Computing and Informatics, Vol. 43, 2024, 1029–1052, doi: 10.31577/cai_2024_5_1029

APPLICATION AND EFFECTIVENESS OF IOT EDGE AND FOG COMPUTING TECHNOLOGIES IN SMART ENERGY DEVELOPMENT WITH THE USE OF ENCRYPTION ALGORITHMS AND SECURITY SYSTEMS

Wei Wu

School of Internet of Things Technology
Wuxi Institute of Technology
1600 Gaolang W Rd
WuXi, China
e-mail: weiwu5226@gmail.com

Abstract. The invention of new technologies and the upgrading of existing ones to carry out assigned duties for human requirements efficiently and quickly are among the new prospects made possible by the evolution of artificial intelligence and information technology. This study aims to evaluate the performance of the Internet of Things solutions based on fog and edge computing. This article examines fog computing architectures and discusses the primary potential security and trust issues. The topical issue is to analyze and improve the security of data transmission from electronic devices and media in fog clouds using data encryption and encryption algorithms. The study confirms that depending on the amount of information to be processed, each block encryption algorithm can have a different key length. The study investigates the Advanced Encryption Standard (AES) algorithm as a more typical and efficient block encryption algorithm with key lengths of 128 and 256 bits. The paper contains a comparative characteristic of data processing speed with the size of 512 MB on the example of the AES and PRESENT block encryption algorithms, demonstrating an effective software implementation. The main scientific novelty of the study is that the efficiency of such coding algorithms as AES and PRESENT increases with a decrease in their size (from 256 to 128-bit and from 128 to 80-bit, respectively). It is necessary to expand our results for a more complex residential system.

Keywords: Internet of Things, smart grids, Advanced Encryption Standard, fog computing

Mathematics Subject Classification 2010: 68P25

1 INTRODUCTION

The introduction of information technology and automated systems capable of processing tasks without the human factor via smart technology and the Internet of Things is highly relevant these days. The advancement of information technology and artificial intelligence opens new opportunities such as the development of new technology and the improvement of existing ones to perform assigned tasks for human needs effectively and in the shortest possible time. As a result of this, new technologies have emerged that allow physical devices (peripheral and electronic devices) to communicate with one another via a communication network. They can exchange and manage data in real-time; for example, by connecting a device to the switching network, a user can control the electricity costs and consumption of a living room [1]. This article examines fog computing architectures and discusses the primary potential security and trust issues. The topical issue is to analyze and improve the security of data transmission from electronic devices and media in fog clouds using data encryption and encryption algorithms.

1.1 Literature Review

Smart systems (smart grids) are one example of the modern technologies. They are outfitted with cutting-edge technologies such as 5G, wireless sensor networks, and a well-developed information technology infrastructure. Through the use of connected actuator sensors and gateways to wireless sensor networks, these technologies are capable of providing necessary services and ensuring the proper operation of water and power supply systems [2]. IoT is regarded as a new evolution for smart energy systems in which data obtained from newly connected IoT devices are used to develop new technologies. This improves real-time decision-making by improving performance and solving critical and failure problems. The primary function of IoT devices is to collect and exchange data about the physical world securely and reliably [3, 4, 5].

Interoperability issues arise with the development of IoT technologies due to the large number of devices connected and data being processed. In this context, integrated software and hardware technologies are applied. They are optimized for a variety of needs and are well-suited to energy smart grids as a means of increasing reliability, security, and performance. IoT is actively being implemented in consumer electronics, smart technology, and medicine, as well as smart energy, where IoT provides tools to implement control and monitoring of energy consumption and reduce the energy consumption [2, 3].

The process advantages of using IoT are as follows [3, 6]:

1. Maximized fault tolerance and detection for network operation under sensitive conditions in case of network resource depletion due to the automation of the route marking and identity management by the operations centers;
2. provides interoperability and operation of complex software, where the interoperability of IoT devices focuses on the choice of protocol gateways which must be transparent, verifiable, and secure, thereby improving the quality of service;
3. The uninterrupted power supply in constant power-up mode ensures stable operation for stand-alone IoT networks;
4. Large amounts of processed data (from petabytes to zettabytes) and formats are interpreted with data information.

IoT is closely connected with cloud computing and peripheral devices, which have significantly changed the perception of information technology. This is due to the lack of upfront investment in information technologies (IT), the proportional costs, and the facilitation of fast processing of the data that comes from electronic devices and media. Depending on the number of devices connected to the network, applications are inherently sensitive to delays due to a large number of data arrivals [7].

Although IoT for smart environments has made great progress, intelligent computing is still a difficult technology to capture, store, process, and analyze enormous amounts of data [8]. The exponential increase in the number of additional sensors makes it more difficult for users to exchange, analyze, and learn from big data [9]. On the other hand, the longevity of computer services is correlated with security and privacy challenges. Due to privacy infringement, users may uninstall a program. Particularly, the IoT in the physical world demands extra care for safety. IoT data and devices may be attacked using malware (such as distributed denial of service, or DDoS), which can cause irreparable harm [10].

While there are several advantages to be gained from enhancing a building's operation and management efficiency, the application of AI technology in Smart Buildings is growing [11]. A possible solution to deal with the data explosion of IoT applications is AI (for example, machine learning) [12, 13]. The knowledge that AI gains from IoT data may have a wide range of positive effects on IoT consumers' Quality of Service [14]. Almost 40 % of all energy used worldwide is now attributed to buildings, but AI technologies provide a major opportunity to cut energy use by enhancing automation, control, and dependability. At the same time, these technologies may be used to raise building inhabitants' levels of security and comfort.

A system-level architecture called fog computing tries to evenly distribute resources and services from the cloud to IoT [15, 16]. Aggregators employ fog nodes to detect data for aggregation and overcome the problem of resource limitations since cloud computing is unable to manage the exponential rise of data from edge devices in smart buildings [17]. The five benefits of fog computing are as follows [18]:

1. Managing the data explosion and network traffic strain;
2. Distributed and low-latency computing;
3. Going beyond the resource limit of terminal equipment;
4. Sustainable energy consumption; and
5. Intelligent computing.

Hence, using intelligent fog computing on IoT devices offers the potential to increase energy sustainability.

By utilizing network and application information, the study [19] introduced a unique load-balancing technique to distribute the burden of SDN across several vehicle sensors. The suggested approach may balance a group of applications across the linked SDN-connected vehicle sensors. The suggested model's performance has been compared with that of well-known heuristic-based models after the authors evaluated this methodology on various datasets. The created model is highly accurate in detecting patterns in both sparse and dense datasets. The deep feedforward model's training instances are greatly increased using the entropy-based active learning technique.

The work [20] uses SDN and deep learning to design intrusion detection for IoT traffic. SDN provides smart management of networks by decoupling control and data planes. Deep learning has proved to be a better approach in almost all areas. The deep learning-based classifiers are providing better results in current IDS as compared to traditional classifiers. The proposed model detects any intrusion in networking systems, in particular IoT networks. The performance of the proposed model is evaluated using F1, Precision, Recall, Accuracy, and other metrics. The results for the proposed model have shown a notable improvement over other intrusion detection models for IoT. Considering future work, the other deep learning classifiers can be explored for improvisation. The simulation work of the proposed model can be tested in a real environment with increased attacks and normal traffic.

In the paper [21], a remote monitoring system for the early detection of chronic diseases and COVID-19 virus infection inpatients is proposed. The system uses machine learning and deep learning methods to classify patients' conditions into healthy and unhealthy classes. For providing patient health status diagnosis Decision Trees, Random Forest, SVM, Gradient Boosting, and Logistic Regression algorithms are used [22]. The results of the experiments on the COVID-19 datasets showed that the Decision Tree, Random Forest, and Gradient Boosting algorithms achieved the best results compared to SVM and Logistic Regression. The evaluation of the proposed approach on the Chest X-ray images dataset is also provided. Logistic Regression, Decision Tree Classifier, Gaussian Naive Bayes, KNN, and SVM algorithms, and ensemble architecture are used to implement the data classification. By applying these methods data are classified into COVID-19 and normal classes.

The paper offers a comparative analysis between classical methods like Logistic Regression, Decision Tree, Gaussian Naive Bayes, KNN, and SVM and an ensem-

ble learning method that has been constructed. While applying methods to the diagnosis of COVID-19 disease, compared to the classical classification algorithms the ensemble learning method developed in this study showed better results. Soft Voting Ensemble and Hard Voting Ensemble were used as the ensemble method. From the experiments of these ensemble methods applied to real datasets, superior results are obtained compared to the separate algorithms.

Inspired by hawk eyes, the paper [23] proposed a hawk-eye-inspired perception algorithm of stereo vision. The shortcomings of binoculars in long-distance measurement and stable perception were addressed by the hawk-eye-inspired perception algorithm. With the hawk-eye-inspired perception algorithm, the UAV, which was equipped with binocular sensors, had completed the reconstruction of 3D point clouds for orchard navigation, and the number and quality of point clouds were significantly improved after reconstruction. This method retained the hardware conditions and achieved the goal of obtaining a high-quality 3D point cloud.

Since cloud computing cannot fully meet the needs and requirements of mobility with a wide range of functionality, fog computing has taken its place. Fog computing (FC) is a highly virtualized platform that connects cloud data centers and output devices by providing storage, computation, and networking services. FC is distinguished by the ability to process large amounts of data locally using installed software on heterogeneous hardware, as well as by dense geographical distribution with mobility support [6]. Location awareness and low latency edge location are features of FC, which include a large number of heterogeneous and decentralized devices that can communicate with one another to perform data storage and processing tasks. FC can provide better service quality in terms of fast response and low power consumption [6, 24].

FC is based on the use of network devices, or nodes, to process data collected from IoT devices while accounting for latency. In an FC environment, FC nodes are heterogeneous components that are deployed in an edge network. Gateways, switches, routers, access points with base stations, and dedicated servers are among the components. FC enables uniform resource management, including network computation and storage distribution [25]. FC must be used in conjunction with the establishment of a comprehensive hardware root of trust that extends to all processors and applications running on them, as well as to the cloud. The lack of a hardware root of trust increases the risk of a software fog infrastructure attack, allowing malware to exist undetected in the fog [24, 25, 26].

FC is viewed as an extension of cloud computing since it allows the cloud's computational and communication capabilities to be moved closer to the sensor nodes. The advantages of such moves are [27]:

- Reduced latency, which makes it easier to develop new IoT applications in real-time;
- Fog can allocate computing resources for large distributed sensor networks;
- Resource allocation improves mobility and location by delivering services to mobile and limited users;

- Fog is inherent in connecting devices in different physical environments allowing them to interact to deliver new services and functions.

Data encryption is critical in the transmission and processing of user data, which is exposed and poses a risk of data leakage to attackers [28]. Since component resources are limited, short-term security can only solve part of the problem. Low battery power and computation speed are resource constraints that are reflected in data encryption, where the use of standard cryptographic algorithms requires more power and reduces component lifetime. To this end, researchers have developed and implemented algorithms based on a lightweight cryptosystem with cryptographic algorithms. Papers [29, 30] propose a novel hybrid encryption scheme to improve healthcare data security in an IoT-enabled healthcare system. The study [31] presents a detailed review of existing watermarking techniques belonging to the spatial domain and frequency domain. In the paper [32], a novel cryptosystem is proposed using DNA cryptography and DNA steganography for the cloud-based IoT infrastructure. In this case, the confidential data is encrypted by using a long secret key. Then, it is hidden in an image. Thus, the proposed cryptosystem not only hides the data, but also encrypts the confidential data before storing it on the cloud server, and it resists many security attacks in the cloud-based IoT infrastructure. Authors of [33] showed that compared with traditional neural networks, an RBF neural network has the advantages of fast convergence and local approximation to eigenvalues when dealing with large-scale network topic data and does not suffer from the problems posed by local minimum. One approach is to modify the Advanced Encryption Standard (AES) algorithm [34, 35].

The motivation for this study. The adoption of AES algorithms is the most efficient and straightforward solution in terms of the security and complexity of IoT implementation. The distinctive aspect is the secret key distribution, which is challenging compared to other symmetric algorithms. A single block of data (128 bits) requires extensive computer work to encrypt or decode, which uses a lot of battery power. Because IoT components are constrained in their resources, using a lot of energy might shorten their lifespan [36]. This is the disadvantage of this system. Block encryption techniques must be in place for FC and edge computing of electronic devices and media to be stable. Data processing and storage on remote cloud servers are made more effective by the use of encryption to stop data leaking. The research gap between previous studies and the purpose of this study is to determine the performance evaluation of the proposed smart energy model based on fog and edge computing, as well as encryption algorithms for data transmission security.

The topical issue is to analyze and improve the security of data transfer from electronic devices and media in fog clouds using data encryption and encryption algorithms. The objectives of the study are the following:

1. To investigate fog and edge computing for data processing and storage in the cloud;

2. To study simple AES block encryption algorithms with the key lengths of 128, 256, and PRESENT 80, 128 bits for data encryption of FC for data processing and security with subsequent storage in the cloud.

The following part of the article includes Sections as 2 Methods and Materials (2.1 IoT Elements, 2.2 Fog and Edge Computing Architecture for IoT, 2.3 Algorithm-Based Data Encryption, 2.4 AES Encryption Algorithm, 2.5 PRESENT Encryption Algorithm); 3 Results (3.1 Comparison of AES and PRESENT Encryption Algorithm); 4 Discussion and 5 Conclusions.

2 MATERIAL AND METHODS

As an experimental setup, the interaction of smart devices in a two-room apartment using smartphones connected to WiFi and Bluetooth networks for signal processing in edge and fog computing was considered. Comparative characterization of the data processing speed for encryption algorithms with AES 128 and 256-bit key length, and for PRESENT 80- and 128-bit with 512 MB size is presented.

2.1 IoT Elements

An IoT device's hardware consists of a battery-powered sensor, an actuator, and a communication system. The sensor's function is to collect data from a specific environment. Flow rate, temperature, pressure, physical movements, distance, weight, and so on are examples of such data. The collected data is processed on the device and then transmitted via the communication network to remote servers [37, 38].

Sensors are IoT devices that collect and process data for transmission to display devices. Hence, sensors are the most energy-intensive devices in the IoT system. The only significant disadvantage of the complete IoT technology implementation is the device's limited autonomous operation time. A large amount of data is collected and then processed consuming high energy. The amount and accuracy of data are directly proportional to the battery's limited life.

2.2 Fog and Edge Computing Architecture for IoT

The comprehensive fog architecture is built on modern computing architecture and includes three major data transmission layers: cloud, fog, and edge. A backbone network connects the cloud and the fog to provide network services. The cloud is located at the top core layer, away from the edge devices. A fog is located in the middle layer, closer to the edge devices than the cloud, and each node in the fog is connected to the cloud and interconnected with one another. This suggests a link between fog computing, fog to the cloud, and fog to edge computing [39].

A cloud consists of high-performance servers and storage devices that are capable of the storage and analysis of large amounts of data. A cloud is a storage

center that allows for remote control and management and the processing of complex tasks. Sending data to the cloud is possible via high-speed wireless and wired communications. When a cloud is used for storage, it provides data storage meeting the needs of users while also performing intelligent data processing.

In contrast to the cloud, fog contains a network of interconnected fog computing nodes that allows for geo-distributed, low-latency data processing and transmission. Each fog node represents an ephemeral data storage center resource. Data collection, data loading and storage, computation, and management are the primary functions of the fog cloud, which facilitates network transformation [40].

Edge computing refers to a group of physical devices that are connected to one of the fog edges and have ubiquitous identification, sensing, and communication capabilities (vehicles, cars, appliances, cellular smartphones, and other electronic media) [39, 40]. Edge devices contain a large number of sensors and local data, which are transmitted to the cloud by end edge devices via the network, which has a high cost and time delay. The distinction between edge computing and FC is that FC is a highly virtualized platform that connects the end devices to cloud computing data centers to provide storage and networking services [41].

2.3 Algorithm-Based Data Encryption

Block encryption algorithms are used to secure and protect FC data, thereby improving the security of processed information between IoT devices [42]. Python and Matlab software with a built-in script library can be used to implement block encryption algorithms in software. Software implementation is used for mathematical optimization and orientation on encryption algorithm platform digit capacity determination. Hardware implementation, such as modern processors with powerful boards and video cards, facilitates calculation and information processing operations. The significant difference between the software and hardware implementations is the speed performance of the block algorithm when using different key lengths, for example, when using the AES algorithm with the key length of 128 bits, and the PRESENT algorithm with the key length of 80 bits.

2.4 AES Encryption Algorithm

The National Institute of Standards and Technology was the first to present an algorithm called the AES. The algorithm is a symmetric key block one. The study considers two key lengths: encryption and decryption require 10 rounds of data processing for 128-bit keys and 14 rounds of data processing for 256-bit keys. Symmetric block ciphers are used to protect and anonymize large amounts of data transmission, allowing and facilitating continuous data processing and transmission.

The rounds consist of three reversible transformations (layers), which are the following [35]:

- S-blocks that have optimal nonlinearity and can implement a nonlinear layer;

- The high degree of character interpenetration for calculating the communication masking block is provided by a linear mixing layer;
- The key addition layer is performed by encryption, which ensures that the first-round input is closed in case of an attack or encryption breaking on known data.

The AES encryption algorithm can run multiple rounds, each with multiple steps. These steps are as follows: The data block can transform from one stage to another before and after each step; each round must implement four inverted transformations; the final round must implement three transformations except the column mixing step. Figure 1 shows a general flow diagram of the AES encryption and decryption algorithm which consists of four steps that will be discussed further below [35, 42, 37].

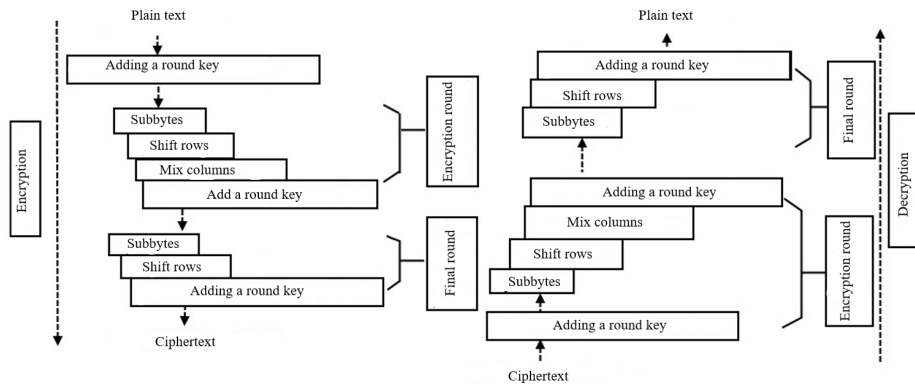

Figure 1. AES encryption and decryption algorithm general flow diagram. Source: [42].

Step 1. The sub-byte transformation, which is used in the encryption section, occurs in the first step. It is explained by non-linear byte substitution, which uses a substitution table to act on each status byte. The corresponding values from the lookup table are used to replace all 16 status byte cells. Figure 2 a) displays the sub-byte operation.

Step 2. The next step involves shifting the status bytes to the left in each row during encryption, which is accomplished through a row shift operation. The number of shifts is determined by the row number (e.g., 0, 1, 2, 3), which is a status matrix. Bytes such as 0 are not shifted, but rows 1–3 bytes are shifted to the left by 1–3 bytes. Figure 2 b) shows the row shift operation.

Step 3. In the next step, a column mixing transformation is performed. The transformation converts each status column into a new column. Such transformation represents a matrix multiplication of a status column by a constant square ma-

trix. The finite field is where all arithmetic operations are done. Figure 2 c) illustrates the column mixing operation.

Step 4. The next step involves adding a round key that works one column at a time, similar to column mixing. Adding the round keyword to each column's matrix explains the round key adding. The key addition step implies the matrix addition. Figure 2 d) shows the round key addition operation [42, 43, 44].

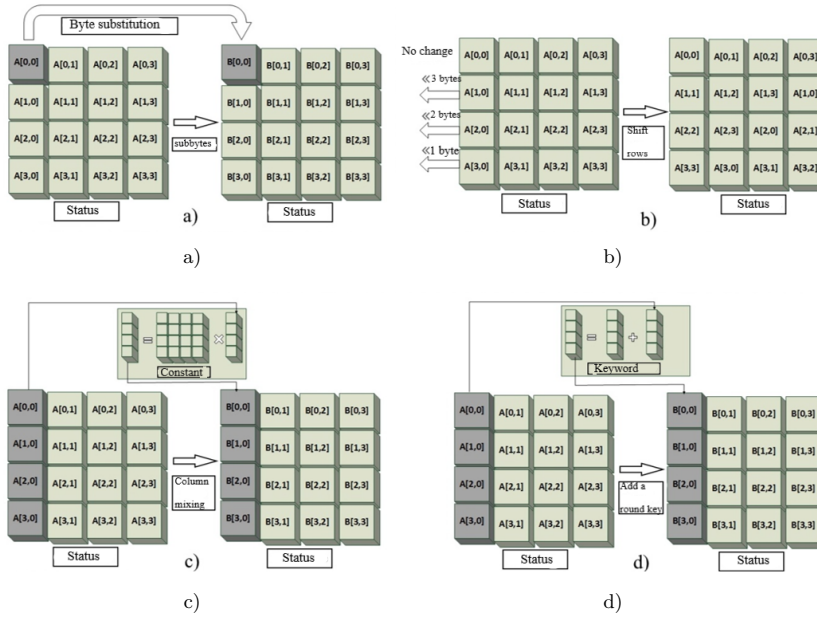

Figure 2. Basic steps of AES encryption and decryption algorithm general flow diagram. Source: [41, 43, 44].

Steps in Figure 2 a), 2 b), 2 c), and 2 d), respectively, are performed directly during the encryption and decryption process in all rounds except the final. The column mixing transformation operation is not carried out in the final round. The decryption process is based on the same structure that is used to encrypt the data. Transformation column mixing inversion is not performed in addition to the nine rounds.

2.5 PRESENT Encryption Algorithm

The PRESENT encryption algorithm is a lightweight block cipher based on the Substitution Permutation Network structure. Input consists of 64 bits of plain text and 80, 128 bits of the key. The cipher includes 31 encryption and decryption

rounds. Every round includes one substitution block and one permutation layer. A 64-bit key generated from the key register performs a plain text operation in each round. For hardware optimization, the algorithm employs 4-bit input and output substitution blocks. The only disadvantage of using the encryption algorithm is that it consumes a lot of power. The block diagram of the PRESENT encryption algorithm is shown in Figure 3 [45].

The following are the benefits of using the PRESENT encryption algorithm [46, 47]:

- Applications are in demand for security level control, allowing the use of an 80-bit key for an easy way of implementation;
- Applications do not involve the encryption of large amounts of data, making the space implementation better optimized and performant without making major changes;
- Physical volume is a priority for peak and average electricity consumption;
- Data encryption in devices that require the most efficient use of physical space is performed while adhering to the status control is used to encrypt and decrypt devices.

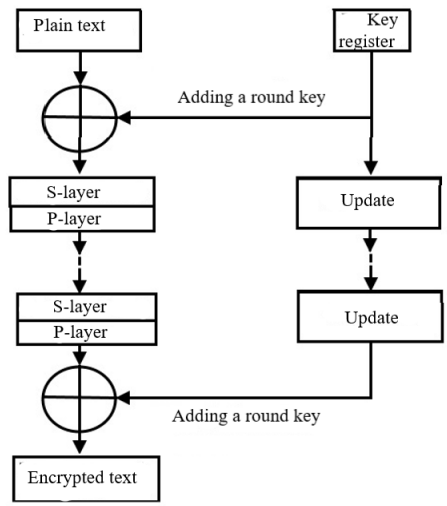

Figure 3. Encryption block diagram of PRESENT algorithm. Source: [47].

The PRESENT algorithm's encryption block diagram, shown in Figure 3, is used in highly specialized radio-frequency identification (RFID) devices and sensor networks. It is a suitable encryption algorithm for fog computing peripherals and electronic devices. The algorithm belongs to the class of compact crypto algorithms

and is intended for hardware implementation and evaluation. The hardware implementation of the PRESENT algorithm requires 2–3 times fewer logical elements than the AES algorithm. The mentioned peculiarity distinguishes the algorithm from the AES algorithm.

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S[x]$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

a) sBoxLayer

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$P(i)$	0	16	32	48	1	17	33	49	2	18	34	50	3	19	35	51
i	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
$P(i)$	4	20	36	52	5	21	37	53	6	22	38	54	7	23	39	55
i	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
$P(i)$	8	24	40	56	9	25	41	57	10	26	42	58	11	27	43	59
i	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
$P(i)$	12	28	44	60	13	29	45	61	14	30	46	62	15	31	47	63

b) pLayer

Figure 4. S-block layers a) and P-block layers b)

Figure 4 shows a block diagram of the S-block (Figure 4 a)) and P-block (Figure 4 b)) implementation of the PRESENT block encryption algorithm. Each of the 31 rounds begins with an XOR operation that introduces the key K_i for $1 < i < 32$ for a linear bitwise permutation and a nonlinear substitution layer. The nonlinear layer uses 4-bit S-blocks that can be applied in parallel 16 times in each round [48].

Adding a round key: Setting the round key $K_i = K^i_{63} \dots K^i_0$ where $1 < i < 32$ in the current status $b_{63} \dots b_0$. For the S-block layer, the current status $b_{63} \dots b_0$ is 16 blocks of 4-bit layers $w_{15} \dots w_0$, where $w_i = b_{4*i+3}||b_{4*i+2}||b_{4*i+1}||b_{4*i}$ for $0 < i < 15$. The bitwise permutation layer of P-blocks that is used in the algorithm is set by values (the i status bit is shifted to the position $P(i)$).

The PRESENT algorithm can use 80 and 128-bit keys. Using the 80-bit key as an example, the provided key is stored in the key register K. The key register is represented as $k_{79}k_{78} \dots k_0$ in the order up to 80. On each round, the 64-bit round key $K_i = k_{63}k_{62} \dots k_0$ includes 64 bits of the current contents of the key register K. Hence, the i^{th} round will be: $K_i = k_{63}k_{62} \dots k_0 = k_{79}k_{77} \dots k_{16}$ [46, 49].

3 RESULTS

Experimental Environment. A two-room apartment with digital smart devices and communication means was considered to investigate the effectiveness of FC with encryption algorithms as smart energy IoT, as shown in Figure 5. There were two bedrooms, an entrance hall, two bathrooms, a kitchen, a balcony, and a log-

gia in the apartment. There were the following IoT components in the apartment: a smart refrigerator, two Smart TVs, a coffee maker, smartphones, an electric stove, a smart washing machine, an air conditioner, and room lighting fixtures. The simulation experiment involved using Java language. The experimental environment was JDK 1.8.0, Eclipse 4.7, Bouncy Castle, and JPBC (2.0.0) library.

Figure 5. Smart electricity study based on a two-room apartment

The following step was to create an FC architecture for the IoT devices that can be connected to the network. Such devices should communicate with peripheral devices via a network of Wi-Fi, Bluetooth, and infrared transmission channels. Figure 6 illustrates how the IoT is linked to smartphones via smart technology, which helps device signals to be processed by FC. Data privacy is protected as a result of data processing using AES or PRESENT encryption algorithms, and data is then moved through FC services to cloud storage. Figure 6 illustrates smart grid devices that can make decisions autonomously, without human intervention.

Three smartphones were used to control smart IoT devices, as shown in Figure 6. Sensors were the items that can be monitored and configured using smartphone apps. AES encryption algorithms improved device security and data storage, preventing data leakage. The possibility of connecting devices with smartphones via WiFi and Bluetooth signal connections was an important issue to consider as well. These connections had waves and an operating range that allowed devices to be tracked. The characteristics of the IoT are shown in Table 1 by device type and power consumption.

Table 1 shows that each device connected to the network consumes energy independently, creating a load for the power grid's operation. Some devices have a power-saving mode that reduces power consumption and allows for greater power savings. The refrigerator (0.6–1.0 kW/day) and air conditioner (0.71–0.72 kW/day) consume the most power. NOUS Smart Wi-Fi Bulb P4 4.5-5 W has the lowest

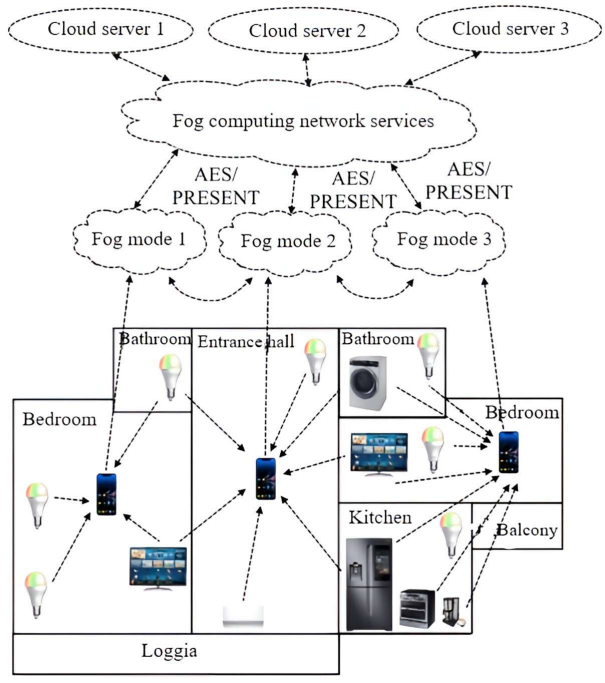

Figure 6. Connecting smart devices with fog computing data processing and storage

Device and Equipment	**Connection Type and Data Transfer**	**Power Consumption**
4K Smart QLED TV Samsung QE55Q80TAUXRU	Ethernet, Bluetooth 4.2 LE, WiFi (2.4 and 5 GHz)	240 W maximum 68.5 W power saving
Cooper & Hunter CH-S09FTXE-NG Air Conditioner	WiFi	0.71–0.72 kW
Samsung Galaxy M52 smartphone	4G, 5G, Bluetooth 5.0, WiFi, NFC	
NOUS Smart WiFi Bulb P4	WiFi	4.5–5 W 220–240 V
LG GN-H702HMHZ refrigerator	WiFi (ThinQ App)	0.6–1.0 kW/
REDMOND Coffee Machine SCYCOFFEE RCM-M1505S-E	Bluetooth 4.0	0.6 kW 220/240 V
Viomi Internet Smart Gas Stove Power 5.2	Bluetooth 4.0	5.2 kW
Samsung WD80T554CBT washing machine	WiFi	5.44 kW

Table 1. Internet of Things characteristics

power consumption. When it comes to IoT devices' power consumption and data security, AES and PRESENT block encryption algorithms are used as performance evaluation tools for edge and FC concepts.

3.1 Comparison of AES and PRESENT Encryption Algorithms

Table 2 lists the technical characteristics of block encryption algorithms as well as the parameters used to implement them. Algorithm implementation includes both software and hardware implementation, resulting in more efficient data transmission equipment and devices. The key length and block sizes of each algorithm differ, indicating the range of processing large and medium data values of the processed information. Correspondingly, the larger key size consumes more resources; for example, using 256-bit AES encryption on a smartphone drains the battery much faster than 128-bit encryption.

Implementation Parameters and Conditions	**Algorithms**	
	Advanced Encryption Standard	**PRESENT**
Algorithm implementation	Software and hardware	Software and hardware
Key length	128/256 bit	80/128 bit
Block size	128 bit	64 bit
Number of rounds	10//14	31
Encryption speed	4–5 Mbytes/sec at 2 GHz	1 Mbyte/s at 2 GHz
RAM requirement	4 640 bytes	1 000 bytes
Implementation memory	160 bytes	18 bytes

Table 2. Comparative characteristics of AES and PRESENT block cipher algorithms

Depending on the key length and block size, different encryption algorithms have varying encryption speeds. Figure 7 shows that the longer key length consumes more resources, resulting in an additional load due to data processing. The figure illustrates that the 256-bit AES key length to compute 512 Mb of information takes more resources; however, it can process a large amount of information, unlike the 80-bit PRESENT key. When processing 512 Mb of data, the 128-bit key lengths for AES and PRESENT at 160–220 Mb/s are nearly identical.

The lowest speed measured in these tests ranges from 135 to 160 MB/s, which is the typical speed of modern storage hard drives. The average speed is 170–200 MB/s, and the maximum speed is 205–230 MB/s. Thus, based on Figure 7, it is possible to conclude that the AES block encryption algorithm outperforms the PRESENT algorithm when processing large data sets.

Based on the data given in Tables 1 and 2, as well as in Figure 7, we can conclude that the use of the PRESENT algorithm developed by us allows us to process much more information with a shorter key length, which allows us to spend less electricity.

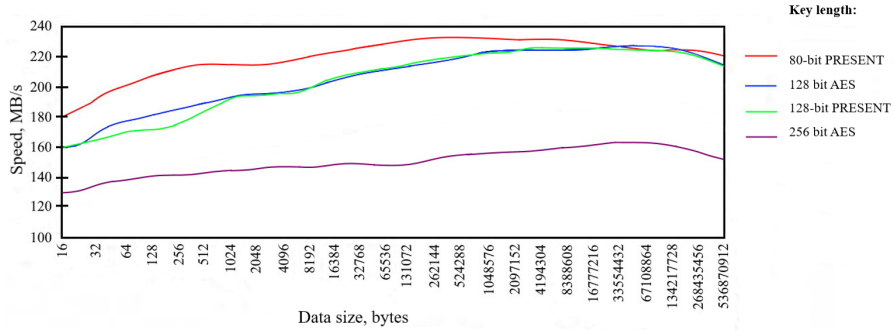

Figure 7. Comparison of AES and PRESENT processing speed

4 DISCUSSION

Many new innovative technologies in today's world allow devices to communicate with each other and exchange information with certain limitations. Using switching communication networks, the IoT enables many devices with limited resources and constraints to exchange data, compute processes, and make decisions. Many challenges exist in heterogeneous IoT environments, including device power consumption, battery power constraints, memory capacity, vendor cost, and security in information and communication technology networks [50, 51].

Information security necessitates the use of highly secure encryption algorithms that are simple to implement. One such example is the Secure and Fast Encryption Routine (SAFER) [52]. This block encryption method considers complicacy and complexity in computational operations and can be used for IoT devices with limited capacity. The study by Panahi et al. [53] employs the block decryption algorithm Skipjack for wireless sensor networks (WSN) data transmission. The study describes the algorithm's low efficiency in embedded IoT devices as well as numerous implementation issues. The Skipjack algorithm encrypts and decrypts 64-bit blocks of data using an 80-bit key and is intended to be used in the Clipper chip to protect audio data transmitted over the telephone, mobile, and wireless networks. The HIGH algorithm, which is useful for omnipresent computing devices, is described in Caruso et al. [54] for wireless sensor systems or network devices. The Feistel network and basic operations are used in this block algorithm, which has 32 rounds, a 128-bit key, and a 64-bit block size.

Based on the trust model, Appavoo et al. [55] propose lightweight functional encryption to protect and preserve the privacy of data and information. In the presence of distrusted parties, this approach seeks to minimize privacy loss and secondary use. The approach is based on a unification solution that employs device smoothing to conceal the sensing source's identity and a pre-computation initialization vector. The latter extracts trigger information only for the appropriate services of the relevant trusted parties. If published sensor readings and trigger informa-

tion are not available, untrusted parties are not able to access end-user information. The service provider is unable to identify the sensor which the data originates from, and the unification scheme precludes determining whether the trigger is active or not.

Tso et al. [56] describe the use of plain text key cryptographic encryption techniques to prevent data leakage in healthcare systems. To prevent data disclosure due to internal attacks, the authors developed a practical approach based on a secure multilateral fair play structure. The proposed method enables software developers to easily interpret and implement security protocols in distributed IoT systems with multiple device connections. This method provides an initial setup that requires each IoT node to store a single secret key before applying it to external data servers, an ideal solution for the limited storage capacity of IoT nodes.

Pérez et al. [57] developed a novel architecture that combines encryption flexibility and speed using attribute-based cryptography approaches and AES symmetric encryption algorithms. The use of this architecture safeguards participant devices' privacy while facilitating secure data sharing. As a result, data sources outsource attribute-based cryptographic operations to a reliable proxy, and the IoT cloud handles encryption key management and file processing.

Two block encryption techniques were discussed in the present study: PRESENT with 80 and 128-bit keys and AES with 128 and 256-bit keys. The comparison revealed that the suggested algorithms can process information from smaller to bigger amounts of data with high processing speed. Data may be protected and stored on distant servers using cloud-based encryption and coding for smart energy carriers (devices, apparatus, and equipment).

Receiving data is converted into an encrypted form by the AES algorithm, which also turns the text into a random string of letters. This allows encrypted data to be retained for a long period. The PRESENT method is an efficient software and hardware solution that can handle data volumes ranging from tiny to extremely big, albeit it is less relevant to IoT technology.

Moreover, we added Table 3 with a comparison of the modern encryption techniques based on available features [58].

The discussed encryption techniques for cloud data security are used according to the user requirements; however, there are some common parameters based on which we can compare these techniques to make selection easy for a novice cloud user. All algorithms discussed earlier perform encryption of cloud data in different scenarios. These algorithms are classified broadly into two categories:

1. symmetric techniques and
2. asymmetric encryption techniques.

Therefore, ranking any algorithm into low or high classes is unfair. Symmetric encryption techniques use the same keys for encryption and decryption, whereas asymmetric techniques use different encryption and decryption keys.

Approach Used	Methodology	Encryption End	Encryption Time	Type of Encryption
PSO, PPSO, BFD, GA, ACO	Optimization procedure	Nil	Nil	Optimization mechanisms
AES	Media Access Control (MAC) address with key size 1 024	Both-server and client-side	Low	Symmetric
Fully Homomorphic Encryption (FHE)	Fully Homomorphic Encryption (FHE) has been used to carry out analytical tasks on encrypted data	Client-side	Low	Symmetric
AES	CSPs use encryption and other techniques to preserve the privacy of client's critical information	Client-side	High	Symmetric
Attribute-based encryption	Use of attributes-based encryption (ABE)	Both server and client-side	High	Asymmetric

Table 3. Comparison of the modern encryption techniques based on available features

5 CONCLUSION

This article discusses the computational and network communication technologies for the Internet of Things with a focus on FC principles and the use of the AES and PRESENT block encryption algorithms. The study came to the conclusion that due to the secure operation of IoT devices employing fog computing, there were concerns about the security of the processed data. Block encryption techniques were used to improve the efficiency of data processing and storage on cloud servers. They stop the leakage of user data from media and technological devices.

As an experimental setup, the interaction of smart devices in a two-room apartment using smartphones connected to WiFi and Bluetooth networks for signal pro-

cessing in edge and fog computing was considered. Three smartphones were used to control smart IoT devices. Sensors are the items that can be monitored and configured using smartphone apps. The possibility of connecting devices with smartphones via WiFi and Bluetooth signal connections was an important issue to consider as well. These connections had waves and an operating range that allowed devices to be tracked.

The study highlights two block encryption methods: AES with 128 and 256-bit keys and PRESENT with 80 and 128-bit keys. The two methods were compared, and it became clear that the recommended algorithms could handle data quickly from smaller to larger volumes. Cloud-based encryption and coding for smart energy carriers can be used to safeguard data and store it on remote servers (devices, apparatus, and equipment).

By transforming the text into a random string of letters, the AES algorithm turns incoming data into an encrypted form. This makes it possible to store encrypted data for a long time. The PRESENT technique is capable of managing data volumes ranging from very small to extremely large. However, it is a less suitable encryption technique for IoT systems. But it is a good method for both software and hardware implementation. The algorithms in this article are crucial to every aspect of the smart grid and IoT, including healthcare, smart electricity, smart city systems, etc.

The prospect of our research is to expand our results for a more complex residential system, for example, an administrative or sports building with more devices, apparatus, or equipment.

REFERENCES

[1] Gaber, M. M.—Aneiba, A.—Basurra, S.—Batty, O.—Elmisery, A. M.—Kovalchuk, Y.—Rehman, M. H. U.: Internet of Things and Data Mining: From Applications to Techniques and Systems. WIREs Data Mining and Knowledge Discovery, Vol. 9, 2019, No. 3, Art. No. e1292, doi: 10.1002/widm.1292.

[2] Raval, M.—Bhardwaj, S.—Aravelli, A.—Dofe, J.—Gohel, H.: Smart Energy Optimization for Massive IoT Using Artificial Intelligence. Internet of Things, Vol. 13, 2021, Art. No. 100354, doi: 10.1016/j.iot.2020.100354.

[3] Renugadevi, N.—Saravanan, S.—Sudha, C. N.: IoT Based Smart Energy Grid for Sustainable Cites. Materials Today: Proceedings, Vol. 81, 2023, No. 2, pp. 98–104, doi: 10.1016/j.matpr.2021.02.270.

[4] Verma, R.—Kumari, A.—Anand, A.—Yadavalli, V. S. S.: Revisiting Shift Cipher Technique for Amplified Data Security. Journal of Computational and Cognitive Engineering, Vol. 3, 2024, No. 1, pp. 8–14, doi: 10.47852/bonviewJCCE2202261.

[5] Gheisari, M.—Liu, Y.—Hajiaghai, H.—Rezaei, R.—Khodabakhshi-Javinani, N.—Khammar, S.: A New Security Alarm Based on Interaction. Journal of Global Humanities and Social Sciences, Vol. 5, 2024, No. 8, pp. 300–303, doi: 10.61360/BoniGHSS242016870801.

[6] Taha, M. Y.—Kurnaz, S.—Ibrahim, A. A.—Mohammed, A. H.—Raheem, S. A.—Namaa, H. M.: Internet of Things and Cloud Computing – A Review. 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), IEEE, 2020, pp. 1–7, doi: 10.1109/ISMSIT50672.2020.9254340.

[7] Okay, F. Y.—Ozdemir, S.: A Fog Computing Based Smart Grid Model. 2016 International Symposium on Networks, Computers and Communications (ISNCC), IEEE, 2016, pp. 1–6, doi: 10.1109/ISNCC.2016.7746062.

[8] Tong, Z.—Ye, F.—Yan, M.—Liu, H.—Basodi, S.: A Survey on Algorithms for Intelligent Computing and Smart City Applications. Big Data Mining and Analytics, Vol. 4, 2021, No. 3, pp. 155–172, doi: 10.26599/BDMA.2020.9020029.

[9] Hajjaji, Y.—Boulila, W.—Farah, I. R.—Romdhani, I.—Hussain, A.: Big Data and IoT-Based Applications in Smart Environments: A Systematic Review. Computer Science Review, Vol. 39, 2021, Art. No. 100318, doi: 10.1016/j.cosrev.2020.100318.

[10] Meneghello, F.—Calore, M.—Zucchetto, D.—Polese, M.—Zanella, A.: IoT: Internet of Threats? A Survey of Practical Security Vulnerabilities in Real IoT Devices. IEEE Internet of Things Journal, Vol. 6, 2019, No. 5, pp. 8182–8201, doi: 10.1109/JIOT.2019.2935189.

[11] Panchalingam, R.—Chan, K. C.: A State-of-the-Art Review on Artificial Intelligence for Smart Buildings. Intelligent Buildings International, Vol. 13, 2021, No. 4, pp. 203–226, doi: 10.1080/17508975.2019.1613219.

[12] Zhou, X.—Liang, W.—Wang, K. I. K.—Wang, H.—Yang, L. T.—Jin, Q.: Deep-Learning-Enhanced Human Activity Recognition for Internet of Healthcare Things. IEEE Internet of Things Journal, Vol. 7, 2020, pp. 6429–6438, doi: 10.1109/JIOT.2020.2985082.

[13] Malek, Y. N.—Najib, M.—Bakhouya, M.—Essaaidi, M.: Multivariate Deep Learning Approach for Electric Vehicle Speed Forecasting. Big Data Mining and Analytics, Vol. 4, 2021, No. 1, pp. 56–64, doi: 10.26599/BDMA.2020.9020027.

[14] Tseng, C. H.—Lin, C.—Chang, H. C.—Liu, C. C.—Serafico, B. M. F.—Wu, L. C.—Lin, C. T.—Hsu, T.—Huang, C. Y.—Lo, M. T.: Cloud-Based Artificial Intelligence System for Large-Scale Arrhythmia Screening. Computer, Vol. 52, 2019, No. 11, pp. 40–51, doi: 10.1109/MC.2019.2933195.

[15] Zhou, X.—Xu, X.—Liang, W.—Zeng, Z.—Yan, Z.: Deep-Learning-Enhanced Multitarget Detection for End-Edge-Cloud Surveillance in Smart IoT. IEEE Internet of Things Journal, Vol. 8, 2021, No. 16, pp. 12588–12596, doi: 10.1109/JIOT.2021.3077449.

[16] Sandhu, A. K.: Big Data with Cloud Computing: Discussions and Challenges. Big Data Mining and Analytics, Vol. 5, 2022, No. 1, pp. 32–40, doi: 10.26599/BDMA.2021.9020016.

[17] He, Z.—Zhou, J.: Inference Attacks on Genomic Data Based on Probabilistic Graphical Models. Big Data Mining and Analytics, Vol. 3, 2020, No. 3, pp. 225–233, doi: 10.26599/BDMA.2020.9020008.

[18] Chen, H.—Zhang, Y.—Cao, Y.—Xie, J.: Security Issues and Defensive Ap-

proaches in Deep Learning Frameworks. Tsinghua Science and Technology, Vol. 26, 2021, No. 6, pp. 894–905, doi: 10.26599/TST.2020.9010050.

[19] Ahmed, U.—Lin, J. C. W.—Srivastava, G.: A Resource Allocation Deep Active Learning Based on Load Balancer for Network Intrusion Detection in SDN Sensors. Computer Communications, Vol. 184, 2022, pp. 56–63, doi: 10.1016/j.comcom.2021.12.009.

[20] Wani, A.—Rubeena Khaliq, R. S.: SDN-Based Intrusion Detection System for IoT Using Deep Learning Classifier (IDSIoT-SDL). CAAI Transactions on Intelligence Technology, Vol. 6, 2021, No. 3, pp. 281–290, doi: 10.1049/cit2.12003.

[21] Abdullayeva, F. J.: Internet of Things-Based Health Care System on Patient Demographic Data in Health 4.0. CAAI Transactions on Intelligence Technology, Vol. 7, 2022, No. 4, pp. 644–657, doi: 10.1049/cit2.12128.

[22] Chen, Z.: Research on Internet Security Situation Awareness Prediction Technology Based on Improved RBF Neural Network Algorithm. Journal of Computational and Cognitive Engineering, Vol. 1, 2022, No. 3, pp. 103–108, doi: 10.47852/bonviewJCCE149145205514.

[23] Zhang, Z.—Chen, J.—Xu, X.—Liu, C.—Han, Y.: Hawk-Eye-Inspired Perception Algorithm of Stereo Vision for Obtaining Orchard 3D Point Cloud Navigation Map. CAAI Transactions on Intelligence Technology, Vol. 8, 2023, No. 3, pp. 987–1001, doi: 10.1049/cit2.12141.

[24] Mahmud, R.—Buyya, R.: Modelling and Simulation of Fog and Edge Computing Environments Using iFogSim Toolkit. In: Buyya, R., Srirama, S. N. (Eds.): Fog and Edge Computing: Principles and Paradigms. John Wiley & Sons, Wiley Series on Parallel and Distributed Computing, 2019, pp. 433–461, doi: 10.1002/9781119525080.ch17.

[25] Firouzi, F.—Chakrabarty, K.—Nassif, S.: Intelligent Internet of Things: From Device to Fog and Cloud. Springer, 2020, doi: 10.1007/978-3-030-30367-9.

[26] Muthanna, A.—Ateya, A. A.—Khakimov, A.—Gudkova, I.—Abuarqoub, A.—Samouylov, K.—Koucheryavy, A.: Secure and Reliable IoT Networks Using Fog Computing with Software-Defined Networking and Blockchain. Journal of Sensor and Actuator Networks, Vol. 8, 2019, No. 1, Art. No. 15, doi: 10.3390/jsan8010015.

[27] Suárez-Albela, M.—Fernández-Caramés, T. M.—Fraga-Lamas, P.—Castedo, L.: A Practical Evaluation of a High-Security Energy-Efficient Gateway for IoT Fog Computing Applications. Sensors, Vol. 17, 2017, No. 9, Art. No. 1978, doi: 10.3390/s17091978.

[28] Pour, M. S.—Nader, C.—Friday, K.—Bou-Harb, E.: A Comprehensive Survey of Recent Internet Measurement Techniques for Cyber Security. Computers & Security, Vol. 128, 2023, Art. No. 103123, doi: 10.1016/j.cose.2023.103123.

[29] Das, S.—Namasudra, S.: A Novel Hybrid Encryption Method to Secure Healthcare Data in IoT-Enabled Healthcare Infrastructure. Computers and Electrical Engineering, Vol. 101, 2022, Art. No. 107991, doi: 10.1016/j.compeleceng.2022.107991.

[30] Das, S.—Namasudra, S.: MACPABE: Multi-Authority-Based CP-ABE with Efficient Attribute Revocation for IoT-Enabled Healthcare Infrastructure. Interna-

tional Journal of Network Management, Vol. 33, 2023, No. 3, Art. No. e2200, doi: 10.1002/nem.2200.

[31] Singh, R.—Saraswat, M.—Ashok, A.—Mittal, H.—Tripathi, A.—Pandey, A. C.—Pal, R.: From Classical to Soft Computing Based Watermarking Techniques: A Comprehensive Review. Future Generation Computer Systems, Vol. 141, 2023, pp. 738–754, doi: 10.1016/j.future.2022.12.015.

[32] Namasudra, S.: A Secure Cryptosystem Using DNA Cryptography and Stenography for the Cloud-Based IoT Infrastructure. Computers and Electrical Engineering, Vol. 104, Part A, 2022, Art. No. 108426, doi: 10.1016/j.compeleceng.2022.108426.

[33] Liu, Y.—Zhao, J.—Xiao, Y.: C-RBFNN: A User Retweet Behavior Prediction Method for Hotspot Topics Based on Improved RBF Neural Network. Neurocomputing, Vol. 275, 2018, pp. 733–746, doi: 10.1016/j.neucom.2017.09.015.

[34] Arpaia, P.—Bonavolontá, F.—Cioffi, A.: Problems of the Advanced Encryption Standard in Protecting Internet of Things Sensor Networks. Measurement, Vol. 161, 2020, Art. No. 107853, doi: 10.1016/j.measurement.2020.107853.

[35] Sultan, I.—Mir, B. J.—Banday, M. T.: Analysis and Optimization of Advanced Encryption Standard for the Internet of Things. 2020 7th International Conference on Signal Processing and Integrated Networks (SPIN), IEEE, 2020, pp. 571–575, doi: 10.1109/SPIN48934.2020.9071380.

[36] Chhaya, L.—Sharma, P.—Kumar, A.—Bhagwatikar, G.: IoT-Based Implementation of Field Area Network Using Smart Grid Communication Infrastructure. Smart Cities, Vol. 1, 2018, No. 1, pp. 176–189, doi: 10.3390/smartcities1010011.

[37] Pan, J.—Jain, R.—Paul, S.—Vu, T.—Saifullah, A.—Sha, M.: An Internet of Things Framework for Smart Energy in Buildings: Designs, Prototype, and Experiments. IEEE Internet of Things Journal, Vol. 2, 2015, No. 6, pp. 527–537, doi: 10.1109/JIOT.2015.2413397.

[38] Zhang, P.—Zhou, M.—Fortino, G.: Security and Trust Issues in Fog Computing: A Survey. Future Generation Computer Systems, Vol. 88, 2018, pp. 16–27, doi: 10.1016/j.future.2018.05.008.

[39] Weber, D.—Schilling, C.—Wisselink, F.: Low Power Wide Area Networks: The Game Changer for the Internet of Things. In: Krüssel, P. (Ed.): Future Telco: Successful Positioning of Network Operators in the Digital Age. Springer, Cham, Management for Professionals, 2019, pp. 175–185, doi: 10.1007/978-3-319-77724-5_15.

[40] Yakubu, J.—Abdulhamid, S. M.—Christopher, H. A.—Chiroma, H.—Abdullahi, M.: Security Challenges in Fog-Computing Environments: A Systematic Appraisal of Current Developments. Journal of Reliable Intelligent Environments, Vol. 5, 2019, No. 4, pp. 209–233, doi: 10.1007/s40860-019-00081-2.

[41] Singh, S.—Sharma, P. K.—Moon, S. Y.—Park, J. H.: Advanced Lightweight Encryption Algorithms for IoT Devices: Survey, Challenges and Solutions. Journal of Ambient Intelligence and Humanized Computing, Vol. 15, 2024, No. 2, pp. 1625–1642, doi: 10.1007/s12652-017-0494-4.

[42] Chowdhury, A. R.—Mahmud, J.—Kamal, A. R. M.—Hamid, M. A.: MAES: Modified Advanced Encryption Standard for Resource Constraint Environments. 2018 IEEE Sensors Applications Symposium (SAS), 2018, pp. 1–6, doi:

10.1109/SAS.2018.8336747.

[43] Nurmalasari, D.—Mulyana, E.—Irfan, M.: Security Implementation of the Internet of Things Using the Advanced Encryption Standard (AES) Algorithm. 2019 IEEE 5th International Conference on Wireless and Telematics (ICWT), 2019, pp. 1–4, doi: 10.1109/ICWT47785.2019.8978232.

[44] Su, N.—Zhang, Y.—Li, M.: Research on Data Encryption Standard Based on AES Algorithm in Internet of Things Environment. 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2019, pp. 2071–2075, doi: 10.1109/ITNEC.2019.8729488.

[45] Chatterjee, R.—Chakraborty, R.: A Modified Lightweight PRESENT Cipher for IoT Security. 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), IEEE, 2020, pp. 1–6, doi: 10.1109/ICCSEA49143.2020.9132950.

[46] Cho, W. L.—Kim, K. B.—Shin, K. W.: A Hardware Design of Ultra-Lightweight Block Cipher Algorithm PRESENT for IoT Applications. Journal of the Korea Institute of Information and Communication Engineering, Vol. 20, 2016, No. 7, pp. 1296–1302, doi: 10.6109/jkiice.2016.20.7.1296 (in Korean).

[47] Hu, C.—Bao, W.—Wang, D.: IoT Communication Sharing: Scenarios, Algorithms and Implementation. IEEE INFOCOM 2018 – IEEE Conference on Computer Communications, 2018, pp. 1556–1564, doi: 10.1109/INFOCOM.2018.8486329.

[48] Abdulraheem, A. N.—Nema, B. M.: Secure IoT Model Based on PRESENT Lightweight Modified and Chaotic Key Generator. 2020 1st Information Technology to Enhance e-Learning and Other Application (IT-ELA), IEEE, 2020, pp. 12–18, doi: 10.1109/IT-ELA50150.2020.9253079.

[49] Bogdanov, A.—Knudsen, L. R.—Leander, G.—Paar, C.—Poschmann, A.—Robshaw, M. J.—Seurin, Y.—Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (Eds.): International Workshop on Cryptographic Hardware and Embedded Systems. Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 4727, 2007, pp. 450–466, doi: 10.1007/978-3-540-74735-2_31.

[50] Selvaraj, J.—Lai, W. C.—Kavin, B. P.—Seng, G. H.: Cryptographic Encryption and Optimization for Internet of Things Based Medical Image Security. Electronics, Vol. 12, 2023, No. 7, Art. No. 1636, doi: 10.3390/electronics12071636.

[51] Beketaeva, A. O.—Naimanova, A. Z.—Shakhan, N.—Zadauly, A.: Simulation of the Shock Wave Boundary Layer Interaction in Flat Channel with Jet Injection. ZAMM – Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik, Vol. 103, 2023, No. 8, Art. No. e202200375, doi: 10.1002/zamm.202200375.

[52] Guo, X.—Hua, J.—Zhang, Y.—Wang, D.: A Complexity-Reduced Block Encryption Algorithm Suitable for Internet of Things. IEEE Access, Vol. 7, 2019, pp. 54760–54769, doi: 10.1109/ACCESS.2019.2912929.

[53] Panahi, P.—Bayilmiş, C.—Çavuşoğlu, U.—Kaçar, S.: Performance Evaluation of Lightweight Encryption Algorithms for IoT-Based Applications. Arabian Journal for Science and Engineering, Vol. 46, 2021, No. 4, pp. 4015–4037, doi:

10.1007/s13369-021-05358-4.

[54] Caruso, A.—Chessa, S.—Escolar, S.—del Toro, X.—López, J. C.: A Dynamic Programming Algorithm for High-Level Task Scheduling in Energy Harvesting IoT. IEEE Internet of Things Journal, Vol. 5, 2018, No. 3, pp. 2234–2248, doi: 10.1109/JIOT.2018.2828943.

[55] Appavoo, P.—Chan, M. C.—Bhojan, A.—Chang, E. C.: Efficient and Privacy-Preserving Access to Sensor Data for Internet of Things (IoT) Based Services. 2016 8th International Conference on Communication Systems and Networks (COMSNETS), IEEE, 2016, pp. 1–8, doi: 10.1109/COMSNETS.2016.7439941.

[56] Tso, R.—Alelaiwi, A.—Mizanur Rahman, S. M.—Wu, M. E.—Shamim Hossain, M.: Privacy-Preserving Data Communication Through Secure Multi-Party Computation in Healthcare Sensor Cloud. Journal of Signal Processing Systems, Vol. 89, 2017, No. 1, pp. 51–59, doi: 10.1007/s11265-016-1198-2.

[57] Pérez, S.—Rotondi, D.—Pedone, D.—Straniero, L.—Núñez, M. J.—Gigante, F.: Towards the CP-ABE Application for Privacy-Preserving Secure Data Sharing in IoT Contexts. In: Barolli, L., Enokido, T. (Eds.): Innovative Mobile and Internet Services in Ubiquitous Computing. Springer, Cham, Advances in Intelligent Systems and Computing, Vol. 612, 2017, pp. 917–926, doi: 10.1007/978-3-319-61542-4_93.

[58] Qureshi, M. B.—Qureshi, M. S.—Tahir, S.—Anwar, A.—Hussain, S.—Uddin, M.—Chen, C. L.: Encryption Techniques for Smart Systems Data Security Offloaded to the Cloud. Symmetry, Vol. 14, 2022, No. 4, Art. No. 695, doi: 10.3390/sym14040695.

Wei Wu has a Master of Engineering degree. He is Teacher of the School of Internet of Things Technology at Wuxi Institute of Technology, WuXi, China. His research interests include Internet of Things, smart grids, Advanced Encryption Standard and fog computing.