scientific reports

Securing Internet of Things Devices Using a Hybrid Approach

R. Sherine Jenny[1], N. Sugirtham[1], B. Thiyaneswaran[2], S. Kumarganesh[3], K. Martin Sagayam[4] , Syed Immamul Ansarullah[5], Farhan Amin[6]*, Isabel de la Torre Díez[7]*, Carlos Osorio García[8] and Alina Eugenia Pascual Barrera[8]

[1]Department of ECE, Dr. Mahalingam College of Engineering and Technology, Coimbatore, Tamil Nadu, India
[2]Department of ECE, Sona College of Technology, Salem, Tamil Nadu, India
[3]Department of ECE, Knowledge Institute of Technology, Salem, Tamil Nadu, India
[4]Division of ECE, Karunya Institute of Technology and Sciences, Coimbatore, Tamil Nadu, India
[5]Department of Management Studies, University of Kashmir, North Campus, Delina, Baramulla 193103, Jammu & Kashmir, India
[6]School of Computer Science and Engineering, Yeungnam University, Gyeongsan 38541, Republic of Korea
[7]Department of Signal Theory and Communications, University of Valladolid, Valladolid, Spain
[8]Universidad Europea del Atlántico, Isabel Torres 21, 39011 Santander, Spain
*farhanamin10@hotmail.com, isator@uva.es

Abstract

With increased Internet of Things (IoT) devices, complexity and protection are more challenging. Lightweight cryptographic algorithms are secure and suitable for limited-resource environments; however, their hash functions provide encrypted data but not integrity. Strong security features are available, but setup is difficult and expensive. Network security mechanisms increase power consumption and latency. As IoT networks grow, managing cryptographic keys and securely authenticating large numbers of devices become complex tasks. Efficient key management strategies are required to ensure the scalability required. Existing state-of-the-art solutions lack standardization, scalability, complex and costly. Thus, this research proposes a secure solution for IoT resource-constrained devices, combining strong data

scientific reports

integrity and lightweight encryption, and is thus named a hybrid. This hybrid approach integrates SHA-512 and the PRESENT cipher in our proposed approach and thus ensuring higher security than state-of-the-art models. This intelligent combination not only enhances the algorithm's resistance against cryptographic attacks but also improves its processing speed. The proposed approach is used to reduce the processing time for encryption in the IoT platform and to preserve the trade-off between security and efficiency. In terms of memory use, execution time, and precision, the proposed approach is compared with recent state-of-the-art research. The experimental results indicate that our approach is efficient using the avalanche, authentication success rate, collision events, and execution time. The efficiency is 53% to 65%, and the avalanche effect indicates sensitivity to input variations, suggesting moderate-to-considerable reactivity to small data changes. The experimental tests conducted across 10,000 and 80,000 runs reveal no collisions and found that the proposed approach is resilient in managing unique IDs. Moreover, our approach performs consistently, with an average execution time of 0.088246 s, ranging from 0.075954 to 0.094583 s. Finally, our approach provides a practical and scalable solution for securing IoT devices in resource-constrained environments, addressing practical problems for IoT devices.

Keywords: Internet of Things, lightweight cipher authentication, avalanche, collision, integrity

Introduction

The Internet of Things (IoT) is a transformative technology that enables physical objects to connect and exchange data via the internet. According to a survey in 2022, [1] estimated that there were 14.4 billion active endpoints despite the recent chip shortage. Moreover, they forecast approximately 30 billion connected devices by 2027. This massive number of devices facilitates seamless data sharing and access to services worldwide, fundamentally altering how people interact with technology. These devices are becoming increasingly popular even in homes, with around 40% of households worldwide having an IoT device. According to a survey in North America, this number is approximately 70% [1]. However, the rapid growth of IoT devices has introduced significant security challenges, such as scalability, complexity, computing power, size, and resource constraints. These interconnected devices, equipped with sensors that capture data from their surroundings, generate vast volumes of data and information. Many IoT devices suffer from intrinsic security vulnerabilities due to limited computing power, low-cost design, and a lack of security [1]. The main function of IoT devices is to collect various types of data sensed through different sensors. According to a survey, the most important concern is the data security for today’s IoT users [11]. Therefore, integration of many features in terms of security for both hardware and software is utmost important to enhance the security of IoT devices and to make sure that the vital sensed data is protected. For this reason, applying the concept of cryptography is a good option for IoT devices’ physical constraints. Before sending to the cloud server at the back end, cryptographic algorithms are used in IoT devices by using authentication mechanisms, like in [12], to maintain the integrity of data and prevent hackers from sniffing the data. Although data security based on cryptography is finding importance at a very increasing rate, it is still more of a challenge to fit various cryptographic algorithms with huge power and area overheads in an IoT device. Efficient and suitable key management is required for the encryption of IoT devices, as security is compromised if the key management is poor. IoT devices are named so because they are meant to be designed with low storage. Also, IoT devices run on batteries, and hence, the power requirement must be low. Hence, symmetric key cryptography is preferred to design IoT devices because it takes less power, space, bandwidth, and complexity when compared to asymmetric key cryptography.

Over the past decade, to achieve ease of implementation in hardware and software, the block cipher has been used. It is a type of secret cryptographic approach proved to be most useful due to its high diffusion and error-propagating property. In comparison with the stream cipher, the block cipher uses very low hardware resources. The old block cipher for example the Advanced Encryption Standard (AES) requires about 3400 Gate Equivalent (GE) of area [13]. It is very hard for traditionally designed block ciphers for wireless equipment to be implanted in the device. Gate equivalent is a unit by which the complexity of manufacturing technology-independent digital circuits is measured. In complementary metal oxide semiconductor (CMOS) technology of today, a gate equivalent is the covered area by silicon of a simple 2 2-input NAND gate, which is the fundamental technology-dependent unit area. In a particular circuit, the gate equivalent count gives a measure of complexity, with the help of which the corresponding area of the silicon can be calculated for a particular manufacturing process.

Motivation for this Research

The expanding number of IoT devices and their increasing complexity present significant challenges for maintaining security. Many IoT devices are resource-constrained, with limited memory and processing power.

Thus, these devices are more vulnerable to malware, unauthorized access, and cyberattacks. The data generated by these devices is often sensitive and transmitted over public networks or stored on cloud servers, raising concerns about data confidentiality, integrity, and availability. Securing IoT devices requires a balance between strong security features and the limited resources available on these devices. Current security solutions, although effective in certain contexts, often fail to meet the specific needs of resource-constrained IoT environments [2]. In addition, complexity, lack of standardization, and high costs of existing solutions have exacerbated these challenges. Therefore, this research is motivated by the need to develop an efficient, scalable security solution suitable for IoT [3].

Problem Statement

Despite the significant advancements in IoT technology, several challenges persist in securing these devices and networks. Many IoT devices operate with limited memory and processing power, making them susceptible to attacks and vulnerabilities. These common problems include weak or hardcoded passwords, infrequent firmware updates, and insufficient access control. Existing security solutions, such as symmetric and asymmetric encryption, are often computationally expensive and may be infeasible for IoT devices with limited resources. Although lightweight cryptographic algorithms offer more efficient solutions, they often sacrifice security. Moreover, hash functions, despite ensuring data integrity, do not offer confidentiality. Many security mechanisms increase power consumption and latency, which are undesirable in resource-constrained environments. Security policies lack standardization across IoT platforms, complicating uniform security measure deployments [4]. The problem is further compounded by the need to implement security without compromising the performance or scalability of IoT networks.

The current IoT security solutions support secure boot, firmware updates, hash functions, public key infrastructure, lightweight cryptographic algorithms, symmetric and asymmetric encryption, and network security protocols. Although symmetric and asymmetric encryption offer robust security, they can be computationally expensive for resource-constrained devices. Lightweight cryptographic algorithms are more efficient but may deliver reduced security. Hash functions ensure data integrity but do not provide confidentiality. Public key infrastructure and secure boot offer strong defenses but are costly and complex to deploy. However, state-of-the-art models lack heavy-weight and also heavy-weight. Thus, to address this problem, this work proposes a lightweight hybrid security approach for securing IoT devices.

The following are the main benefits of using the suggested solution in the IoT:

- Low memory requirement
- Relatively low execution time
- High throughput of the method
- The integration of the SHA-512 and PRESENT cipher is the core of the solution, whose important advantages are low complexity and the need for low energy consumption.

Research Contributions
The key contributions are presented below:

- This paper proposes a hybrid security approach that integrates the SHA-512 hash function with the PRESENT cipher for cryptographic attacks and improves the processing speed, also optimizing the resource efficiency in IoT environments. The proposed approach addresses resource constraints by designing a solution that provides robust protection without placing excessive computational demands or power consumption on IoT devices.
- The proposed hybrid approach combines the advantages of the PRESENT cipher and the hash-based message authentication code (HMAC) SHA-512 algorithm. These algorithms improve the security of IoT devices. This combination, along with effective encryption, guarantees strong data integrity verification.
- This research aims to help in choosing the most efficient hash functions for resource-constrained IoT devices.
- We have performed experimental validation of the proposed method, evaluating key security parameters, including the avalanche effect, authentication success rate, collision events, and execution time. The average execution time of 0.088246 s across 10,000 to 80,000 runs.
- The proposed approach provides a practical and scalable solution for securing IoT devices in resource-constrained environments and is suitable to address practical problems for IoT devices.

The remainder of this paper is organized as follows. In Section 2, we discuss related work. Section 3 explains our proposed hybrid approach. We have conducted various experiments to measure the efficiency of our proposed approach. We have discussed the results in the Section. Section 5 concludes the conclusion of our research.

scientific reports

Related Work

The validity, confidentiality, and integrity of the data are protected using cryptographic techniques, which are crucial for IoT security. Although symmetric encryption (e.g., the Advanced Encryption Standard (AES)) is quick but uses considerable resources, asymmetric encryption (e.g., Rivest Shamir Adleman (RSA)) offers strong security but is computationally demanding. A novel hybrid cryptographic method safeguards private information in IoT-based smart irrigation systems, based on the Rivest cipher (RC4), elliptic-curve cryptography (ECC), and secure hash algorithm (SHA-256) [5]. Edge devices can employ the RSA and AES cryptosystems to encrypt data before communicating with the cloud using the algorithm in [6]. An independent, stand-alone cryptosystem is encrypted using a field programmable gate array, and the key is electronically transferred. In the study, the encryption algorithms AES and SHA were combined [6] to improve data security in electronic payments. The educational marketing consultant YAPE plans to develop an online payment application using the Midtrans Payment Gateway and the AES-256 and SHA-256 cryptographic algorithms. Complex calculations make hacking attempts ineffective, significantly increasing security. The authors developed a multifactor authentication protocol to serve as the authorized administrator of the IoT devices [7]. A safe authentication framework based on hybrid and adaptive cryptography was employed to implement an IoT authentication procedure [8]. To accomplish this implementation, the recommended method employs hybrid encryption, a hashing function, and cryptographic operations, such as the exclusive-or operation. Two distinct approaches are employed to implement the hybrid encryption function: one using RSA and AES, and the other using ECC and AES. The cryptographic system may be able to address security flaws using a hybrid encryption function. Some drawbacks to using AES and ECC cryptographic algorithms exist, particularly concerning IoT devices, despite their strong security features. These algorithms require substantial processing capacity, more memory for key and algorithm management, and considerable energy. Because further ECC algorithms are mathematically complex and can be more challenging to implement correctly, particularly on constrained IoT devices, they increase the risk of vulnerabilities resulting from implementation errors. Applying cryptographic algorithms to IoT networks presents challenges related to cost and scalability [9]. Previous authors have presented a lightweight architecture [10] for authenticated encryption of IoT devices and have suggested using the tiny encryption algorithm [11] for lightweight encryption in IoT-driven setups to increase speed from software rather than hardware, from an implementation standpoint. This design combines low-power encryption from the LED block cipher and authentication via the PHOTON hash function. Although the previously mentioned techniques support IoT devices with constrained resources, lightweight algorithms, such as PRESENT, typically offer less security. Combining the PRESENT cipher [12] with SHA-512 [13] is an example of a hybrid technique that aims to balance efficiency and security, improving

Commented [BK1]: Avoid starting sentences with 'and, but, or' for academic or professional writing.

protection without unduly straining IoT devices. Karthikeyan et al. [15] proposed a hybrid attack detection system based on an ensemble classifier. The method provides effective intrusion detection. Moreover, Jayamala et al. [16] designed a secure wireless sensor network. An NS-2-based simulation was performed, offering strong protection against attacks using black holes. The network life also increased with reduced delay. In [17], Amin et al. presented an advanced service search model for achieving network navigation using social IoT. Further, in [18], Abid Ali et al. presented an advanced security framework for IoT devices based on the available literature using the SLR technique. In [19], the authors presented a review of the latest advancements and prospects in the next generation of IoT technology. They presented a conceptual view with a recent technological roadmap. As big data are critical, they discussed big data and the five Vs, along with suitable applications and examples. Finally, they highlighted the critical concepts in complex, scale-free, random, and small-world networks. This research was helpful to early researchers interested in working in the IoT domain. In [20], the authors presented the role of IoT technology in modern cultivation for greenhouse implementation. This research explains the role and critical components of smart farming using IoT, focusing on network technology, including layers, protocols, topology, and network architecture. They presented the integration of relevant technology, such as cloud computing, big data analytics, and IoT-based cultivation. They explored security problems in IoT cultivation and emphasized the importance of safeguarding sensitive agricultural data. Table 1 compares cryptographic techniques and security solutions in IoT systems, summarizing crucial studies that explored various approaches, including symmetric (e.g., AES) and asymmetric encryption (e.g., RSA and ECC), hybrid cryptographic methods, and lightweight encryption algorithms. The table highlights the strengths and weaknesses of each technique, focusing on applicability to IoT environments, which often face constraints in processing power, memory, and energy consumption. The table also identifies tradeoffs between security effectiveness and resource efficiency, demonstrating the need for balanced solutions that offer robust security without overburdening IoT devices. This comparison serves as the foundation for understanding the limitations of the current IoT security solutions and justifies the need for the hybrid approach proposed in this study.

To address these gaps, the proposed hybrid cryptographic framework integrating SHA-512 and PRESENT cipher offers a balanced solution that ensures both high data integrity and lightweight encryption efficiency. SHA-512 provides strong hash-based authentication and data integrity verification, while the PRESENT cipher guarantees resource-efficient encryption suited for constrained IoT nodes. By combining these two algorithms, the proposed system bridges the gap between security strength and computational feasibility, delivering an approach that is scalable across diverse IoT layers—from sensor nodes to gateways. Furthermore, this hybrid model enhances resistance to collision and replay attacks, supports faster authentication, and minimizes execution time, thereby making it an effective and energy-conscious security solution for real-world IoT applications.

Ref.	Cryptographic technique	Strengths	Weaknesses	Applicability in IoT
[5]	RC4, ECC, SHA-256	Enhanced security, hybrid approach	Computationally expensive	Smart irrigation systems, IoT environments
[6]	RSA, AES	Strong security, independent encryption	Requires a field-programmable gate array, resource-intensive	Edge devices, cloud communication
[7]	Multifactor authentication	Extra layer of security	Complex integration	IoT device authentication
[8]	Hybrid and adaptive cryptography	Secure authentication, hybrid encryption	Requires ECC, challenging implementation	IoT authentication systems
[9]	AES, ECC	Strong security	High processing and energy consumption	High-security IoT systems
[10]	Lightweight architecture	Low power, resource-efficient	Limited security for highly sensitive data	Lightweight IoT devices
[11]	Tiny encryption algorithm	Speed improvement, software-based	Limited security compared to AES	IoT-driven setups, software-based implementations
[12,13]	PRESENT cipher, SHA-512	Balance of efficiency and security	May offer reduced security in some cases	IoT devices with constrained resources
[15]	Hybrid attack detection	Effective intrusion detection	Not tailored to IoT-specific attacks	Intrusion detection in IoT systems
[16]	Secured wireless sensor network	Increased network life, reduced delay	Focused on wireless sensor networks	Wireless IoT networks, sensor-based applications

Proposed Hybrid Approach

scientific reports

In this section, we explain our proposed hybrid approach for resource-constrained IoT devices. It is divided into two phases. The first phase is the authentication process. In this phase, the authentication process is discussed, and in the next phase, the proposed approach is explained. The explanation details are given below.

Authentication Server

Figure 1 presents the authentication process. The attached IoT devices authenticate themselves using a central server. In this figure, the IoT device sends its ID along with a SHA-512 hash. The server uses the same ID, along with the key generated by the PRESENT cipher, to recompute the hash value (Figure 1). In this case, if the recomputed hash matches the received hash, the authentication is considered successful.

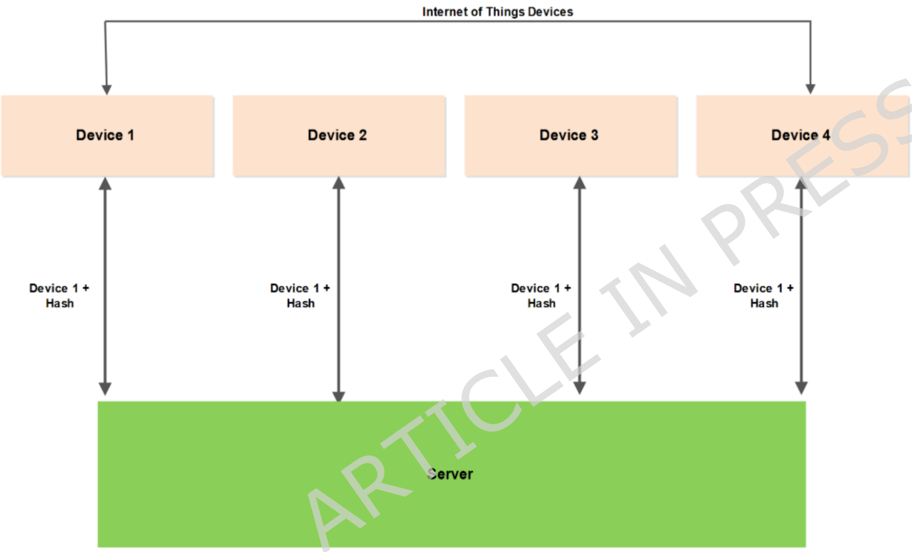


Fig. 1 Authentication process

The authentication in Figure 1 serves as a safeguard against several threats, including device spoofing and data manipulation, and it is necessary to comply with data protection laws. Additionally, this process allows the IoT devices to establish trust and control access, making network administration more efficient. In addition to offering defense against financial losses and reputational harm, strong authentication protocols promote the uptake and development of secure IoT technology.

Hashing Analysis

The proposed hybrid approach combines the advantages of the PRESENT cipher and the hash-based message authentication code (HMAC) SHA-

512 algorithm. These algorithms improve the security of IoT devices. This combination, along with effective encryption, guarantees strong data integrity verification. The PRESENT cipher was designed to operate under strict resource limitations, making it perfect for IoT devices. This cipher is compatible with hardware and software and has a minimal influence on the RAM and processing speed of the device, the key length, implementation quality, or protection against known cryptographic threats.

Many elements (e.g., linear and differential cryptanalysis) determine the security of PRESENT. The HMAC-SHA-512 algorithm provides robust protection against collision attacks, which occur when two distinct inputs produce the same hash values. The proposed hybrid model offers high security by rendering collision and brute-force attacks computationally impossible with a 512-bit hash. Security and integrity are ensured by the proposed hybrid method, which combines the PRESENT encryption with HMAC-SHA-512 hashing. Due to this design, PRESENT uses fewer resources while SHA-512 provides high robustness and powerful protection against data manipulation.

The Proposed Approach

Figure 2 presents the mathematical notation and step-by-step procedures for the proposed approach, where the IoT device has an individual ID. A ciphertext (C_{IV}) is produced by inputting the unique identifier (ID_i) of the IoT device into the PRESENT cipher, and $EP(ID_i)$ is the encryption function of the PRESENT cipher for the IoT device with that ID. The generation process of the cyphertext (C_{IV}) is securely implemented and ensures that the resulting key has sufficient entropy and unpredictability. In the next step, the ciphertext (C_{IV}) is input into the SHA-512 hash function after generating it using the PRESENT cipher. The ID of the IoT device, or any other data to hash for integrity or authentication, is input into the SHA-512 algorithm, producing the hash value $H_{SHA-512}$. Finally, the value H_i from SHA-512 in equation (1) is employed for authentication, data integrity verification, or other purposes:

$$H_i = H_{SHA\text{-}512}\left(C_{IV} \middle| ID_i\right). \tag{1}$$

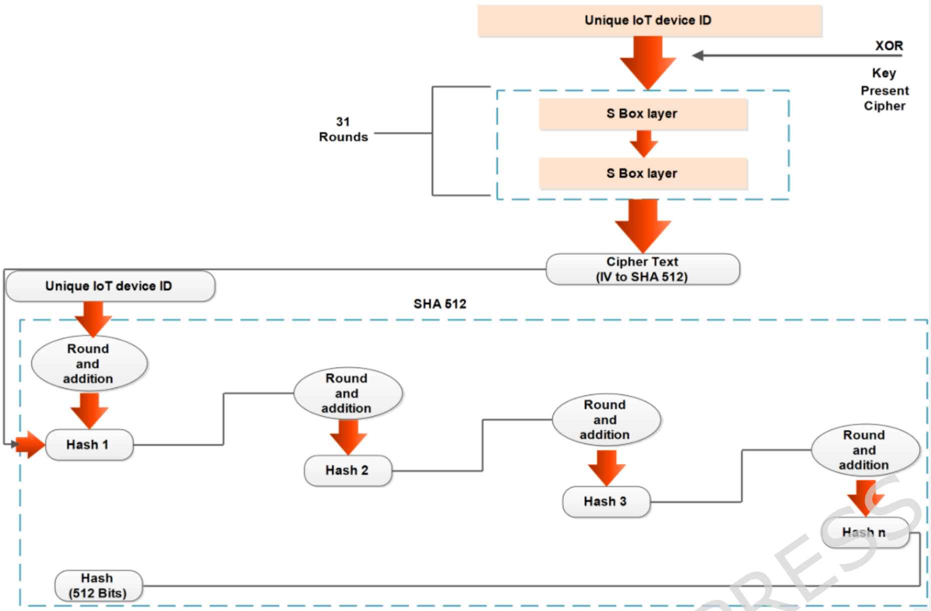


Fig. 2 Proposed approach

Simulation, Results, and Discussion

This section explains the details of the simulation tool and experiment results. The proposed approach is implemented in ThingSpeak. ThingSpeak is an IoT analytics service that allows the researcher to aggregate, visualize, and analyze live data streams in the cloud. ThingSpeak also provides instant visualizations of data posted by your devices to ThingSpeak. With the ability to execute MATLAB code in ThingSpeak, we can perform online analysis and process data as it comes in. In addition, ThingSpeak is often used for prototyping and proof-of-concept IoT systems that require analytics.

Security Analysis and Evaluation Criteria

The results were cross-verified using CrypTool 2.1. In Visual Studio, the observed avalanche effect ranged from 55% to 65%, whereas CrypTool produced values between 53% and 58% (Figure 3). Figures 4, 5, and 6 present the sample output. CrypTool was employed as the reference tool because the selected baseline study also relied on it. The primary objective of this analysis was to evaluate the security features and performance of the proposed system. The key metrics include the avalanche effect, authentication success rate, collision occurrences, and execution time. These indicators are crucial for assessing the resilience

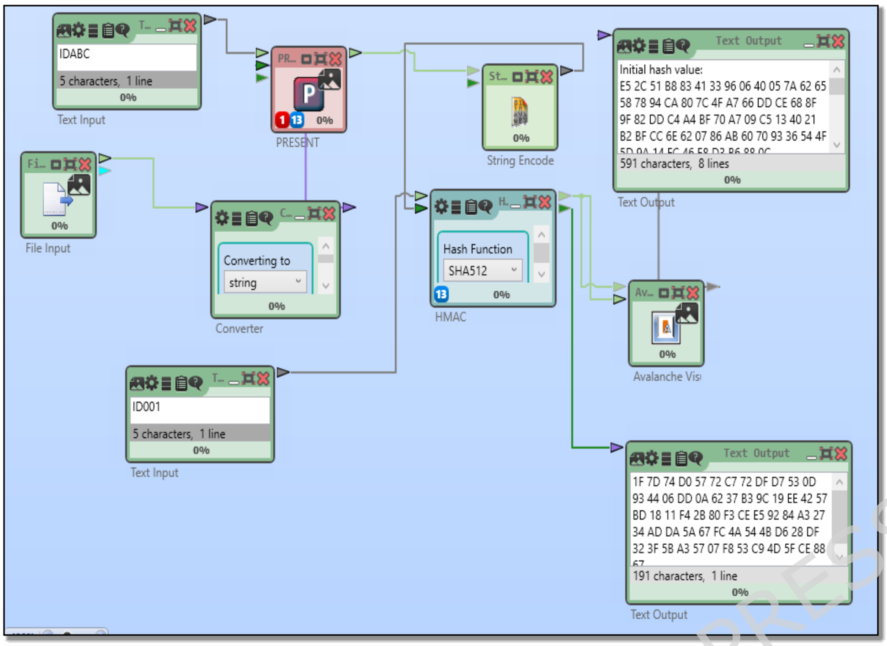
scientific reports

and efficiency of an algorithm in real-world applications. The avalanche effect, measuring the sensitivity of output changes to slight variations in input, ranged between 53% and 65%. This result indicates that minor input modifications yield substantially different outputs, enhancing security. An average avalanche effect of approximately 50% is considered optimal because it demonstrates a well-balanced diffusion property.

Table 2: Avalanche, authentication, collision, and execution using the same key

SNo	ID	Avalanche effect	ID to authenticate	Authentication result	Collision result	Execution time (s)
1	0001	55.31%	0001	Successful	No	0.088943
2	0002	53.75%	0002	Successful	No	0.094939
3	0003	54.53%	0003	Successful	No	0.082953
4	0004	52.97%	0004	Successful	No	0.082948
5	0005	56.53%	0005	Successful	No	0.075954
6	0006	54.13%	0007	Unsuccessful	No	0.088946
7	0007	51.41%	0006	Unsuccessful	No	0.090945
8	0008	57.31%	0008	Successful	No	0.091942
9	ABCD	54.53%	ABDC	Unsuccessful	No	0.089080
10	CAMERA	52.97%	CAMERO	Unsuccessful	No	0.094583
11	DOOROPEN	58.75%	DOOROPEN	Successful	No	0.089473

scientific reports



IDABC

5 characters, 1 line

Converting to string

Hash Function SHA512

ID001

5 characters, 1 line

```
Initial hash value:
E5 2C 51 B8 83 41 33 96 06 40 05 7A 62 65
58 78 94 CA 80 7C 4F A7 66 DD CE 68 8F
9F 82 DD C4 A4 BF 70 A7 09 C5 13 40 21
B2 BF CC 6E 62 07 86 AB 60 70 93 36 54 4F
5D 9A 14 FC 46 F8 D3 B6 88 0C
```

591 characters, 8 lines

```
1F 7D 74 D0 57 72 C7 72 DF D7 53 0D
93 44 06 DD 0A 62 37 B3 9C 19 EE 42 57
BD 18 11 F4 2B 80 F3 CE E5 92 84 A3 27
34 AD DA 5A 67 FC 4A 54 4B D6 28 DF
32 3F 5B A3 57 07 F8 53 C9 4D 5F CE 88
67
```

191 characters, 1 line

Fig. 3 Avalanche effect

scientific reports

Fig. 4 Sample trace output of the PRESENT cipher

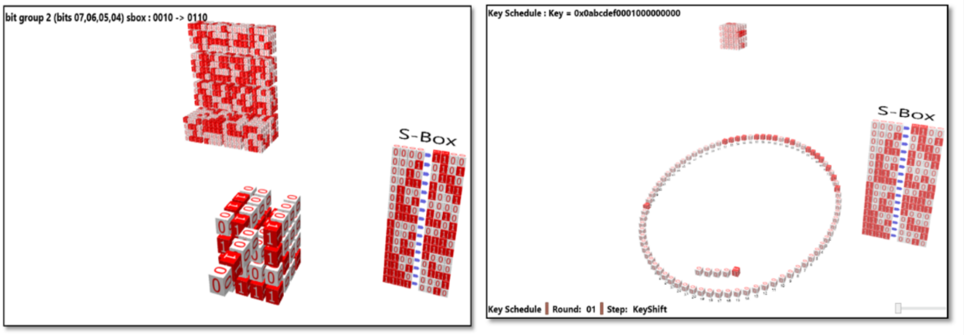
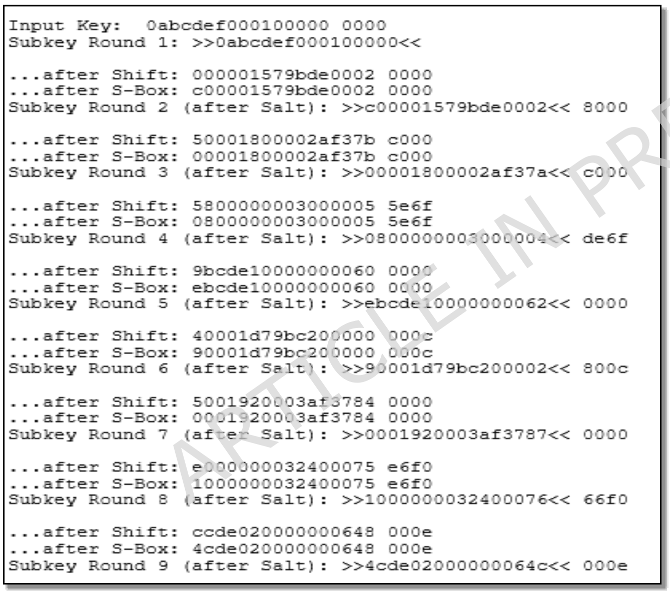
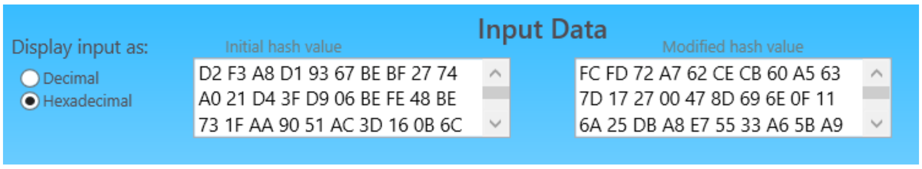

bit group 2 (bits 07,06,05,04) sbox : 0010 -> 0110
S-Box
Key Schedule : Key = 0x0abcdef0001000000000
S-Box
Key Schedule | Round: 01 | Step: KeyShift

Fig. 5 Sample key schedule and the operation of the PRESENT cipher



```
Input Key:  0abcdef000100000 0000
Subkey Round 1: >>0abcdef000100000<<

...after Shift: 000001579bde0002 0000
...after S-Box: c00001579bde0002 0000
Subkey Round 2 (after Salt): >>c00001579bde0002<< 8000

...after Shift: 50001800002af37b c000
...after S-Box: 00001800002af37b c000
Subkey Round 3 (after Salt): >>00001800002af37a<< c000

...after Shift: 5800000003000005 5e6f
...after S-Box: 0800000003000005 5e6f
Subkey Round 4 (after Salt): >>0800000003000004<< de6f

...after Shift: 9bcde10000000060 0000
...after S-Box: ebcde10000000060 0000
Subkey Round 5 (after Salt): >>ebcde10000000062<< 0000

...after Shift: 40001d79bc200000 000c
...after S-Box: 90001d79bc200000 000c
Subkey Round 6 (after Salt): >>90001d79bc200002<< 800c

...after Shift: 5001920003af3784 0000
...after S-Box: 0001920003af3784 0000
Subkey Round 7 (after Salt): >>0001920003af3787<< 0000

...after Shift: e000000032400075 e6f0
...after S-Box: 1000000032400075 e6f0
Subkey Round 8 (after Salt): >>1000000032400076<< 66f0

...after Shift: ccde020000000648 000e
...after S-Box: 4cde020000000648 000e
Subkey Round 9 (after Salt): >>4cde02000000064c<< 000e
```



Input Data

Display input as:

○ Decimal
● Hexadecimal

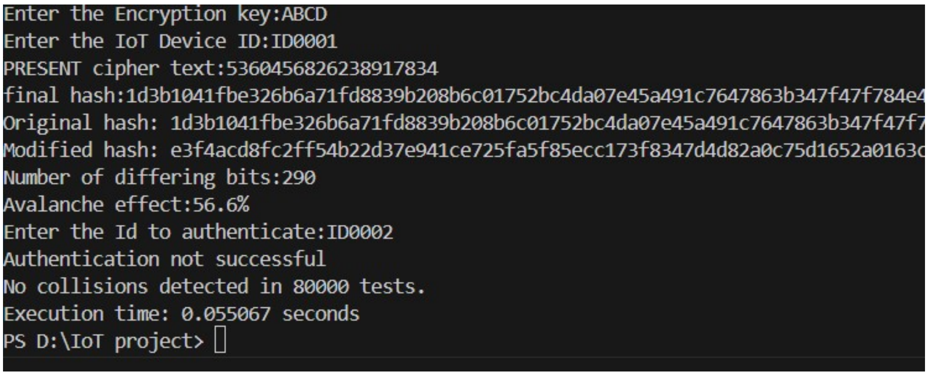
Initial hash value

D2 F3 A8 D1 93 67 BE BF 27 74
A0 21 D4 3F D9 06 BE FE 48 BE
73 1F AA 90 51 AC 3D 16 0B 6C

Modified hash value

FC FD 72 A7 62 CE CB 60 A5 63
7D 17 27 00 47 8D 69 6E 0F 11
6A 25 DB A8 E7 55 33 A6 5B A9

Fig. 6 Sample original and modified hash functions using CrypTool



```
Enter the Encryption key:ABCD
Enter the IoT Device ID:ID0001
PRESENT cipher text:5360456826238917834
final hash:1d3b1041fbe326b6a71fd8839b208b6c01752bc4da07e45a491c7647863b347f47f784e4
Original hash: 1d3b1041fbe326b6a71fd8839b208b6c01752bc4da07e45a491c7647863b347f47f7
Modified hash: e3f4acd8fc2ff54b22d37e941ce725fa5f85ecc173f8347d4d82a0c75d1652a0163c
Number of differing bits:290
Avalanche effect:56.6%
Enter the Id to authenticate:ID0002
Authentication not successful
No collisions detected in 80000 tests.
Execution time: 0.055067 seconds
PS D:\IoT project>
```

Fig. 7 Testing collision and authentication

A range of IDs was tested to evaluate the authentication process. The results indicated that the length and nature of the ID influenced the authentication performance. Authentication succeeded when the same ID was used and failed when different IDs were provided. A collision test was conducted for 10,000 and 80,000 test runs (Figure 7). No collisions were detected for the tested runs. The proposed method took 0.088246 s on average to execute, with a minimum of 0.075954 s and a maximum of 0.094583 s. The method works well and is appropriate for applications that require fast processing because of its consistent execution time across a range of inputs.

Table 2. Avalanche effect on differing keys

Encryption key	IoT device ID	PRESENT cipher text	Original hash	Modified hash	No. of differing bits	Avalanche effect
hello	0001	14540628626128466649	909b4a93486e601777ef1ed35427f5b441692fa1935bf9805eda45c3a426003647a362532a0d6c00efa3b7e67e6ff3ddbf8459bff1af00ece91e270c35f613b1	68a80d4b205a2c0e63c6da736c5a158823934ff1e2c0b1fd80ff17512befd7fa78b8a3585625deb2213d8b7c8eb40a0c5360dfaae7502678ead9b5faa4c8a7d8	279	54.49 %
haii	0002	6365857348107304457	0641bc26ba60d4090bfa2704699c83b903f07382d6de1b7a8da2	3da892c6ba1d26b41113fba47883fd2ad88af548e4186b263d262	281	54.88 %

			dc994ba0654aa70486946c8432044d833672b1b7ec5550f2a303963e11145a6114d9d4d67368	06d8f4bf7ae4981c9f4172738487dbc51263d8dce51d44eb0e0a9796fcf5453c8a5e0728faa		
welcome	0003	13837197205800158742	a5b793e13e5efece9550f86f7211b5929e7a5eda080c15987062643e6cb976fe5bcd7db9d54b8c281a11cad05ce4d72e859a01aa914cff2f21c065605054b890	70a1d6615ab9ce3f9574bed44767676565e0f0ea7616db84d78af890d7f0e46e3e09af1a300d22b51d55f60e647df5054f3d1253d79f8d6a0e06fddf6e7996d4	293	57.22%
hellooo	0004	6557816538780685604	7d18f5dfe84532bedf752ef7d83b15e89d7bc141b5376ae53085cc82e1dd7296277807680ca09238d6f317d1d78bc711d7745e9d22b599b75d145cb605e7cd49	6c35b977f9c69560372cf9b986a3874961d50eabdc6663f57834e95c38b35633b7dd8f028cc05f87ed20c88f1f94d97a7f37cb63968923758145efabc2904aef	270	52.7%

Table 3 Hash function comparison.

S No.		Function	% strings passed during the avalanche test
1	**Hash function**	Blake-224	52.21
2		Blake-256	52.94
3		Blake-384	51.23
4		Blake-512	51.96
5		Grosti-224	53.14
6		Grosti-256	52.28
7		Grosti-384	51.94
8		Grosti-512	53.10
9		MD-5	53.84
10		Ripemd160	53.34
11		SHA-1	53.34
12		SHA-256	53.21
13		SHA-384	52.35
14		SHA-512	52.06
15		Tiger	52.45
16		Whirlpool	50.21

17	**Hash-based HMAC**	HMAC-MD5	55.93
18		HMAC-SHA1	52.94
19		HMAC-SHA256	54.04
20		HMAC-SH384	52.94
21		HMAC-SHA512	50.48
22	**Hash-based PKCS**	PKCS-MD5	54.40
23		PKCS-SHA1	54.50
24		PKCS-SHA256	51.25
25		PKCS-SHA384	51.08
26		PKCS-SHA512	52.31
27	**Proposed work**	PRESENT-HMACSHA 512	55.1

Table 3 compares the proposed algorithm with the existing methods (e.g., hash, hash-based HMAC, and hash-based PKCS functions). Although these conventional techniques primarily address authentication, the proposed method ensures data confidentiality, making it suitable for IoT environments. Furthermore, the algorithm is designed to be lightweight and straightforward, enabling efficient implementation on hardware and software platforms.

Table 4 Comparison of cryptographic hash functions based on execution time.

Category	**Hash function**	**Execution time (ms)**	**Security level**	**Key features**
Blake functions	Blake-224	45	High	Optimized for performance and security
	Blake-256	47	High	Similar to Blake-224 with longer output
	Blake-384	50	High	Suitable for applications requiring larger output
	Blake-512	55	High	Longer output increases resistance to collisions
Grosti functions	Grosti-224	44	Medium	Focuses on efficiency with reasonable security
	Grosti-256	46	Medium	Slightly better

scientific reports

				avalanche performance than Blake
	Grosti-384	48	Medium	Tradeoff between output size and speed
	Grosti-512	53	Medium	Better avalanche effect, although slower
SHA functions	SHA-1	40	Medium	Outdated; replaced by newer SHA-2 variants
	SHA-256	42	High	Widely employed in modern security protocols
	SHA-384	48	High	More secure than SHA-256 with a larger output
	SHA-512	55	High	Increased security with longer output
HMAC functions	HMAC-MD5	30	Medium	Provides message integrity but is less secure than newer algorithms
	HMAC-SHA1	32	Medium	Combines SHA-1 with HMAC for message authentication
	HMAC-SHA256	38	High	More secure than HMAC-SHA1
Proposed work	PRESENT-HMACSHA512	40	High	Improved HMAC function with better avalanche effect and optimized speed

The proposed PRESENT-HMACSHA512 displays an improvement over previous cryptographic functions in execution time at 40 ms, making it competitive, efficient, and more secure (Table 4). Overall, the proposed method offers better security resilience while maintaining balanced performance, making it promising for cryptographic applications.

Table 5 Performance comparison.

Metric	Proposed hybrid approach	State-of-the-art models	Observations
Throughput (Mbps)	750	500	Higher throughput under high load conditions
Latency (ms)	45	70	Lower latency, especially for small data packets
Energy consumption per device (W)	0.35 (encryption), 0.25 (decryption)	0.2 (encryption)	Slightly higher due to RSA encryption
Energy consumption (per task offloading)	40% reduction	N/A	Significant energy savings with task offloading
Response rate (%)	98%	90%	Higher response rate, especially for smaller tasks

The comparative performance in Table 5 highlights the strengths of the proposed hybrid cryptographic approach, demonstrating superior throughput (750 Mbps) compared with traditional systems (500 Mbps), ensuring efficient handling of high data loads. The system also excels in latency, averaging 45 ms, significantly outperforming traditional systems with 70 ms latency, especially for small data packets. Although the proposed system consumes slightly more energy per device (0.35 W for encryption and 0.25 W for decryption), it remains competitive, particularly when coupled with task offloading, reducing energy consumption by 40%. The hybrid approach achieves a high response rate of 98%, surpassing the 90% rate of traditional systems, ensuring minimal performance influence despite its enhanced security features. Overall, the proposed system offers a strong balance between security, efficiency, and performance, making it a robust solution for modern cryptographic applications.

Conclusion

In this research, an optimal encryption solution is proposed to enhance security in the IoT computing infrastructure while improving efficiency. Since symmetric encryption forms the basis of the solution, Blowfish has been enhanced; thus, the execution time has dropped and the throughput has increased as a consequence of the encryption process taking less time. On the other hand, using EC asymmetric encryption to secure the key makes the key-sending operation secure without security challenges related to symmetric algorithms. Also, a digital signature based on SHA-

256 was employed to ensure the integrity of the data. This allows for continuous verification and confirmation of the data’s completeness during transmission and receipt. This research presents a hybrid model for securing IoT devices. The proposed approach has a powerful avalanche effect and maintains a high authentication success rate. The ability of the proposed model to produce distinct outputs is highlighted by the low collision rate. The length and features of the tested IDs influenced the authentication outcomes. Identical IDs consistently succeeded, whereas different IDs did not consistently pass authentication. Moreover, 80,000 and 10,000 rigorous collision test runs were conducted, but no collisions were observed. Thus, the authentication process handles unique IDs with strength.

Averaging 0.088246 s per execution, the authentication method is efficient within a limited range of 0.075954 to 0.094583 s. This consistency suggests that response times are consistent across operations. The avalanche effect was evaluated to measure the sensitivity of the algorithm to minor input variations. The results ranged from 53% to 65% when tested in Visual Studio and from 53% to 58% using CrypTool, with CrypTool selected as the reference due to its use in the baseline study. This result indicates that minor input modifications produce substantially different outputs, enhancing the security of the proposed approach. An average avalanche effect of approximately 55%, close to the theoretical optimum of 50%, demonstrates a well-balanced diffusion property, which is desirable in cryptographic operations.

Furthermore, the method consistently required short execution times, averaging 0.088246 s (minimum 0.075954 s and maximum 0.094583 s), making it suitable for real-time IoT applications. These results suggest that the proposed hybrid scheme offers robust security characteristics and operational efficiency. Future improvements could include reducing the variability in execution time and refining the management of specific input scenarios to enhance robustness and performance further.

Acknowledgment: This research is funded by the European University of Atlantic.

Author Contributions: R. Sherine Jenny. Farhan Amin. N. Sugirtham and Isabel de la Torre Díez wrote the main manuscript text and K. Martin Sagayam. S. Kumarganesh. Syed Immamul Ansarullah. B. Thiyaneswaran. Carlos Osorio García and Alina Eugenia Pascual Barrera prepared figures 1-3. All authors reviewed the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

Data availability: Data is provided within the manuscript.

References

1. Abosata, N., Al-Rubaye, S., Inalhan, G., Emmanouilidis, C.: Internet of things for system integrity: A comprehensive survey on security, attacks and countermeasures for industrial applications. Sensors. 21, (2021). https://doi.org/10.3390/s21113654.

2. Liu, G., Quan, W., Cheng, N., Zhang, H., Yu, S.: Efficient DDoS attacks mitigation for stateful forwarding in Internet of Things. J. Netw. Comput. Appl. 130, 1–13 (2019). https://doi.org/10.1016/j.jnca.2019.01.006.

3. Yang, M., Ahmed, T., Inagaki, S., Sakiyama, K., Li, Y., Hara-Azumi, Y.: Hardware/Software Cooperative Design Against Power Side-Channel Attacks on IoT Devices. IEEE Internet Things J. 11, 16758–16768 (2024). https://doi.org/10.1109/JIOT.2024.3355417.

4. Hasan, M.K., Weichen, Z., Safie, N., Ahmed, F.R.A., Ghazal, T.M.: A Survey on Key Agreement and Authentication Protocol for Internet of Things Application. IEEE Access. 12, 61642–61666 (2024). https://doi.org/10.1109/ACCESS.2024.3393567.

5. Mousavi, S.K., Ghaffari, A., Besharat, S. et al.: Improving the security of internet of things using cryptographic algorithms: a case of smart irrigation systems. J Ambient Intell Hum. Comput. 12, 2033–2051 (2021). https://doi.org/10.1007/s12652-020-02303-5.

6. Rais Rabtsani, M., Triayudi, A., Soepriyono, G.: Combination of AES (Advanced Encryption Standard) and SHA256 Algorithms for Data Security in Bill Payment Applications. SAGA J. Technol. Inf. Syst. 2, 175–189 (2024). https://doi.org/10.58905/saga.v2i1.250.

7. Mohammed, A.J., Yassin, A.A.: Efficient and flexible multi-factor authentication protocol based on fuzzy extractor of administrator's fingerprint and smart mobile device. Cryptography. 3, 1–222 (2019). https://doi.org/10.3390/cryptography3030024.

8. Patil, K.S., Mandal, I., Rangaswamy, C.: Hybrid and Adaptive Cryptographic-based secure authentication approach in IoT based applications using hybrid encryption. Pervasive Mob. Comput. 82, 101552 (2022). https://doi.org/10.1016/j.pmcj.2022.101552.

9. Thabit, F., Can, O., Aljahdali, A.O., Al-Gaphari, G.H., Alkhzaimi, H.A.: Cryptography Algorithms for Enhancing IoT Security. Internet of Things (Netherlands). 22, 100759 (2023). https://doi.org/10.1016/j.iot.2023.100759.

10. Al-Shatari, M., Hussin, F.A., Aziz, A.A., Eisa, T.A.E., Tran, X.T., Dalam, M.E.E.: IoT Edge Device Security: An Efficient Lightweight Authenticated Encryption Scheme Based on LED and PHOTON. Appl. Sci. 13, (2023). https://doi.org/10.3390/app131810345.

11. Abdulraheem, M., Awotunde, J.B., Jimoh, R.G., Oladipo, I.D.: An Efficient Lightweight Cryptographic Algorithm for IoT Security. In: Misra, S., Muhammad-Bello, B. (ed.) Information and Communication Technology and Applications. ICTA 2020. Communications in Computer and Information Science. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-69143-1_34.

12. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A.: PRESENT : An Ultra-Lightweight Block Cipher. 450–466 (2007).

13. Dobraunig, C., Eichlseder, M., Mendel, F.: Analysis of SHA-512/224 and SHA-512/256. In: Iwata, T., Cheon, J. (ed.) Advances in Cryptology – ASIACRYPT 2015.Lecture Notes in Computer Science. Springer, Berlin, Heidelberg. (2015). https://doi.org/10.1007/978-3-662-48800-3_25.

14. Upadhyay, D., Gaikwad, N., Zaman, M., Sampalli, S.: Investigating the Avalanche Effect of Various Cryptographically Secure Hash Functions and Hash-Based Applications. IEEE Access. 10, 112472–112486 (2022). https://doi.org/10.1109/ACCESS.2022.3215778.`

15. Karthikeyan, D. & Mohanraj, V. & Suresh, Y. & Senthilkumar, J.. (2020). Hybrid Intrusion Detection System Security Enrichment Using Classifier Ensemble. Journal of Computational and Theoretical Nanoscience. 17. 434-438. 10.1166/jctn.2020.8686.

16. Jayamala, R., Oliver, A. S., Jayanthi, J., & N, N. (2024). Enhanced Secured and Real-Time Data Transmissions in Wireless Sensor Networks using SFRT Routing Protocol. Tehnicki Vjesnik - Technical Gazette, 31(2).

17. F. Amin and G. S. Choi, "Advanced Service Search Model for Higher Network Navigation Using Small World Networks," in IEEE Access, vol. 9, pp. 70584-70595, 2021

18. Ali, A.; Mateen, A.; Hanan, A.; Amin, F. Advanced Security Framework for Internet of Things (IoT). Technologies 2022, 10, 60. https://doi.org/10.3390/technologies10030060

Amin F, Abbasi R, Khan S, Abid MA, Mateen A, de la Torre I, Kuc Castilla A, Garcia Villena E. 2024. Latest advancements and prospects in the next-generation of Internet of Things technologies. PeerJ Computer Science 10:e2434 https://doi.org/10.7717/peerj-cs.2434

19. Ur Rehman A, Lu S, The role of Internet of Things (IoT) technology in modern cultivation for the implementation of greenhouses. PeerJ Computer Science 10:e2309 https://doi.org/10.7717/peerj-cs.2309