Int J Syst Assur Eng Manag (December 2022) 13(Suppl. 3):S1164–S1176
https://doi.org/10.1007/s13198-021-01533-w

ORIGINAL ARTICLE

Security threat model under internet of things using deep learning and edge analysis of cyberspace governance

Zhi Li[1] · Yuemeng Ge[2] · Jieying Guo[1] · Mengyao Chen[1] · Junwei Wang[1]

Received: 18 May 2021 / Revised: 23 October 2021 / Accepted: 18 November 2021 / Published online: 21 January 2022

Abstract Under the background of information age, it is essential to cope with network security problems, ensure the popularization of Internet of Things (IoT) technology based on the Internet, and guarantee the information security, life security, and property security of all countries and individuals. Therefore, the principle and advantages of deep learning (DL) technology is expounded first, and then an IoT security threat model is established combined with edge computing (EC) technology. Additionally, the traditional algorithm is improved to be adapted to the application environment of the current United Nations cyberspace governance actions, and is trained and optimized by data sets. Finally, a modification plan is formulated according to the actual test results. In the experiment, EC is used to establish an excellent IoT security threat model with an efficient and accurate algorithm. The result shows that DL technology and EC technology significantly improve the judgment ability of the IoT security threat model and promote the efficiency of network space governance. This model can inspire the application of emerging computer technology to the IoT network and cyberspace governance, guarantee the construction of global information interconnection, and provide a reference for future research.

Keywords Deep learning · Internet of Things · Security threat model · Cyberspace governance · Edge computing

1 Introduction

During the booming global information age, followed by the explosive growth of network security threats and the continuous deterioration of the network environment, various data leakage events and network attacks emerge endlessly worldwide. Besides, there is full of fraud and malicious information on the Internet, damaging the vitality of the global economy and global moral atmosphere. Consequently, the United Nations has called on all countries to carry out cyberspace governance actions, in response to the deterioration of the global Internet environment and cybersecurity (Kumar 2020). With the advent of the 21st century, the Internet of Things (IoT) technology has entered thousands of households, realizing the information exchange and communication between the real world and the Internet, and comprehensively perceiving the data from the physical world. However, the IoT network undergoes diversified Internet data leakage and data attacks, further affecting the lives of people in the real world.

The IoT network can collect real-time information about various things by devices and technologies such as sorts of information sensors, radio frequency identification (RFID),

✉ Junwei Wang
wangjunwei@nit.zju.edu.cn

Zhi Li
lizhi@nit.zju.edu.cn

Yuemeng Ge
0619036@zju.edu.cn

Jieying Guo
guojieying@nbt.edu.cn

Mengyao Chen
chenmengyao@nbt.edu.cn

[1] School of Media and Law, NingboTech University, Ningbo, China

[2] Faculty of International Tourism and Management, City University of Macau, Macau, China

global positioning system (GPS), infrared sensors, and laser scanners. It can also connect things with things and things with people through diverse network accesses, and realize the intelligent perception, recognition, and management of things (The 2019). With the development of IoT, human beings have entered the era of interconnection of all things surrounded by IoT terminals. The safe use of the IoT depends on the security of Internet environment. Moreover, the information processed by the IoT terminal comes not only from its own interior, but also from the external world, which is the operating basis of the terminal (Shao 2020). The traditional security threat analysis model has a good prevention effect on traditional network attacks or security vulnerabilities. However, it cannot defend against unconventional network attacks and security vulnerabilities, cannot accurately describe, effectively discriminate, and deal with potential foreign threats in time. Therefore, a new IoT security threat model is proposed here based on DL (deep learning) technology, as a theoretical support for analysts (Big and Model 2019).

Edge computing (EC), also known as proximity computing, is a data processing technology at the location physically close to the data generation address. The EC technology can deal with the relationship between the timeliness and cost of response to events well, considering the size and power of IoT terminals. The EC technology aims to solve the problem of insufficient timeliness of the existing IoT security threat model and invalid judgment of new security threats. The introduction of EC technology here improves the time-validity of the security threat model. Besides, this technology will effectively improve the timeliness and efficiency of United Nations cyberspace governance and reduce the governance costs (Van 2019).

The IoT security threat model is constructed based on deep learning (DL), and the EC technology is used to train the model to improve the model's performance. Moreover, experiments are performed on this model to verify its feasibility. This model is conducive to the United Nations cyberspace governance and lays a solid foundation for future academic research.

2 Literature review

Rizvi et al. (2020) investigated the vulnerability identification of devices based on device configuration, network topology, and user strategy, and studied attack vectors of IoT equipment including three central domains (health care, business, and family) (Rizvi et al. 2020a). Aufner (2020) explained the origin of the research on the IoT, and listed common threat modeling frameworks. The author further explored the current status of security research on the IoT, explained the generation of these threats in principle, and briefly introduced the methods to eliminate them (Aufner 2020). Ghazal et al. (2020) applied a complete information security algorithm to the IoT security. They also utilized the IoT equipment to transmit data to solve the problem that fiscal addition could not deal with different types of security vulnerabilities (Ghazal et al. 2020). Bayat et al. (2019) proposed an enhanced scheme to eliminate the security loopholes in the traditional scheme of realizing IoT user authentication through communication between sensor nodes and gateways in wireless sensor networks (Bayat et al. 2019). Singh et al. (2018) provided an optimal practice solution for existing Near Field Communication (NFC) access control applications subjected to security attacks. The solution was aimed at curbing the leakage of user key data caused by attacks like denial of service, and preventing any vulnerability that might affect any NFC application and technology (Singh et al. 2018). In summary, the existing detection methods of various traditional attacks for the IoT lack diversity, which are likely to fail to detect new attack methods and prevent them consequently. Furthermore, the previous IoT security threat model has slow response speed and poor timeliness. In view of this, a security threat model based on DL is innovatively proposed, which can actively learn the detection method of security threat vulnerabilities. Meanwhile, EC technology is utilized to improve the effectiveness of the security threat model in real-time prevention and alarm of threats.

Wang et al. (2019) stated that IoT networks would be the indispensable part of the 5G (fifth generation) network, but unfortunately, the resources of the IoT devices are strained, and many security mechanisms are difficult to implement. Finally, they designed an intrusion detection method, established the event database, and implemented an event analyzer to realize intrusion detection. They found that the intrusion detection system could detect three types of IoT attacks, namely interference attacks, false attacks, and reply attacks (Wang et al. 2019). Yang et al. (2019) proposed a data aggregation security protection scheme based on anomaly detection, and reconstructed the IoT as a network composed of small devices distributed on the Internet. In view of the limitations of previous research, they utilized the state estimation and sequence hypothesis testing technology to design the scheme. The main idea of their design was to use the high spatial and temporal correlation between continuous observations in environmental monitoring of the IoT network to predict future observations according to previous comments (Yang et al. 2019). Azmoodeh (2019) built a new data set suitable for IoT attack detection in military environment. In their experiment, the IoT network consisted of various devices connected to the Internet, from medical equipment to wearable technology. Moreover, they used a detection method based

on sequence code class selection as the classification of sample resources, created a feature map for each sample, and employed the DL method to classify malware (Azmoodeh 2019).

3 Model construction and research methodology

3.1 DL

In 2006, the new research direction DL emerged in the research field of machine learning, and researchers began to investigate it and gradually apply it to the industry. In 2012, Stanford University first built a training model called Deep Neural Network (DNN) using a parallel computing platform with 16,000 Central Processing Unit cores. This technology has made great breakthroughs in many application scenarios such as speech recognition and image recognition. In 2016, AlfaGo, an artificial Go software based on DL, defeated Li Shishi, the world's top Go master. After that, many famous high-tech companies worldwide began to invest vast resources in DL, establish research institutes for DL, and attract numerous technical research and development personnel in the DL field (Basodi and Ji 2020).

Machine learning technology can learn new knowledge or skills by studying how computers simulate or implement animal learning behaviors, to rewrite existing data structures and improve program performance. From a statistical point of view, it predicts the distribution of data, learns a model from data, and then predicts new data, which requires that test data and training data must be identically distributed. The fundamental feature of machine learning is to try to imitate the mode of information transferring and processing between neurons in the brain (Ahn et al. 2017). The most notable applications of machine learning are in the fields of computer vision and natural language processing (NLP). Obviously, DL is strongly related to the neural network, the primary algorithm and method of machine learning. In other words, DL can be regarded as an improved neural network (Dalal et al. 2019).

The artificial neural network (ANN) is a mathematical model or calculation model that imitates the structure and function of biological neural networks. ANN conducts calculation through a mass of artificial neurons connections. In most cases, ANN is an adaptive system able to change the internal structure based on external information. The modern neural network is a nonlinear statistical data modeling tool, commonly used to model the complex relationship between input and output, or to explore the mode of data. The neural network is an operational model connecting plenty of nodes (or "neurons"), and each node represents a specific output function called activation function. The connection between each two nodes signifies a weighted value of the connection signal, called weight, equivalent to the memory of the ANN. The output of the network varies from different connection modes, weights, and activation functions. The network itself is usually an approximation to some algorithms or functions in nature, or an expression of a logical strategy. Its construction concept is inspired by the functions of biological neural networks (of human or other animals). On the one hand, ANN is generally an optimization of a learning method based on mathematical statistics, as well as an application of mathematical statistics method, through which massive local structure spaces can be obtained, with the expression form of functions. On the other hand, in the field of artificial intelligence, the application of mathematical statistics can make decisions for artificial perception. In other words, through statistical methods, ANNs can possess simple decision ability and simple judgment ability similar to people, which is superior to formal logical calculus (Herzog et al. 2020).

The McCulloch and Pitts (M-P) neuron model is a pioneering artificial neuron model with dominant influence, expressing the complex biological neuron activity through a simple mathematical model. It aggregates signals from multiple other neurons into a total signal, to compare the total signal with the threshold. If the total signal exceeds the threshold, the excitation signal is generated and output. Otherwise, the model converts into the inhibitory state.

As everyone knows, a polyhedron is a geometric body surrounded by several plane polygons, which involves the concept of space. The essence of the two-dimensional space is a plane, referring to any combination of two variables without any constraints. For the unified form, any straight line L in the two-dimensional space is expressed as $a_0 + a_1x_1 + a_2x_2 = 0$. In machine learning, to compare 0 in the above equation with the dependent variable (category) y, the geometric line L is used as the algebraic expression y = 0. Similarly, the n-dimensional plane can be deduced accordingly. The set of n-dimensional vectors is called the n-1 dimensional hyperplane of the n-dimensional vector space. A straight line can divide a plane into two planes, while a plane divides a three-dimensional space into two three-dimensional spaces. Similarly, a n-1 dimensional hyperplane divides an n-dimensional space into two n-dimensional space, belonging to different categories. This classifier is the neuron. The essence of M-P neurons is to divide the feature space into two parts, and the two parts belong to two categories, respectively. Figure 1 illustrates the M-P neuron model.

In Fig. 1, the input is an eigenvector, representing the direction of change (Lambers et al. 2019). The absolute value of weight denotes the influence of input signal on the neuron.

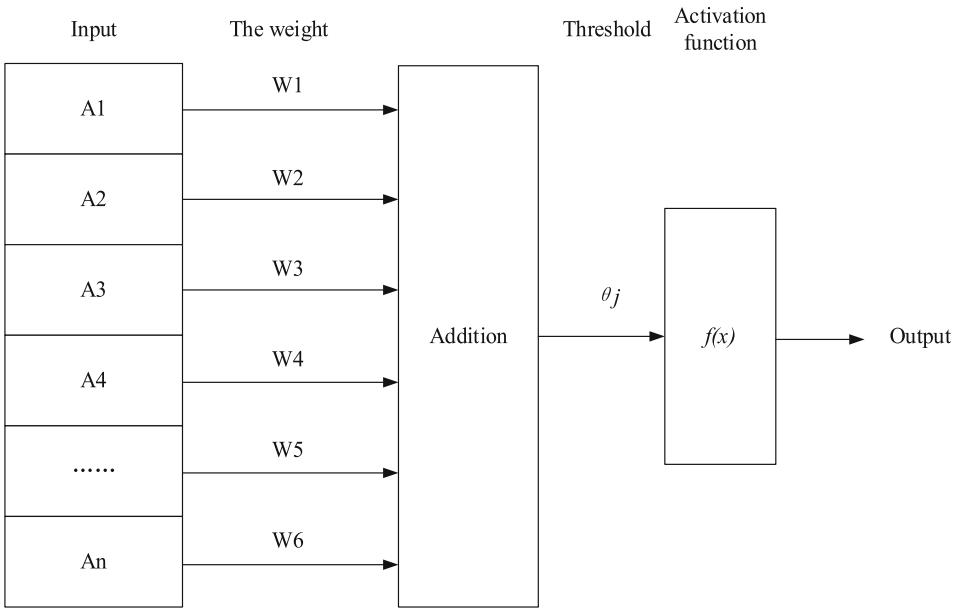

Fig. 1 M-P neuron model

The activation function can be described as follows.

1. When the absolute value of x is less than c, the ramp function can be described as Eq. (1).

$$f(x) = kx + c \tag{1}$$

The ramp function equals T when x is greater than c. Otherwise, it equals $-T$.

2. The threshold function equals $fx + 1$ when x is greater than c. When x is less than $-c$, it is 0.

The nonlinear function can be presented as follows.

1. S-type function can be displayed as Eq. (2).

$$f(x) = \frac{1}{1 + e^{-\alpha x}} (x \in R) \tag{2}$$

2. The derivative of Sigmoid function is shown in Eq. (3).

$$f'(x) = \frac{\alpha e^{-\alpha x}}{(1 + e^{-\alpha x})^2} = \alpha f(x)[1 - f(x)] \tag{3}$$

3. Bipolar Sigmoid function is shown in Eq. (4).

$$h(x) = \frac{2}{1 + e^{-\alpha x}} - 1 = (x \in R) \tag{4}$$

4. The derivative of bipolar Sigmoid function is shown in Eq. (5).

$$h'(x) = \frac{2\alpha e^{-\alpha x}}{(1 + e^{-\alpha x})^2} = \alpha \frac{1 - h(x)^2}{2} \tag{5}$$

Among the above equations, α represents the weight of each node. The initial weights and thresholds of the neural network need to be normalized between 0 and 1, because the transfer function of neurons is quite different between [0,1]. When the transfer function is greater than 1, its value changes little (and its derivative or slope is small), which is not conducive to the implementation of the backpropagation algorithm. The backpropagation algorithm needs to use the gradient of each neuron's transmission function. When the input of the neuron is too large (e.g., greater than 1), the gradient value of the independent variable at this point is too small, and the weight and threshold cannot be adjusted.

DL, containing the convolutional neural network (CNN) and deep belief network (DBN) (Sinha and Dhanalakshmi 2020), mainly simulates human neurons. Each neuron in a CNN processes received information and then transmits it to all adjacent neurons. Figure 2 illustrates the processing method of CNN.

Through Fig. 2, the small block area in the CNN can be considered as the input data at the bottom of the hierarchical structure. The information passes through all layers of the network through the forward propagation, and every layer is composed of filters, so that some significant features of the observed data can be obtained. Therefore, CNN is also called a DL method. CNN consists of the input layer, hidden layer and output layer. The input layer can process multi-dimensional data. Generally, the input layer of the one-dimensional CNN receives one-dimensional or

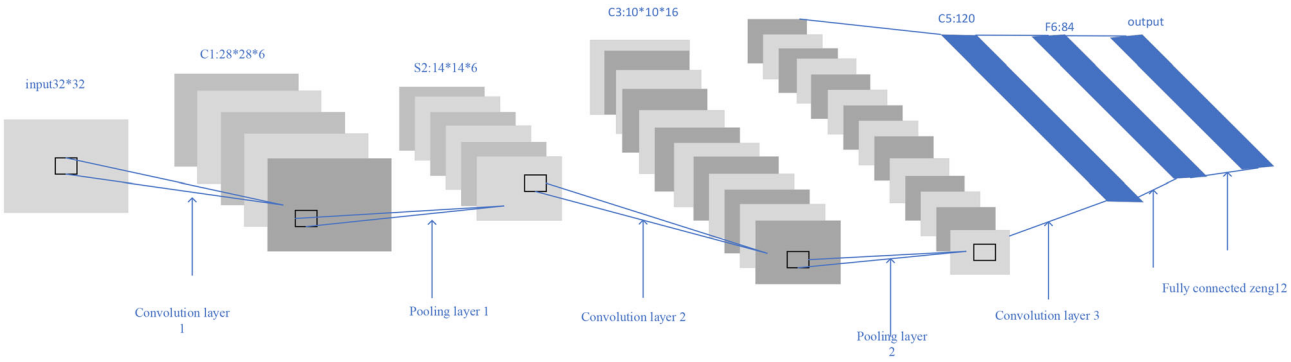

Fig. 2 Processing method of CNN

two-dimensional arrays. The one-dimensional array is usually time or spectrum sampling, and a two-dimensional array may contain multiple channels. The input layer of the two-dimensional CNN receives two-dimensional or three-dimensional arrays, and the input layer of the three-dimensional CNN receives four-dimensional array. The hidden layer includes three layers, namely convolution layer, pooling layer, and full connection layer. The convolution layer extracts the characteristics of input data, the pool layer is responsible for feature selection and information filtering, and the function of full connection layer is to perform nonlinear combination of the extracted features to obtain the output. For the image classification problem, the output layer outputs the classification label using logic function or normalized exponential function. In object detection, the output layer can be used to output the central coordinates, size, and classification of objects. In image semantic segmentation, the output layer directly outputs the classification results of each pixel (Dhillon and Verma 2020).

3.2 RFID technology

3.2.1 (1) RFID

RFID technology is a kind of automatic identification technology and a critical part of realizing the IoT technology (Yang and Chen 2020). It primarily carries out non-contact bidirectional data communication through radio frequency (RF), and uses RF to read and write recording media (electronic tag or radio frequency card), to achieve target identification and data exchange (Yan et al. 2020). The complete RFID system consists of the reader, tag, and data management system. The reader is a device that reads the information in the tag or writes the information to be stored in the tag. According to the structure and technical principal, the reader can be a read/write device or an information control and processing center of an RFID system. The electronic tag is composed of transceiver antenna, AC/DC circuit, demodulation circuit, logic control circuit, memorizer, and modulation circuit (Abdulkawi and Sheta 2020). At present, RFID technology has been widely used in all walks of life.

Ali and Haseeb (2019) found that RFID played a major role in the supply chain operation of the textile and garment industry, and it had a significant and positive impact on the supply chain performance (Ali and Haseeb 2019). Singh et al. (2017) proved that RFID tag sensors based on inkjet printing nano materials could be easily printed on flexible paper, plastics, textiles, glass, and metal surfaces, showing broad application prospects in flexible and wear-resistant electronic technology (Singh et al. 2017). Landaluce et al. (2020) proposed to combine RFID and wireless sensor networks to change their limitations and apply them to wearable sensors, which made new and promising IoT applications possible (Landaluce et al. 2020).

(2) Mobile RFID (M-RFID) network and security analysis

RFID network is a manifestation of IoT. It is a network that combines RFID technology with the Internet and provides information services (Jaballah and Meddeb 2021). The M-RFID network is an information network based on Internet and RFID technology, which exchanges information through mobile communication networks. The M-RFID network takes advantage of the unique identification characteristics of the object based on the electronic product code EPC (electronic product code) in the tag. Therefore, it can obtain the EPC information in the tag according to the RF signal between the mobile terminal implanted with the reader chip and the tag loaded on the object. Besides, it can access the EPC network through the mobile communication network connection to obtain the relevant information or services of the article, and carry out online transactions, which greatly broadens the access to information (Hou et al. 2021). However, because the M-RFID network adopts wireless data exchange technology, it facilitates data acquisition, but meanwhile, brings a huge security risk (Chiou et al. 2018).

There are two main forms of attacks in the M-RFID network, i.e., physical attacks and spoofing identity. Physical attacks are also called direct attacks. The hacker attacks different physical locations in the M-RFID network from the perspective of electronic communication. There are many ways to implement identity spoofing attacks. For example, the attacker can forge tags to deceive mobile terminals, to transmit information containing illegal data to the information management platform. Moreover, the attacker can fake the reader and writer to directly read the information in the user's label without being noticed by the user and steal the user's personal information. The RF equipment can also be used to intercept the communication information between the tag and the reader/writer to forge the electronic tag or reader/writer, to realize the next attack, such as sniffing, replay attack, or tracking (Aghili et al. 2018).

3.3 Establishment of IoT security threat model

The security threat is a potential event jeopardizing the system security, with possibilities of uncontrollable negative impacts on the system. Every system during operation faces various security threats. The security threat model can simulate the potential vulnerabilities, detect the internal operation, and assist security personnel in monitoring and judging the system state (Nicolas et al. 2019). The model can also determine the risk level of the real attack to decide the processing sequence, processing strength, and treatment scheme against multiple threats. Figure 3 reveals the establishment process of security threat model.

In Fig. 3, the establishment of a security threat model is generally divided into five steps. a. Identification of security targets. b. Building system architecture. c. Decomposing systems. d. Identifying and documenting threats. e. Assessing the risk of the threat (Aufner 2020; Rizvi et al. 2020b). Table 1 displays the primary task of each step.

Table 2 presents three primary types of security threat model.

A threat tree model is adopted in this experiment.

(2) Threat tree model.

The tree structure of logically simple threat tree model can clearly and accurately describe many threats and the operation modes of attacks. Then, it can quickly add new threats and delete invalid threats according to existing logical rules. Besides, the threat tree model has strong scalability with sub-modules. A complete complex tree structure can be split into individual sub-modules for operation, or can use a single sub-module repeatedly. Therefore, the threat tree model is highly structured and reusable (Hosseinzadeh et al. 2020).

The logic of threat tree models relies on the cause and effect of events. The tree root is the final attack target. The

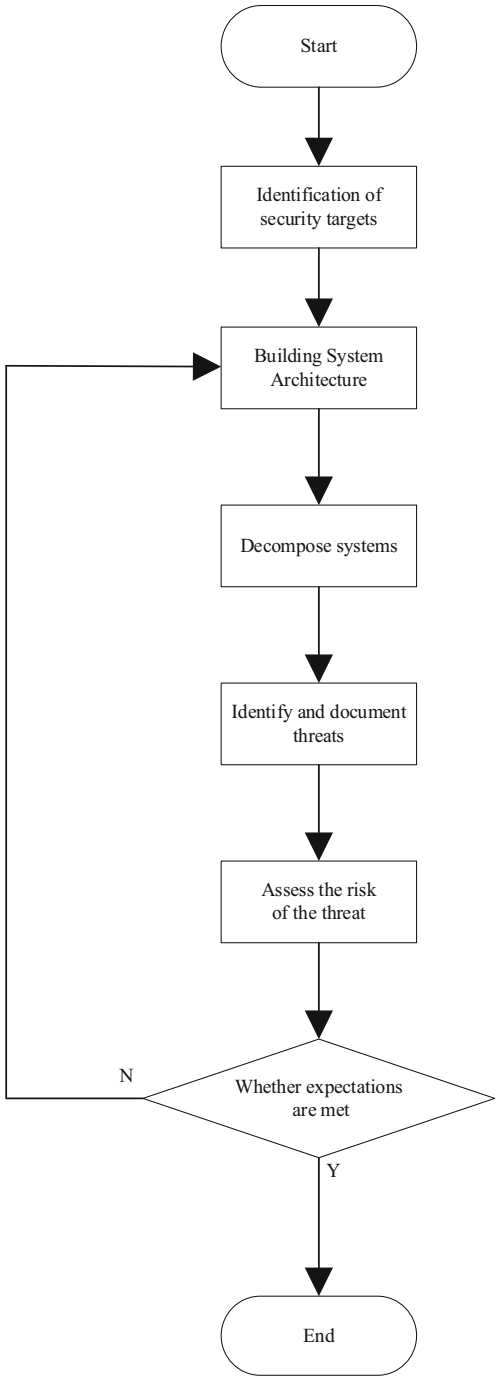

Fig. 3 Establishment process of the security threat model

leaf nodes at all levels below the root represent the method to achieve the target represented by the nodes at the higher level, and the leaf at the top level denotes the simplest attack method.

Each threat tree model can be described as a triple $T:\{G,E,G_0\}$. Among them, G denotes the node, while E stands for the connection between nodes, and G_0 represents the root node. There are two types of relationship

Table 1 Main tasks of the five steps

Step	Main task
Identification of security targets	The model detects the valuable parts needing protection in the system. This stage is conducive to clarifying the purpose and next task plan
Building system architecture	Understand the target system meticulously, involving system functions, system architecture, compositions, physical deployment, and solution. Determine user permissions, and uses chart hierarchical system (subsystem, bounded confidence, data stream) to find the potential defects in the actual design process of the system
Decomposing systems	Clarify the system components (data server, communication device, data acquisition module), the relationship, as well as the processing process between the components
Identifying and documenting threats	Confirm security objectives, list all potential threats in the system, and identify the security threats potentially affecting the system based on the understanding of the system architecture and potential defects
Assessing the risk of the threat	Figure out the impact degree and risk level of the threats, and give priority to the threat with greater risk

Table 2 Primary security threat models

Name	Application	Difficult point
Threat tree model	It is effective against serial threats. This model is easy to design and can provide accurate and clear detection results	
Threat model based on threat attributes	It adopts security timing analysis method to detect threats by extracting attribute features of threats	Building the feature base and model base
Bayesian network model	It can detect distributed threats, but it is not accurate enough in practical applications	The design and implement of the detection algorithm

between the nodes of the threat tree model, namely "AND" and "OR" (Aditya et al. 2019).

a. "AND": each lower-level node needs to be implemented to achieve the upper-level node, thus to launch an attack, as shown in Eq. (6).

$$\{G_1 \cap G_2 \cap G_3 \cap \cdots\} = Goal\ G_0 \tag{6}$$

b. "OR": just one lower-level node achieved can perform the upper-level nodes, as presented in Eq. (7).

$$\{G_1 \cup G_2 \cup G_3 \cup \cdots\} = GoalG_0 \tag{7}$$

Figure 4 indicates the complete threat tree model.

4 Research on threat risk assessment method

The threat factor set is described as $\{ \mathrm{W}_1, \mathrm{W}_2, \ldots, \mathrm{W}_n \}$. Denote $P_{(i,j)}$ as the probability that a threat factor W_i will attack the system, i.e., the probability of threat factor W_i to W_j, which can be written as Eq. (8), where $P_{(i,j)} \in (0, 1)$.

$$P(\mathrm{W}_n) = P(\mathrm{W}_n / \mathrm{W}_1\ \mathrm{W}_2 \ldots \mathrm{W}_n) \tag{8}$$

T_i represents the risk level of W_i, indicating the harmful scale to system security. R_i denotes the reliability of the i-th subsystem. $R_{i,j}$ expresses the reliability of the j-th secondary subsystem in the i-th subsystem (Buldas et al. 2020). The reliability of the parallel subsystem with associated threat factors is shown as Eq. (9). The reliability of the subsystem with threat factors in "OR" relationship is presented as Eq. (10).

$$R_{(i,j)} = 1 - \prod_{\mathrm{i}-1}^{\mathrm{n}} (1 - P(\mathrm{W_i})); i = 1, 2, 3, \ldots n \tag{9}$$

$$R_{(i,j)} = \prod_{\mathrm{i}-1}^{\mathrm{n}} (1 - P(\mathrm{W_i})); i = 1, 2, 3, \ldots n \tag{10}$$

The useability R_W of threat factor W_i is expressed by Eq. (11).

$$R_W = 1 - \prod_{i=1}^{n} \left(1 - \prod_{j=1}^{mj} R_{(i,j)} \right) \tag{11}$$

Threat factor W_i corresponds to t risk level ($K_i^t = k_i^{(t_{max})}$). The security threat of W_i is shown in Eq. (12), where a is 0.295.

$$Q_i = R_i \times a \tag{12}$$

<table>
<tr><td colspan="12">Information security</td></tr>
<tr><td colspan="4">confidentiality</td><td colspan="5">integrity</td><td colspan="2">availability</td><td></td></tr>
<tr><td colspan="4">sniffer</td><td colspan="5">The witch attack</td><td>Component failure</td><td colspan="2">Dos attack</td></tr>
<tr><td colspan="3">Network eavesdropping</td><td>Pass off as a legitimate tag or reader</td><td colspan="2">Forging multiple identities</td><td>Forging multiple identities</td><td>Component damage</td><td>Run out of energy</td><td colspan="2">Flooding attack</td></tr>
<tr><td>The malicious node joins the route</td><td>Intercept and analyze the passing information</td><td>Destroy the communication key</td><td>Can be used to counterfeit labels or terminal devices</td><td>ID of multiple dummy values</td><td>Monitoring communication link</td><td>Analysis packet</td><td>Label damaged
Place high power RF</td><td>Radio frequency device</td><td colspan="2">Broadcast the hello packets</td></tr>
</table>

Fig. 4 Complete threat tree model

4.1 Edge computing

4.1.1 (1) An Overview of EC

EC is an open platform close to the one side of things or data source, with the core capabilities of network, calculation, storage, and application. The edge of the network can be any entity including the data source and the cloud computing center. These entities are equipped with EC platforms integrating the network, computing, storage, and core application competence, providing real-time, dynamic, and intelligent computing services for end users. Different from the processing and algorithm decision in cloud computing, EC makes intelligence and computing closer to the reality.

EC can reduce time delay because it can process data closer to the data source, rather than in the external data center or cloud. Additionally, it costs little, enabling companies to spend less on data management solutions for local devices than on cloud and data center networks. With the increase in IoT terminals, the speed of data generation and record transmission grows. Therefore, the network bandwidth becomes limited, making the cloud overwhelmed and causing greater data bottlenecks. EC can run fast and efficiently with little delay. Mobile edge computing (MEC) greatly improves users' service experience, because it has an independent server with strong computing and storage capabilities and is extremely physically close to the network terminal equipment. Moreover, the model can constantly learn and adjust itself according to individual needs to provide personalized interactive experience. The EC distributes concentratedly (Garg et al. 2019), avoiding personal privacy disclosure.

4.1.2 (2) EC Based on DL

A Mobile Edge Computing (MEC) network is designed based on distributed DL. By deploying a computing server on the client, it avoids tracing the traffic generated by the application back to the remote data center, and provides an effective method to build a bridge between the user and the edge server. Besides, MEC utilizes K parallel DNNs to effectively generate diversion decisions, reduce the delay of executing computing tasks, and save energy consumption for those delay-sensitive cloud computing applications. The purpose of MEC algorithm is to find a diversion strategy function and generate the optimal diversion strategy. The entities applied to the MEC framework include related external objects, the center, and system management of MEC. The core of the MEC network diversion algorithm is the center, containing the platform and the materialized virtual infrastructure. The center can be further divided into the virtual infrastructure layer, the platform, and application. The platform is a set of necessary functions to run MEC applications on the virtual infrastructure, including virtualization management and functional components of the MEC platform. Virtualization management takes virtual infrastructure as a platform to realize the organization and configuration of MEC. The application provides a flexible and efficient operation environment with resources allocated on demand and multiple applications running independently. The functional components of MEC platform mainly provide various services, with access to the application through the open Application Programming Interface. These services include wireless network information, location, data distribution rules, persistent storage of access, and configuration of Domain Name Server proxy service. The MEC algorithm is composed of distributed actions and DL (Sha et al. 2020; Pereira et al. 2021). The size of input data and output data is expressed as d. For each input d, K distributed shunting decision makers efficiently generate K candidate shunting decisions $\{x_k|k \in \eta\}$, where $\eta = \{1, 2, 3, \cdots, K\}$. Then, the diversion action with the lowest total energy cost is selected as the output, expressed as x^*. Finally, the memory data group (d, x^*) is further

stored in the memory structure to train the K distributed diversion decision makers. When the MEC network device is maliciously attacked, it will allocate some computing resources to start its defense mechanism. Figure 5 indicates the algorithm flow.

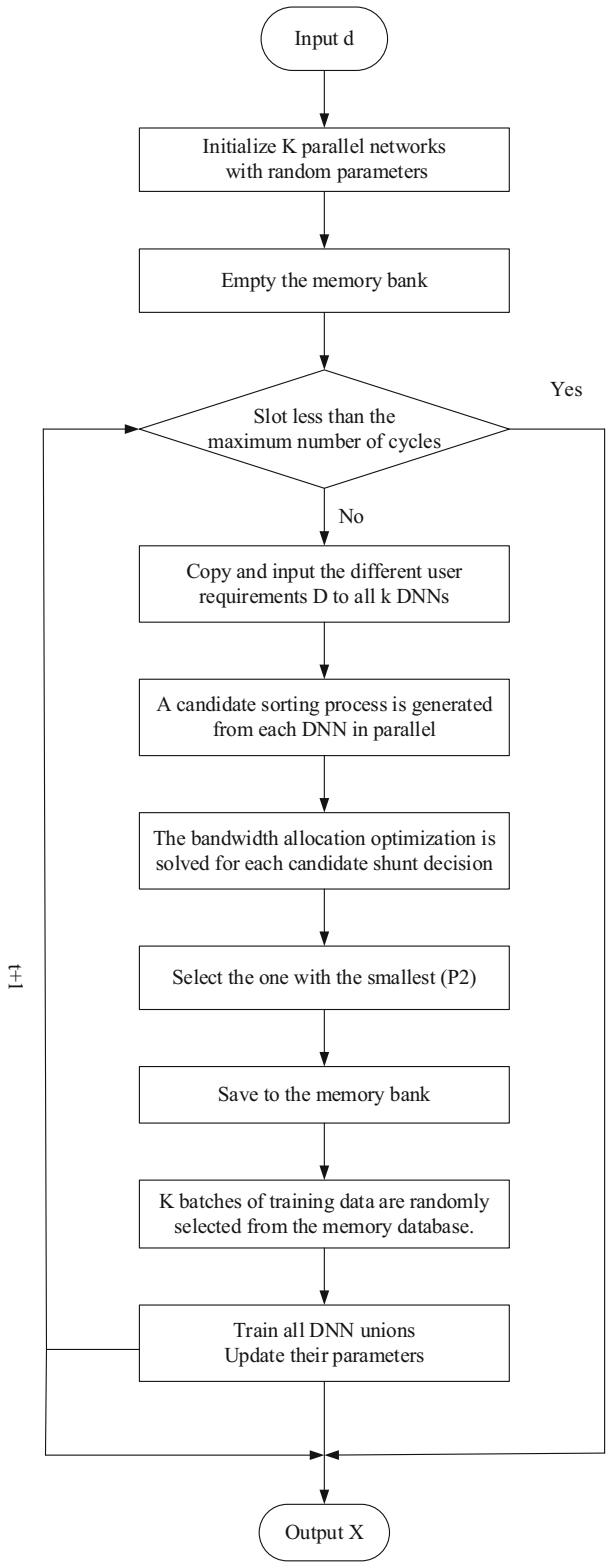

Fig. 5 Steps of MEC algorithm

1) Enter d_t;
2) Use random parameters θ_k to initialize K DNNs, where $k \in \eta$;
3) Empty the memory bank;
4) When the gap t is less than the maximum number of cycles G, perform the following operations; otherwise, the training ends and jumps out of the cycle;
5) Copy and input the requirement d_t of different users to all K DNNs;
6) Generating a candidate diversion decision from each DNN in a parallel manner;
7) When the candidate shunting decision $\{x_k\}$ is available, the optimal bandwidth allocation is solved for each candidate shunting decision respectively;
8) Select the one that minimizes the bandwidth allocation problem from the candidate shunting decision $\{x_k\}$;
9) Save (d_t, x_t^*) to memory;
10) Randomly select K batches of training data from the memory;
11) Train all DNNs and update the parameter $\theta_{k,t}$;
12) Skip to step 5 and add 1 to time slot t;
13) Output x_t^*.

4.2 Experimental design and data sets

Firstly, the threat degree of the boundary threat factor for the security target can be calculated using the given threat risk calculation method. Then, based on this result, the threat risk value of each threat factor and each attack path in the tree structure can be obtained, according to the possibility of each threat factor being utilized.

The specific steps are as follows. a. Define the reliability of the boundary threat factors. b. Use the reliability of boundary threat factors to calculate the probability of each threat attack and the reliability of each threat factor or path. c. Calculate weights for each threat factor or the full utilization path based on the correlation. d. Determine the risk value and path utilization value of threats by the reliability and weight of threat factors.

Simulation results display the comparison between detection results of the Linux Intrusion Detection System of the security threat model and the attack detection system based on the mainstream security threat model, from the perspectives of Precision, Recall, and F1-score. Precision and recall are usually used to evaluate the analysis effect of binary classification models. However, when these two indicators conflict, it is difficult to compare between models. Therefore, F1-score is proposed, which is an index

to measure the accuracy of the model. It takes into account the precision and recall of the classification model at the same time. It can be regarded as a harmonic average of the accuracy and recall of the model. Its maximum value is 1 and its minimum value is 0. F1-score is calculated according to Eq. (13).

$$F_1 = \frac{2 * (precision * recall)}{(precision + recall)} \tag{13}$$

In Eq. (13), *precision* represents the precision, *recall* denotes the recall rate. F_1 represents an index used to measure the accuracy of binary classification model in statistics, which can be seen as a harmonic mean of precision and recall of the model.

The IoT simulation data set contains 7 million network transactions, and Cooja of Contiki simulates multiple locations in smart home networks. The open-source network penetration test framework Scapy is used to extract data by stripping each network packet. The input data preprocessing procedure simplifies 7 million network transactions and reduces the input data set to 697,880 records. The network data set is collected from two separate simulations, of which the first simulating all benign network transactions, and the second simulating mixed malignant network transactions. Each network transaction in the second network simulation is marked as malicious affairs because the entire network is affected by the vicious activities that occur within the network. Among these 697,880 records, a total of 390,540 records belong to malicious affairs, and the remaining 307,340 records belong to benign affairs. The original data set can be obtained by processing the files generated by simulation, which contains information such as packet serial number, source IP, target IP and data value, and protocol types. These columns (attributes) are used to calculate the transmission and reception rates of nodes. The test data set includes 232,394 records (33.333% of the input data set). Among them, each record consists of six values, namely transmission rate, receiving rate, transmission-receiving ratio, duration, transmission mode, source Internet Protocol (IP), target IP, data value information, and binary label information. Binary label information indicates whether network transactions are benign or malicious. The training data set containing 464,788 labeled records (66.667% of the input data set) is used to train the model. During the operation of the test data set, no binary label information is provided, which shows which binary classification each record belongs to.

5 Experimental results of DL-based IoT security threat model and edge analysis

5.1 Results of IoT security threat model based on DL

The threat tree model is used to perform the safety judgement, and is continuously trained by DL, which is conducive to the detection of the security holes in the system and the attack sets defined by the security holes. Figure 6 reveals the threat tree model after pruning.

Furthermore, Table 3 displays the risk level data of mobile RFID network calculated by the equations in Sect. 2.3.

According to the data in Table 3, the threat factor with higher reliability and degree are more likely to be utilized, and the threat factor with higher risk grade is more dangerous. Among them, the risk grade of the counterfeit communication system EPCIS and the relay device is the highest, reaching 3. The risk grade of RF power plant and intercepted communication is the lowest, which is only 1. Additionally, the reliability of the counterfeit communication system EPCIS of the model is the lowest, only 0.2, but the reliability of most threat factors of the model is more than 0.6, which belongs to the normal reliability range. The risk coefficients of tags intercepted by the attacker and the mobile terminal communication using relay equipment to retransmit information are the highest. Therefore, limited prevention and treatment are needed for these two attacks, namely radio frequency eavesdropping and replay attacks.

Reliability refers to the possibility and correlation of system risk based on risk identification and estimation by comprehensively considering the probability of risk occurrence, loss range, and other factors. Risk level is the risk level of an enterprise determined by comparing the possibility of risk with the recognized safety standards, determining whether control measures are needed and to what extent. The threat factors with higher reliability are more likely to be utilized, and the threat factors with higher risk level are more dangerous. Threat factors primarily include tampering with data, information disclosure, identity deception, denial, privilege promotion, and denial of service.

5.2 Results of the EC of the united nations cyberspace governance

Figure 7 provides the computational resource consumption with iterations at the time point t.

As Fig. 7 shows, the computational resource consumption of edge network terminals is proportional to the

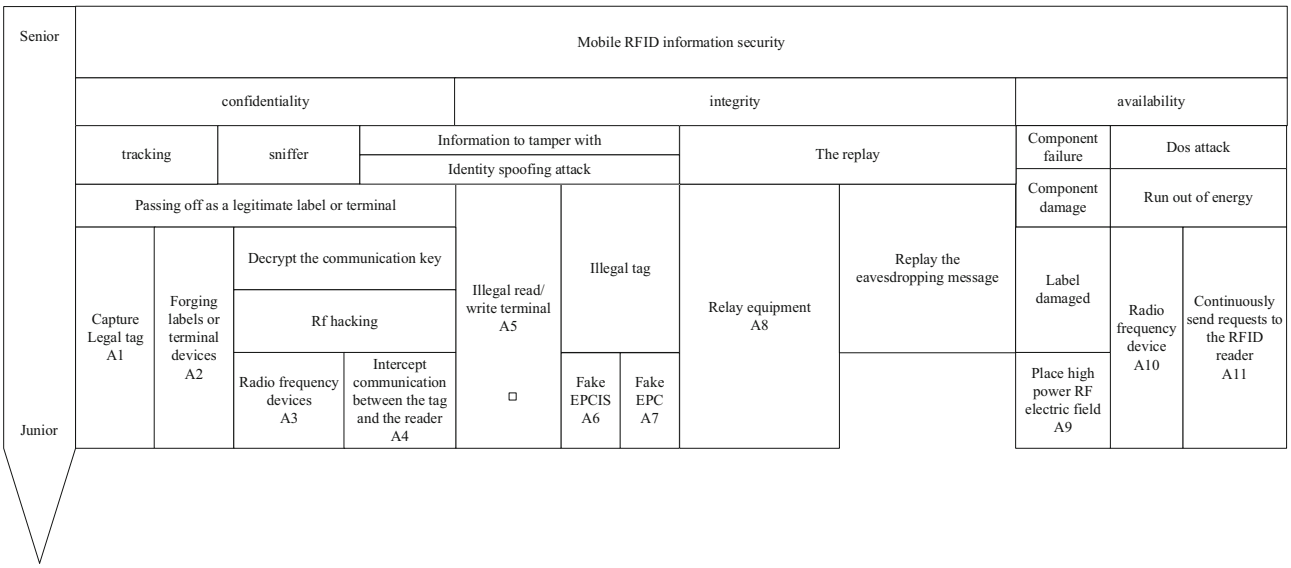

Fig. 6 The threat tree model after pruning

Table 3 Risk level data sheet

Threat factor	Reliability	Degree	Risk grade
A1	0.61	4	2
A2	0.64	5	2
A3	0.77	4	3
A4	0.56	5	1
A5	0.24	1	2
A6	0.20	2	3
A7	0.61	3	2
A8	0.83	4	2
A9	0.78	4	2
A10	0.71	4	3
A11	0.68	5	1

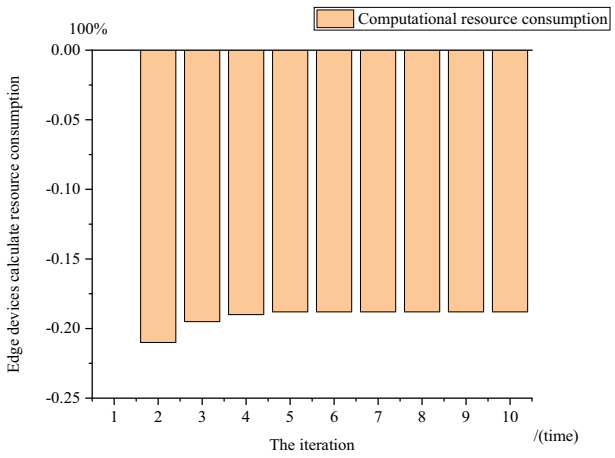

Fig. 7 The computational resource consumption of edge network terminals

number of iterations. The former increases with the growth of the latter, and then tends to be stable at a certain level, no longer increasing or decreasing. The reason is that at a fixed time point, the attack intensity is constant, and the security defense strategy called before by the edge network terminal is improved to the optimal strategy. Moreover, according to previous studies, since the response time of the defense mechanism enlarges with the growth of attack intensity of the system, the computational resource consumption of edge network terminals increases correspondingly. This is because the edge network terminal will schedule a part of the computational resource for the security defense strategy generally within the allowable range, causing less burden on the whole system, thus avoiding long response time.

5.3 Performance comparison of security threat models

Figure 8 illustrates the performance comparison between the mainstream security model and the proposed security model.

From Fig. 8, at the beginning of the model operation, the detection rate of the proposed security threat model is higher than 90%, and the Precision is 1.25 times higher than that of the original mainstream model of 40%. Moreover, with the increase of operation times, the Precision of the original mainstream model first increases to 85% and then decreases to 73%, which is unstable. At the beginning of the model operation, the Recall of the proposed model is higher than 85%, and the Precision is 10% higher than that of the original mainstream model, which is 70%. As the running continues, the Recall of the original

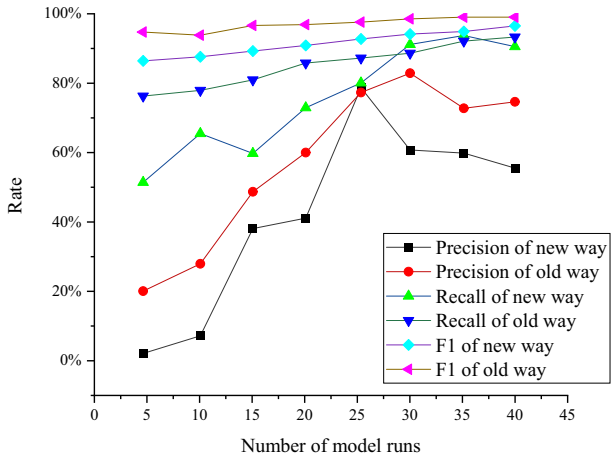

Fig. 8 Performance comparison between the mainstream security model and the proposed security model

mainstream model first rises to 95% and then falls to 50%, which is unstable. At the beginning of the model operation, the F1-score of the proposed model is higher than 90%, and the Precision is nearly doubled that of the original mainstream model, which is 50%. With the increase in operation, the Recall of the original mainstream model gradually increases to 93%. The results demonstrate that the performance of the proposed model is significantly better than that of the original mainstream model.

6 Conclusion

The IoT security threat model based on DL and EC is constructed for the United Nations cyberspace governance through combining the data and the actual situation of IoT security threats. Then, the model is trained by DL and tested on actual data. The experiment proves that the IoT security threat model based on DL can make the computer autonomously learn from attacks, and achieve the purpose of continuous model training through the application of the actual data set. Moreover, the EC speeds up the data processing and analysis of the model, and reduces the required network traffic and time delay, guaranteeing the security of the computer system. Moreover, EC can effectively protect personal privacy and information security.

There are still some deficiencies in this work. (1) The factors considered in the evaluation are not comprehensive. (2) Other methods are not considered to improve the efficiency of equivalent partition and boundary value analysis of test input space. Therefore, future research will study whether there is a correlation between the impact caused by the successful implementation of attacks with reputation loss, economic loss, and the recurrence of successful attacks, to reduce the threat to the IoT security. In addition, in later research, lightweight formal methods, such as model checking and theorem proving, can be used to formally verify the key functional modules in the design and implementation of software system.

Author contributions All authors listed have made a substantial, direct and intellectual contribution to the work, and approved it for publication.

Funding This work was supported by the National Social Science Fund of China of the Youth Project "A Comparative Study on the Laws of Global Cyberspace Security Governance and Its Enlightenment to China" (Grant No. 19CXW039).

Declarations

Conflict of interest All Authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

Abdulkawi WM, Sheta AA (2020) High coding capacity chipless radiofrequency identification tags. Microw Opt Technol Lett 62(2):592–599

Aditya S, Ramasubbareddy S, Govinda K (2019) Estimation of web vulnerabilities based on attack tree and threat model analysis. J Comput Theor Nanosci 1(2):122–149

Aghili SF, Ashouri-Talouki M, Mala H (2018) DoS, impersonation and de-synchronization attacks against an ultra-lightweight RFID mutual authentication protocol for IoT. J Supercomput 74(1):509–525

Ahn J, Cho S, Chung DH (2017) Analysis of energy and control efficiencies of fuzzy logic and artificial neural network technologies in the heating energy supply system responding to the changes of user demands. Appl Energy 5(190):222–231

Ali A, Haseeb M (2019) Radio frequency identification (RFID) technology as a strategic tool towards higher performance of supply chain operations in textile and apparel industry of Malaysia. Uncertain Supply Chain Manag 7(2):215–226

Aufner P (2020) The IoT security gap: a look down into the valley between threat models and their implementation. Int J Inf Secur 19(1):3–14

Azmoodeh K (2019) Robust malware detection for intemet of Battlefield things devices using deep eigenspace leaming. IEEE Trans Sustain Comput 4(1):88–95

Basodi S, Ji C (2020) Gradient amplification: an efficient way to train deep neural networks. Big Data Mining Analytics 3(3):196–207

Bayat M, Atashgah MB, Barari M et al (2019) Cryptanalysis and improvement of a user authentication scheme for internet of things using elliptic curve cryptography. Int J Netw Secur 21(6):897–911

Buldas A, Gadyatskaya O, Lenin A (2020) Attribute evaluation on attack trees with incomplete information. Comput Secur 88(6):1–17

Chiou SY, Ko WT, Lu EH (2018) A secure ECC-based mobile RFID mutual authentication protocol and its application. Int J Netw Secur 20(2):396–402

Dalal G, Gilboa E, Mannor S (2019) Chance-constrained outage scheduling using a machine learning proxy. Power Syst Trans 9(3):232–251

Dhillon A, Verma GK (2020) Convolutional neural network: a review of models, methodologies and applications to object detection. Prog Artif Intell 9(2):85–112

Garg S, Aujla G, Kumar N (2019) Tree-based attack-defense model for risk assessment in multi-UAV networks. Consum Electron Mag 8(6):35–41

Ghazal TM, Hasan MK, Hassan R et al (2020) Security vulnerabilities, attacks, threats and the proposed countermeasures for the internet of things applications. Solid State Technol 63(1):1566–1567

Herzog S, Tetzlaff C, Wrgtter F (2020) Evolving artificial neural networks with feedback. Neural Netw 123(1):153–162

Hosseinzadeh M, Quan T, Ali S et al (2020) A hybrid service selection and composition model for cloud-edge computing in the Internet of Things. Access 8(1):99–100

Hou Y, Liang H, Liu J (2021) Super lightweight mobile RFID authentication protocol for bit replacement operation. Int J Mobile Comput Multimed Commun (IJMCMC) 12(1):63–77

Jaballah A, Meddeb A (2021) A new algorithm based CSP framework for RFID network planning. J Ambient Intell Humaniz Comput 12(2):2905–2914

Kumar C (2020) The united nations and global environmental governance. Strateg Anal 3(4):99–107

Lambers L, Ricken T, Knig M (2019) Model order reduction (MOR) of function-erfusion-rowth simulation in the human fatty liver via artificial neural network (ANN). PAMM 19(1):23–51

Landaluce H, Arjona L, Perallos A et al (2020) A review of IoT sensing applications and challenges using RFID and wireless sensor networks. Sensors 20(9):2495

Nicolas G, Andrei M, Bradley G (2019) Deep learning from 21-cm tomography of the cosmic dawn and reionization. Mon Not R Astron Soc 3(1):1–7

Pereira P, Araujo J, Melo C (2021) Analytical models for availability evaluation of edge and fog computing nodes. J Supercomput 7(2):191–213

Poltavtseva M (2019) Big data management system security threat model. Autom Control Comput Sci 53(8):903–913

Rizvi S, Pipetti R, Mcintyre N et al (2020a) Threat model for securing internet of things (iot) network at device-level. Internet of Things 11(2):240–241

Rizvi S, Pipetti R, Mcintyre N (2020b) Threat model for securing internet of things (IoT) network at device-level. Internet Things 11(2):100–121

Sawyer S (2019) The. J Am Soc Inform Sci Technol 70(6):638–639

Sha K, Yang T, Wei W (2020) A survey of edge computing-based designs for IoT security. Digital Commun Netw 6(2):195–202

Shao B (2020) Design method of agricultural environmental monitoring system based on the internet of things. J Agri Mech Res 5(1):344–391

Singh R, Singh E, Nalwa HS (2017) Inkjet printed nanomaterial based flexible radio frequency identification (RFID) tag sensors for the internet of nano things. RSC Adv 7(77):48597–48630

Singh MM, Adzman K, Hassan R (2018) Near field communication (NFC) technology security vulnerabilities and countermeasures. Int J Eng Technol 7(31):298–305

Sinha B, Dhanalakshmi R (2020) Building a fuzzy logic-based McCulloch-Pitts neuron recommendation model to uplift accuracy. J Supercomput 1(8):12–28

Van K (2019) Asymmetrical training scheme of binary-memristor-crossbar-based neural networks for energy-efficient edge-computing nanoscale systems. Micromachines 2(3):23–42

Wang N, Wang P, Alipour-Fanid A et al (2019) Physical-layer security of 5gwireless networks for iot: challenges and opportunities. IEEE Internet of Things J 6(5):8169–8181

Yan B, Chen X, Yuan Q et al (2020) Sustainability in fresh agricultural product supply chain based on radio frequency identification under an emergency. CEJOR 28(4):1343–1361

Yang H, Chen W (2020) Game modes and investment cost locations in radio-frequency identification (RFID) adoption. Eur J Oper Res 286(3):883–896

Yang L, Ding C, Wu M et al (2019) Robust detection of false data injection attacksfor data aggregation in an Internet of Things-based environmental surveillance. Computer Networks 129(1):410–428