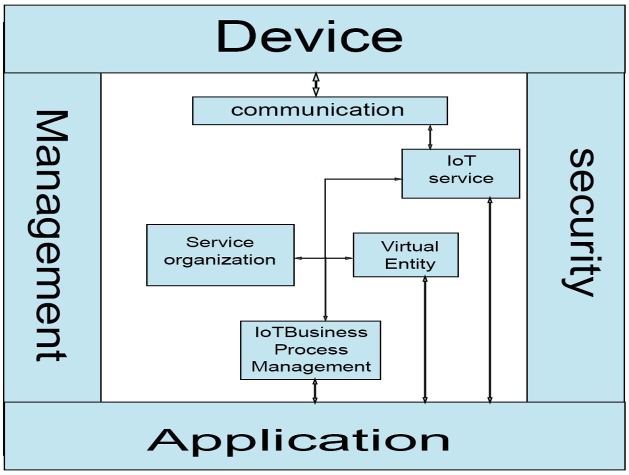


Figure 3 "IoT-A" practical perspective [18, 22]

- **Cisco proposed the Internet of Things paradigm.**

7-layered IoT reference architecture was approved by cisco. The model is represented in figure 4. These simple paradigms comprise fundamental building components with security across all tiers [20].

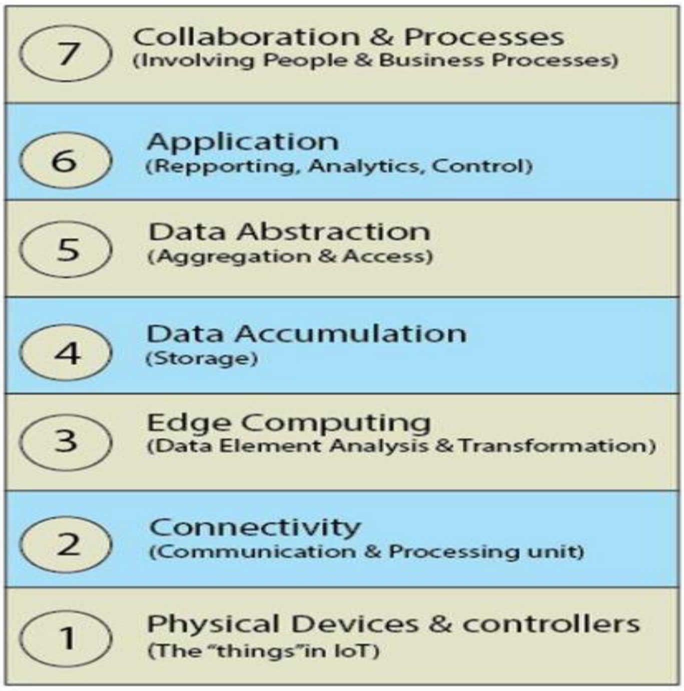

7 Collaboration & Processes
(Involving People & Business Processes)
6 Application
(Repporting, Analytics, Control)
5 Data Abstraction
(Aggregation & Access)
4 Data Accumulation
(Storage)
3 Edge Computing
(Data Element Analysis & Transformation)
2 Connectivity
(Communication & Processing unit)
1 Physical Devices & controllers
(The "things"in IoT)

Figure 4 cisco internet of things reference model [23]

3. IoT protocols

To communicate information across devices and backend processors, IoT-based devices use a variety of short and long-range communication protocols. Some of these protocols and standards are direct descendants of the TCP/IP architecture, while others are created explicitly for Internet of Things (IoT) systems. Table 1 summarizes the most common communication protocols utilized in IoT systems [24].

Technology	Operating frequency	Data rate	Coverage	Latency	Power usage	Use cases
ZigBee	2.4 GHz, 868 MHz, 915000 KHz	250 Kb/s	0.05–0.1 Km	16 ms	Weak	e-healthcare, intelligent metering
Bluetooth	2400 MHz	250 Kb/s	0.01 Km	100 ms	Weak	e-healthcare
Wi-Fi	2400 MHz ,5000 MHz, 802.11n	54000 Kb/s, 6750 Mb/s	140 m 100 m	46 ms	Medium	metering, waste disposal automation, energy conservation, infotainment, and automation
IEEE 802.11p	5850–5925 MHz	6 Mb/s	1 Km		Weak	Motor communication, V2V/ V2I, infotainment
DASH7	433, 868, 915 MHz	55.5 kb/s, 200 kb/s	1 Km	15 ms	Weak	ITS, automation
DSRC/WAVE	5800, 5900 MHz	6 Mb/s	1 Km	200 µs	Weak	ITS (V2V/V2I)
6LoWPAN	2.4 GHz, 868, 915 MHz	250 kb/s	100 m		Weak	ITS, intelligent metering, logistics
LoRaWAN	433, 868, 780, 915 MHz	50 kb/s	2–5 km		Weak	ITS, intelligent metering, waste management

Technology	Operating frequency	Data rate	Coverage	Latency	Power usage	Use cases
GSM/GPRS	850, 900, 1800, 1900 MHz	80–384 Kb/s	5–30 km	1.5–3 s	Heavy	ITS, intelligent metering, m-health, energy conservation, logistics, infotainment
3G	850 MHz	3 Mb/s	5–30 km	100 ms	Heavy	ITS, intelligent metering, energy conservation, m-health, logistics, infotainment
LTE/LTE-Advanced	700, 750, 800, 1900, 2500 MHz	1 Gb/s, 500 Mb/s	5–30 km	5 ms	Heavy	ITS, intelligent metering, mobile health, logistics, infotainment

Table 1 list of crucial communication protocol for IoT systems [24].’

4. Security issues in IoT base systems

A widely used system like the Internet of Things is vulnerable to many security risks and assaults. As previously stated, this is due to the system's inherent nature. If we look at it from a distance, we can see that it comprises various devices from various manufacturers, platforms, and communications methods and Protocols. Before this, the system was made up of "things" that humans did not intend to link to the internet when they initially developed them. Furthermore, because of the mobility of people and equipment, IoT systems lack specific boundaries; finally, IoT systems have limited power, making modern security procedures and tools challenging to implement [25].

Because authentication between nodes adds transmission cost to an IoT network with hundreds or thousands of nodes, trust between these nodes is presumed to be given. We build a vulnerability that malevolent attackers may exploit by utilizing a rogue node that can easily control the network by making this assumption. [26]. These nodes, on the other hand, communicate via various network protocols like "Wi-Fi", "Bluetooth", and "ZigBee".

An IoT getaway or border router is needed to link these devices to the internet. Figure 5 demonstrates how the data is delivered to the server or database.

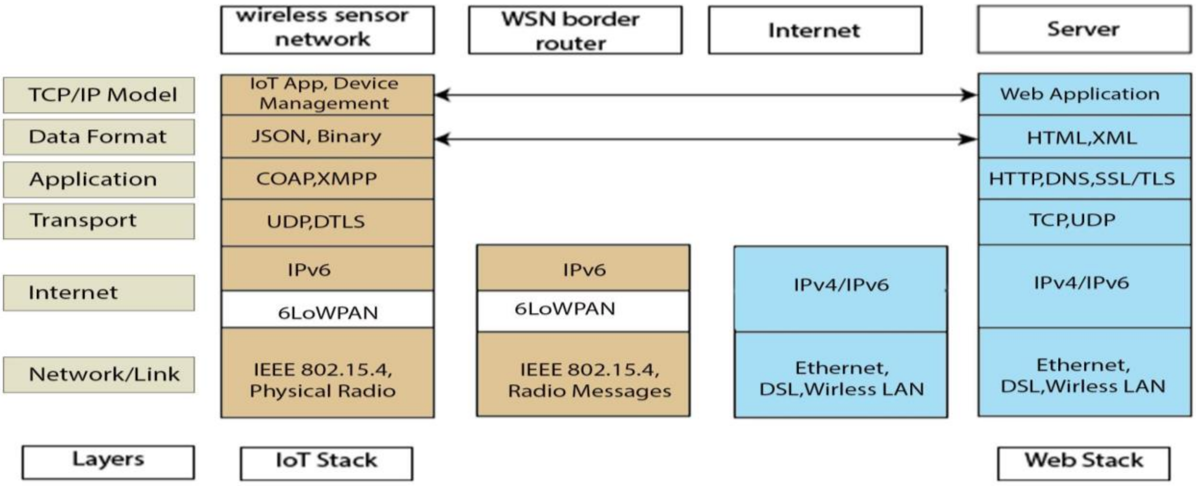


Figure 5 transmitted data IoT environment [11]

If we look at figure 5, we see that ipv6 is a firm root for IoT; it works as an enabler because the ipv4 cannot handle the massive size of the IoT system. That led to the security suggestions and consideration of ipv6 as the basis for IoT security [27]. a nutshell, the Internet of Things (IoT) sits at the crossroads of the physical and cyber realms, resulting in a larger attack surface. The danger dimension exploited by these assaults may be classified as access method and other networked IoT systems related to sensors and network services [8]. The subsections that follow provide an overview of these aspects.

4.1 Access method

Using the exact access mechanism to engage with an IoT device is typical (desktop, mobile phone, Web App). Users will engage with the IoT using a mobile application loaded on their smartphone while operating in a bright home environment. Because of the quick depreciation of cellphones, attackers may masquerade malware as innocent programs with public storage that may be downloaded without identifying the infection [28]. Furthermore, even if we successfully ensure the installation of malware-free programs, what about the growing danger of platform hacking? Studies have demonstrated that platform vulnerabilities, such as Android vulnerabilities, may be used to compromise the system. As a result, the device's data and information will be exposed, with the chance of virus infiltration. Eavesdropping, Denial of Service attacks, location tracking, and other assaults are possible because of the user interface platform [29, 30].

4.2 Connection of multiple IoT systems

The Internet of Things (IoT) system is meant to need little or no human input. The IoT device makes decisions about communication and interactions with some other IoT devices, such as sensing devices in smart automobiles and intelligent cities, on its own. This kind of contact allows for both sovereign and collaborative functioning. IoT systems may interact with one another while still performing separate duties and providing services. For example, the authors in [31] explain a case in which a thermometer identified an increased temperature in an indoor environment. An intelligent plug recognized that the air conditioning (HVAC) was turned off, causing the window to be opened due to these two sensed values. Attackers may access the window opening actuator by altering the temperature via its interface, which would compromise the actuator internally [31]; an innovative example focuses our thoughts on the notion that the feeble link may cause problems in other areas.

This illustration emphasizes that a chine's most fragile connection may jeopardize other portions.

Vulnerabilities grow as the number of networked devices in IoT systems grows, as does the effect of any assault, where one harmed device might affect billions of others. Such a circumstance might impact any external network or system. Research published in [32] demonstrates the influence of an experimental intelligent Hue Philips bulb carried by agents. Despite the solid cryptographic authentication procedures established against agents' fraudulent firmware upgrades, the attack was effective since it compromised all of the lights in the network. A similar assault may take control of light provisioning for a whole city or use the devices to launch a distributed denial-of-service assault on an external system [32].

An Internet of Things (IoT) system comprises multiple networked devices linked through a wired or wireless link. Due to the numerous vulnerabilities of the sensors and actuators, any network with a significant number of devices will have a poor security rating WSNs have no restrictions when exchanging data with other parties. When coupled with conventional network services, the traditional network's security suffers.[33]

4.3 Protocol level attacks

When comparing the IoT system communication protocols to the traditional internet protocols, we found that the IoT protocol is lightly weighted to address the data rate, computing power, and energy constraints. A comprehensive description of IoT protocol-based attacks is provided by [34]. Moreover, we will also dig into it in the following sections.

4.4 Radio frequency identifier (RFID)

This technology automatically transfers information between tags and readers using radio waves over unprotected wireless channels; the information is available to unauthorized readers. RFID systems encounter many security threats compared to a traditional wireless systems [35].

Some of the techniques used in attacking RFIDs are listed below

- **Replay attacks.** The attacker stealthily obtains knowledge regarding the IoT device by replaying eavesdropped data to prevent being discovered.
- **Attacks through the relay.** An unauthorized device intercepts the data exchange between the tag and the reader. This device is used to modify and forward the information to other systems.
- **Tag disable.** By forcibly eliminating the tag or manipulating the tag memory with a killing instruction, this assault is conducted to prevent the tag from interacting with the reader.
- **Tag modification.** This attack enables the attacker to compromise the confidentiality of valuable data stored in the tag memory.
- **Cloning tags.** After getting the tag information, the attacker will pretend to be the tag.
- **Snooping.** The attacker will provide the unauthorized reader to interact with the tag.

4.5 ZigBee communication protocol.

This protocol is among the most widely used internet of things communications protocols. For its reasonable cost, minimal power consumption, and scalability. Security issues were taken into account throughout the design process of this protocol. However, trade-offs had to be made to make the device economical. Several security Approaches were unable to be applied, resulting in security flaws. The following are some of the most severe security risks [34].

- **Packet Sniffing.** Because ZigBee does not use encryption as the primary countermeasure against sniffing attacks, it becomes susceptible to such assaults.

● **Replay attack.** This attack is based on opposing data, sniffing raw data, and then resending it as regular traffic.

● **Eavesdropping.** A MITM attack may be used to spy on the ZigBee connection and divert data.

4.6 Wireless fidelity.

When developers design the IoT system, they mainly rely on enabling technologies. One of which is Wi-Fi. Here are some of the common attacks that may face this technology.

● **Injection**. To accomplish message injection in this attack, software must be installed. A hostile actor may insert forged data, change the packet's preamble or tail, and change any data packet field. After implanting the data into the transmission stream, the attacker has complete control over the transmission process [36].

● **Eavesdropping**. Wi-Fi works in an open environment, making it vulnerable to eavesdropping attacks. Even if the data is encrypted using some related tools, the attacker can access the data [36].

● **Session Hijacking.** The attacker used specific tools to disconnect the station from the access point in the initial stage of the assault. In the second step, the attacker will connect to the access point by impersonating an actual station. The attacker will be able to hijack the communication session and manipulate the sending and receiving of messages [36].

● **Forged AP.** The packet header in the wireless transmission is delivered in plain text and contains the AP's MAC address. Using any sniffing tool, the attacker may get access to it and mimic a real access point by altering its MAC address [36].

4.7 Bluetooth.

This section will go through Bluetooth low energy (BLE), a lightweight version of Bluetooth targeted for low-power applications. The following is a list of common assaults.

● **Man-in-the-Middle (MITM) Attacks.** In this attack, an unauthorized party intercepts the connection between the transmitter and the receiver. It is vital to prevent such an assault by ensuring that the end device the connection began is infected with the desired device [37].

● **PIN breaking attack.** This attack unfolds during the pairing step of the authentication procedure. This attack uses a frequency sniffer tool to collect the targeting device’s RAND ("random number") and Bluetooth device address. After the comprises-force approach is used to determine all potential PIN combinations. The pin gets broken after a few tries [37].

- **Worm attack.** The attacker will send compromised data (malware or trojan) to the victim; running these files, the malware will be activated [38].
- **Bluebugging.** This attack exploits security flaws in outdated device firmware, providing the attacker access to phone call records and the possibility to connect to the internet without the user's knowledge [39].
- **Bluesnarfing.** Privilege is given to the attacker to access and retrieve information and redirect incoming calls [39].

4.8 RPL Protocol

This standard was designed to facilitate point-to-point and point-to-multipoint communication. RPL attacks may be divided into three kinds based on the vulnerability they attempt to exploit. These are the categories [40]:

- Topology-based **attacks.** Network topology has also been divided into two by learning from the previous sample's sub-categories:

1. **Sub-optimization:** attacking the performance of the network by diminishing its optimal path.
2. **Isolating attacks:** the attacker aims to isolate the RPL nodes by preventing communication with the core node.

- **Resource base attack.** An attacker aims to drain the network resources such as storage, energy, and computation power as the availability of the network is compromised.
- **Traffic base attacks.** Here the concerns are towards the traffic, and we can classify it into two categories:

1. **Passive attacks.** The attacker executes an eavesdropping activity, such as analyzing the network traffic.
2. **Deception attacks.** The attacker imitates the activities of an authorized node. Such an attack is usually used as the first step in launching other attacks.

5. intrusion detection system (IDS)

Since 1970, when intrusion detection systems were first used in computer science, we have been urged to establish a technology that identifies potential incursions or threats early. The IDS comprises three main modules: (1) an input module that accumulates all of the data, most of which will include some indications of an attack, and (2) a recognition module. Most analytical operations will occur here, including detecting attack patterns and the (3) reporting module, where the attack reporting method will be developed. In the input module, data from all components of the IoT system is sent to the Intrusion detection systems, which are then evaluated to define the IoT system's usual behavior, allowing malicious activity to be identified. The recognition unit may be built using a variety of methodologies and models; however, in recent years, ML\DL algorithms have acquired a lot of traction owing to their capacity to identify benign and harmful behaviors in an IoT environment based on interactions between devices and IoT systems. Furthermore, by learning from previous samples, ML/DL algorithms may anticipate new attack patterns by learning from previous samples, making them a formidable defensive line against zero-day assaults [41]. Figure 6 depicts an essential representation of the (IDS).

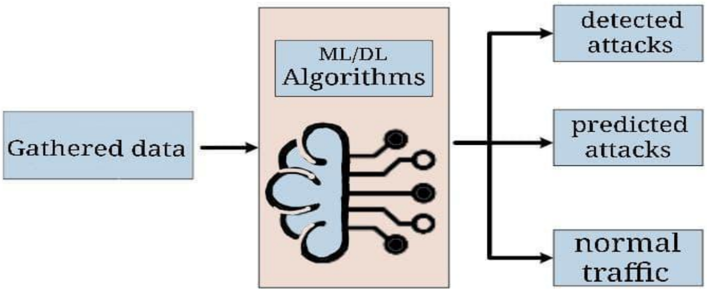


Figure 6 simple illustration of ML/DL-based IDS[42]

5.1 Detection Methods of IDS

Intrusion detection systems rely on a few carefully designed techniques that help them make an accurate assumption about any malicious activities. The following subsection will list and dive into the most common detection methods.

5.1.1 signature-Based method

Most assaults have a pattern that they follow when launched, and our detection approach uses this previous information to identify the assault. The signature base solution compares system input or activities to a database or repository of attack patterns. Any time a match pattern is discovered, a threat warning is triggered. Although this strategy is effective against old, well-known assaults, it is substantially less effective against new assaults whose pattern has yet to be established.[43].

5.1.2 Anomaly-Based method

The anomaly-based method depends on defining a set of rules that specifies the expected behavior, which is the opposite of how the signature-based method works. Instead of having a predefined pattern and comparing it against the system traffic, the traffic is analyzed to see if the rules have applied any drift from these rules is considered abnormal behavior. The threat alert is raised without specifying the type of attack. This method works best in detecting zero-day attacks. Still, it has some drawbacks. One of them is the difficulty in constructing the rules that specify the expected behavior formed using a specified algorithm, leading to a few faulty assumptions [44]

5.1.3 specification-Based method.

The working model for the previous methods is evident. The expected behavior of the system is specified through such a short comparison. These rules that we compare against in the anomaly base method are learned automatically by using algorithms, whereas, in specification-Base, these rules are specified manually by human experts. This allows for a lower number of faulty assumptions compared to the anomaly-based method [45].

5.1.4 Hybrid-Based method

This method combines the previously mentioned methods to overcome limitations and provide an optimized methodology for detecting existing and new attacks [46].

6. Deep learning (DL) techniques for IDSs

When it comes to massive datasets, DL shows significantly promising results compared to ML, making it a perfect fit for ubiquitous and large-scale environments that produce a vast amount and variety of data, such as the IoT environment. Furthermore, the DL algorithms can debrief complex representations from the data. DL automatically enables IoT devices to interact with their application using a deep linking protocol [47].

DL algorithms may be categorized as a subset of ML techniques that use various non-linear levels of learning to recognize which sets and their capability to accommodate hierarchical attribute vectors in complex deep networks. After the requisite preprocessing, these feature sets are employed for pattern recognition [48]. Figure 7 shows the classification of DL methods and used algorithms.

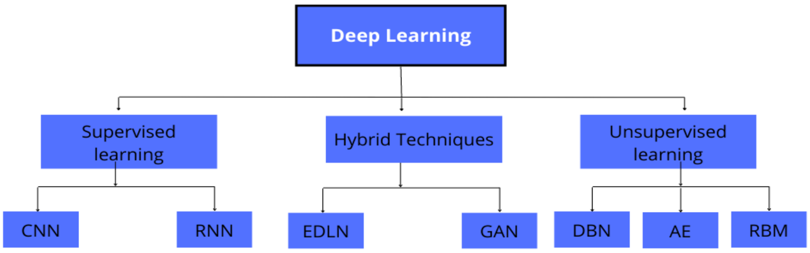


Figure 7 classification of DL technologies for IoT[49]

In the following subsections, various primary DL-base techniques used to implement IDSs are discussed

6.1 RNNs

The recurrent neural network is a backpropagation-based method commonly used in research fields dealing with sequential data including audio, video, and text [50]. The output of this algorithm tends to be adaptive to past inputs, which is contrary to any regular feed-forward network. The RNN algorithm comprises a transit layer to seize the sequential information and then recognize complex patterns using the remote unit of the recurrent elements. Specific adjustments are made to these hidden units to correlate with the data held by the neural network, leading to constant updates and explicit reflection of the current state of the neural network[51].

As shown in figure 8, the recurrent neural network consists of the input, hidden, and output layers; the hidden layer is the core of this algorithm where all the work occurs. Information flows from the input layer and then to the hidden layer. In the hidden layer, we see a cyclic neuron connection which enables us to Memorize the previous information and apply it to the current output. RNNs are becoming increasingly crucial in IoT security applications, particularly in detecting network intrusion. RNNs are useful in IoT security applications, particularly network intrusion detection since IoT settings generate enormous volumes of sequential data, such as network traffic flows LSTM network design, a recurrent neural network, has also been used to create IDS.

The most remarkable characteristic of RNNs based on LSTM is their capacity to retain information or neuron status for later use on the net. As a result, they're ideal for examining periodic data that changes over time. As a result, for identifying abnormalities in time-series sequence data, Lstm is used. While RNNs have shown promise in anticipating time series data, applying these predictions to identify unusual traffic remains tricky [52].

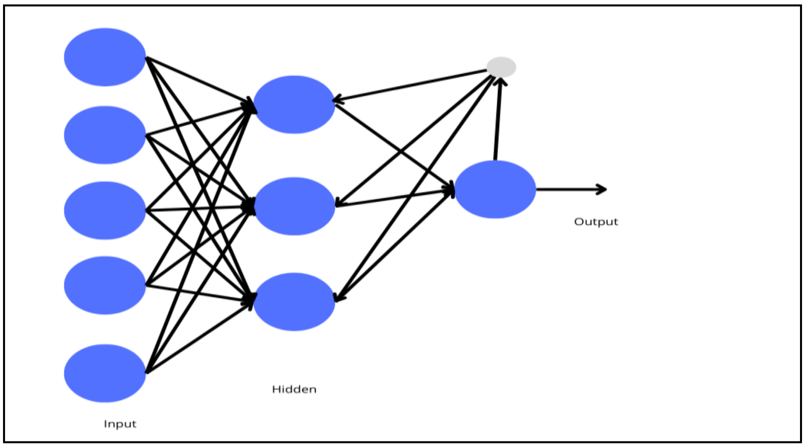


Figure 8 is an illustration of the RNN algorithm[53]

6.2 CNN

Convolutional neural networks (CNNs) are a kind of discriminative deep learning approach that uses equivariant formulation, sparse interactivity, and parameter exchange to decrease the amount of data inputs needed for the standard neural network [54]. Consequently, CNN is more scalable, and training time is reduced. Figure 9 illustrates the general structure of CNNs, which contains two components: custom pattern extractors and a categorizer. Each layer of the network in the pattern extraction layers obtains its output from the layer before input and sends it on to the next layer as output. The three types of layers that make up the CNN architecture are "convolution," "max-pooling," and "classification." In the network's bottom and intermediate levels, convolution layers and pooling layers are the two types of layers. An even number of layers are employed for convolutions, whereas an odd number of layers are used For max-pooling processes. At each of the convolution layer's nodes, convolution operations on the input nodes capture information from the input data [55].

The most advanced features are created using features transmitted from lower-level layers to the top-level layer. Because higher-level characteristics are created using features relayed from bottom layers. The kernel structure for both the convolutional and max-pooling layers reduces the dimensionality of features as they propagate to the highest layer or level.

Nevertheless, the number of feature maps is typically expanded to represent more delicate characteristics of the input photographs to guarantee classification accuracy. The classification layer receives the output of the CNN's final layer, which is transmitted into a fully connected network. The gathered features are utilized as inputs in the classification layer, with the dimension of the last neural network's weight matrix considered [56].

On the other hand, the entirely interconnected layers are costly in terms of network or learning assets. Consequently, using CNN to secure IoT devices with minimal resources is challenging. A distributed architecture partly solves this difficulty by training and implementing a reduced version of a Deep neural network with just a subset of relevant output classes. At the same time, the cloud's enormous computational power is used to complete the algorithm's training [57].

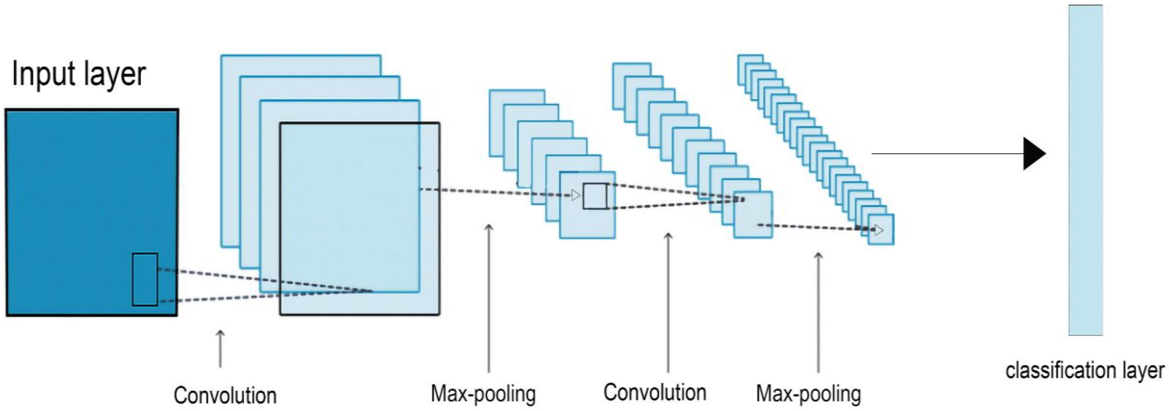


Figure 9 is an illustration of the CNN algorithm [58].

6.3 DAEs

Deep autoencoders is an unescorted method that tries to provide the same results as their input. The use of a decoder technique, including hidden units that specify a code for input representations. This encoding function is the second function in an autoencoder neural network, and it converts the obtained input into code. During training, mistakes in reconstruction must be underrated. [58]. Feature extraction from datasets is one of AE's applications. These, on the other hand, are hampered by the need for a lot of processing power. Deep AEs are more accurate than SVM and KNN in detecting network-based malware[59].

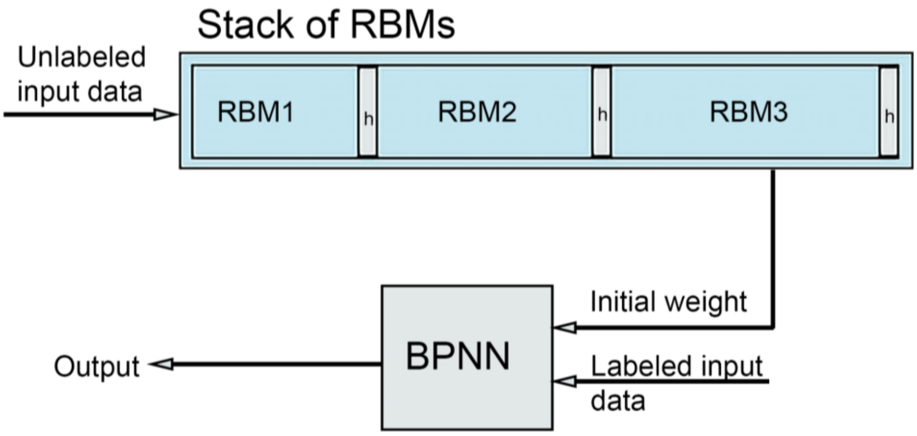
6.4 Restricted Boltzmann Machine (RBM)

Deep Belief Networks, is a practical method for creating deep networks in 2006, igniting a deep learning research boom. The DBN pre-trains the network's weights using the Restricted Boltzmann Machine (RBM), then fine-tunes the weights using the gradient descent approach. The RBM and the Auto-Encoder (AE) are fundamental deep learning-building pieces. The RBM, in contrast to the AE, is an energy-based model, and it may also be thought of as a form of Markov Random Field (MRF). The generic RBM is a valuable tool for describing the dependence structure between random variables. The RBM has recently piqued the attention of the machine learning Artificial intelligence communities. The RBM was designed with categorization and representational learning in mind.

In any layer of an RBM, no two nodes are connected. There are two layers in an RBM: visible and concealed layers. The visual layer contains known input parameters, while the buried layer contains unknown potential variables. Characteristics acquired from such a statistic are handed to the next layer using a hierarchical method. RBMs have been used in research for network/IoT intrusion detection systems. RBMs are challenging to implement on low-watt IoT devices since they demand a lot of computing power. Furthermore, Single RBM is incapable of representing features. On the other hand, this limitation may be overcome by constructing a Deep Belief Network by layering two or even more RBMs (DBN)[60, 61].

6.5 Deep Belief Network (DBN)

As illustrated in figure 10, we have a DNN with several restricted Boltzmann machines and a back-propagation neural network. With an RBM stack of three RBMs connected from the lowest to the highest levels. Like with numerous other deep neural networks, the main principle behind DBN is to train feedforward neural networks unsupervised using unlabeled data before fine-tuning them using labeled data. The contrastive divergence is used to prepare the first RBM in the pre-training phase. The CD approach facilitates the computation of the log-probability gradient by matching expectations with a restricted number of Gibbs sampling rounds that are started with visual units set to training data. As illustrated in figure 10, the learned states of the hidden units in the first RBM are utilized as input data for the visible units in the second RBM. All RBMs' weights are taught the same way, layer by layer, until the final RBM. Lower-layer RBMs in the stack of RBMs learn lower-level features of the training data, whereas higher-layer RBMs learn higher-level features. The top-most layer of RBM weights is utilized as the start weights of the FFNN when the unsupervised pre-training of RBMs is finished. Using labeled training data and learning techniques such as the back-propagation technique, the FFNN is fine-tuned or trained[62]. Although research in [63] examined malicious attack detection using DBNs and found that they performed better than ML methods.



Figurer 10 DBN structure [62].

6.6 A handful of DL Networks

Multiple DL algorithms may be employed in parallel to yield better results than each component DL algorithm by grouping them into an ensemble. Discriminative, generative, or hybrid DL algorithms are all acceptable to be used in this strategy. This approach performs better in uncertain contexts with many characteristics and is best suited for tackling complicated situations. Classifiers from various genres are used in a heterogeneous implementation, while classifiers from the same genre are used in a homogeneous performance. Both compositions increase efficiency and provide precise outcomes [64]. Further research and analysis using this strategy for IoT security are needed to assess the feasibility of enhancing the IoT security system's efficiency and reliability [8].

7. conclusion

usage of IoT devices has increased in all sectors of life over the past decade. At the same time, IoT security flaws put users' privacy and security at risk. As a result, more robust security solutions for the Internet of Things are required. This study reviews the most common deep learning approaches in detecting IoT malware the study shows that deep learning got a great potentials in detecting zero day attacks due to the deep learning algorithm structure and its design.

Ethical Statements

Manuscript title: Simplified review on cyber security threats detection in IoT environment using deep learning approach.

The authors whose names are listed immediately below certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers’ bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements).

All Authors were aware of about the paper content and approved its submission. This manuscript has not been published and is not a consideration for publication elsewhere.

Ethics approval and consent to participate

(Not applicable)

Consent for publication

(Not applicable)

Availability of data and materials

(Not applicable)

Competing interests

The authors have declared that no competing interests exist

Funding

No fund has been received for this research study.

Conflict of interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

Acknowledgment

The authors would like to thank the Department of Computer Networks Engineering, Collage of Information Engineering, Al-Nahrain University, Baghdad, Iraq, for the logistic support this work.

References

1. Stoyanova M, Nikoloudakis Y, Panagiotakis S, Pallis E, Markakis EK. A survey on the internet of things (IoT) forensics: challenges, approaches, and open issues. IEEE Communications Surveys & Tutorials. 2020;22(2):1191-221.
2. Wheelus C, Zhu X. IoT Network Security: Threats, Risks, and a Data-Driven Defense Framework. IoT. 2020;1(2):259-85.
3. H. F. Atlam and G. B. Wills, "IoT security, privacy, safety and ethics," in Digital twin technologies and smart cities, Springer, 2020, pp. 123–149.
4. Zhong M, Zhou Y, Chen G. Sequential model-based intrusion detection system for IoT servers using deep learning methods. Sensors. 2021;21(4):1113.
5. Moustafa N. Designing an online and reliable statistical anomaly detection framework for dealing with large high-speed network traffic: University of New South Wales, Canberra, Australia; 2017.
6. Kolias C, Kambourakis G, Stavrou A, Voas J. DDoS in the IoT: Mirai and other botnets. Computer. 2017;50(7):80-4.

7. Al-Garadi MA, Mohamed A, Al-Ali AK, Du X, Ali I, Guizani M. A survey of machine and deep learning methods for internet of things (IoT) security. IEEE Communications Surveys & Tutorials. 2020;22(3):1646-85.
8. NICULA S, ZOTA R-D. Technical and Economic Evaluation of IOT Attacks and their Corresponding Vulnerabilities. Informatica Economica. 2021;25(1).
9. S. Kirmani, A. Mazid, I. A. Khan, and M. Abid, “A Survey on IoT-Enabled Smart Grids: Technologies, Architectures, Applications, and Challenges,” Sustainability, vol. 15, no. 1, p. 717, 2023.
10. D. Vasan, M. Alazab, S. Venkatraman, J. Akram, and Z. Qin, “MTHAEL: Cross-architecture IoT malware detection based on neural network advanced ensemble learning,” IEEE Transactions on Computers, vol. 69, no. 11, pp. 1654–1667, 2020
11. H. HaddadPajouh, A. Dehghantanha, R. Khayami, and K.-K. R. Choo, “A deep recurrent neural network based approach for internet of things malware threat hunting,” Future Generation Computer Systems, vol. 85, pp. 88–96, 2018
12. A. Azmoodeh, A. Dehghantanha, and K.-K. R. Choo, “Robust malware detection for internet of (battlefield) things devices using deep eigenspace learning,” IEEE transactions on sustainable computing, vol. 4, no. 1, pp. 88–95, 2018
13. S. Jeon and J. Moon, “Malware-detection method with a convolutional recurrent neural network using opcode sequences,” Inf Sci (N Y), vol. 535, pp. 1–15, 2020
14. 17 . G. Radhakrishnan, K. Srinivasan, S. Maheswaran, K. Mohanasundaram, D. Palanikkumar, and A. Vidyarthi, “A deep-RNN and meta-heuristic feature selection approach for IoT malware detection,” Mater Today Proc, 2021.
15. Sethi P, Sarangi SR. Internet of things: architectures, protocols, and applications. Journal of Electrical and Computer Engineering. 2017;2017.
16. Jamali MAJ, Bahrami B, Heidari A, Allahverdizadeh P, Norouzi F. IoT Architecture. Towards the Internet of Things. 2020:9-31.
17. Kafle VP, Fukushima Y, Harai H. Internet of things standardization in ITU and future networking technologies. IEEE Communications Magazine. 2016;54(9):43-9.
18. Weyrich M, Ebert C. Reference architectures for the internet of things. IEEE Software. 2015;33(1):112-6.

19. Soares N, Monteiro P, Duarte FJ, Machado RJ. Extending the scope of reference models for smart factories. Procedia Computer Science. 2021;180:102-11.
20. Hadzovic S, Mrdovic S, Radonjic M. Identification of IoT Actors. Sensors. 2021;21(6):2093.
21. Wu M, Lu T-J, Ling F-Y, Sun J, Du H-Y, editors. Research on the architecture of Internet of Things. 2010 3rd international conference on advanced computer theory and engineering (ICACTE); 2010: IEEE.
22. Torkaman A, Seyyedi M. Analyzing IoT reference architecture models. International Journal of Computer Science and Software Engineering. 2016;5(8):154.
23. Λαμτζίδης O. An IoT edge-as-a-service (Eaas) distributed architecture & reference implementation 2020.
24. Silva BN, Khan M, Han K. Internet of things: A comprehensive review of enabling technologies, architecture, and challenges. IETE Technical review. 2018;35(2):205-20.
25. Tawalbeh La, Muheidat F, Tawalbeh M, Quwaider M. IoT Privacy and security: Challenges and solutions. Applied Sciences. 2020;10(12):4102.
26. Razzaq MA, Gill SH, Qureshi MA, Ullah S. Security Issues in the Internet of Things (IoT): A Comprehensive Study. 2020.
27. Gont F, editor Results of a Security assessment of the internet protocol version 6 (ipv6).
28. Selvaganapathy S, Sadasivam S, Ravi V. A Review on Android Malware: Attacks, Countermeasures and Challenges Ahead. Journal of Cyber Security and Mobility. 2021:177–230-177–230.
29. Alaba FA, Othman M, Hashem IAT, Alotaibi F. Internet of Things security: A survey. Journal of Network and Computer Applications. 2017;88:10-28.
30. Steinhubl SR, Muse ED, Topol EJ. The emerging field of mobile health.
31. Zhou W, Jia Y, Peng A, Zhang Y, Liu P. The effect of IoT new features on security and privacy: New threats, existing solutions, and challenges yet to be solved. IEEE Internet of Things Journal. 2019;6(2):1606-16.
32. Ronen E, Shamir A, Weingarten A-O, O’Flynn C. IoT Goes Nuclear: Creating a Zigbee Chain Reaction. IEEE Security & Privacy. 2018;16(1):54-62.

33. Shouran Z, Ashari A, Priyambodo T. Internet of things (IoT) of smart home: privacy and security. International Journal of Computer Applications. 2019;182(39):3-8.
34. Abdul-Ghani HA, Konstantas D, Mahyoub M. A comprehensive IoT attacks survey based on a building-blocked reference model. International Journal of Advanced Computer Science and Applications. 2018;9(3):355-73.
35. Aghili SF, Ashouri-Talouki M, Mala H. DoS, impersonation and de-synchronization attacks against an ultra-lightweight RFID mutual authentication protocol for IoT. The Journal of Supercomputing. 2018;74(1):509-25.
36. Wang S, Li B, Yang M, Yan Z, editors. Intrusion detection for WiFi network: A deep learning approach. International Wireless Internet Conference; 2018: Springer.
37. Mawgoud AA, Taha MHN, Khalifa NEM. Security threats of social internet of things in the higher education environment. Toward Social Internet of Things (SIoT): Enabling Technologies, Architectures and Applications: Springer; 2020. p. 151-71.
38. Lonzetta AM, Cope P, Campbell J, Mohd BJ, Hayajneh T. Security vulnerabilities in Bluetooth technology as used in IoT. Journal of Sensor and Actuator Networks. 2018;7(3):28.
39. Hamby MF. Defensive Strategies for the Internet of Things Sensors Using Bluetooth Low Energy: Capitol Technology University; 2020.
40. Almusaylim ZA, Alhumam A, Jhanjhi N. Proposing a secure RPL based internet of things routing protocol: a review. Ad Hoc Networks. 2020;101:102096.
41. Al-Hadhrami Y, Hussain FK. ReReal-timeataset generation framework for intrusion detection systems in IoT. Future Generation Computer Systems. 2020;108:414-23.
42. Kang M-J, Kang J-W. Intrusion detection system using deep neural network for in-vehicle network security. PloS one. 2016;11(6):e0155781.
43. Keshk M, Sitnikova E, Moustafa N, Hu J, Khalil I. An integrated framework for privacy-preserving based anomaly detection for cyber-physical systems. IEEE Transactions on Sustainable Computing. 2019.
44. Keshk M, Moustafa N, Sitnikova E, Turnbull B. Privacy-preserving big data analytics for cyber-physical systems. Wireless Networks. 2018:1-9.

45. Santos L, Rabadao C, Gonçalves R, editors. Intrusion detection systems in Internet of Things: A literature review. 2018 13th Iberian Conference on Information Systems and Technologies (CISTI); 2018: IEEE.
46. Bdair AH, Abdullah R, Manickam S, Al-Ani AK. Brief of intrusion detection systems in detecting ICMPv6 attacks. Computational Science and Technology: Springer; 2020. p. 199-213.
47. Li H, Ota K, Dong M. Learning IoT in edge: Deep learning for the Internet of Things with edge computing. IEEE network. 2018;32(1):96-101.
48. Fadlullah ZM, Tang F, Mao B, Kato N, Akashi O, Inoue T, et al. State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems. IEEE Communications Surveys & Tutorials. 2017;19(4):2432-55.
49. Benavides E, Fuertes W, Sanchez S, Sanchez M. Classification of phishing attack solutions by employing deep learning techniques: A systematic literature review. Developments and advances in defense and security. 2020:51-64.
50. Yin C, Zhu Y, Fei J, He X. A deep learning approach for intrusion detection using recurrent neural networks. Ieee Access. 2017;5:21954-61.
51. Nweke HF, Teh YW, Al-Garadi MA, Alo UR. Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges. Expert Systems with Applications. 2018;105:233-61.
52. Bai Y. Study on Attention-based LSTM Model for Multivariate Time-series Prediction: 서울대학교 대학원; 2019.
53. Alfarraj M, AlRegib G. Petrophysical property estimation from seismic data using recurrent neural networks. SEG Technical Program Expanded Abstracts 2018: Society of Exploration Geophysicists; 2018. p. 2141-6.
54. Dhillon A, Verma GK. Convolutional neural network: a review of models, methodologies and applications to object detection. Progress in Artificial Intelligence. 2020;9(2):85-112.
55. Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, Nasrin MS, et al. A state-of-the-art survey on deep learning theory and architectures. Electronics. 2019;8(3):292.
56. Akhtar N, Ragavendran U. Interpretation of intelligence in CNN-pooling processes: a methodological survey. Neural Computing and Applications. 2020;32(3):879-98.

57. Tian Y, Yuan J, Yu S, Hou Y. LEP-CNN: A Lightweight Edge Device Assisted Privacy-preserving CNN Inference Solution for IoT. arXiv preprint arXiv:190104100. 2019.
58. Shao H, Jiang H, Lin Y, Li X. A novel method for intelligent fault diagnosis of rolling bearings using ensemble deep auto-encoders. Mechanical Systems and Signal Processing. 2018;102:278-97.
59. Yousefi-Azar M, Varadharajan V, Hamey L, Tupakula U, editors. Autoencoder-based feature learning for cyber security applications. 2017 International joint conference on neural networks (IJCNN); 2017: IEEE.
60. Mayuranathan M, Murugan M, Dhanakoti V. Best features based intrusion detection system by RBM model for detecting DDoS in cloud environment. Journal of Ambient Intelligence and Humanized Computing. 2021;12(3):3609-19.
61. Zhang N, Ding S, Zhang J, Xue Y. An overview on restricted Boltzmann machines. Neurocomputing. 2018;275:1186-99.
62. Movahedi F, Coyle JL, Sejdić E. Deep belief networks for electroencephalography: A review of recent contributions and future outlooks. IEEE journal of biomedical and health informatics. 2017;22(3):642-52.
63. Chen Y, Zhang Y, Maharjan S, Alam M, Wu T. Deep learning for secure mobile edge computing in cyber-physical transportation systems. IEEE Network. 2019;33(4):36-41.
64. Heghedus C, Rong C. Artificial Intelligence Models Used for Prediction in the Energy Internet. Energy Internet: Springer; 2020. p. 321-52.