Al-Khwarizmi Engineering Journal
ISSN (printed): 1818-1171, ISSN (online): 2312-0789
Vol. 21, No. 1, March (2025), pp. 35–48

Al-Khwarizmi Engineering Journal

Secure and Lightweight Cipher for Resource-constrained IoT Healthcare Applications using Snake Key Generation

Saja Kareem Ismael[1*], Mohamed Ibrahim Shujaa[1] and Ahmed Bahaaulddin A. Alwahhab[2]

[1]*Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq*
[2]*Technical College of Management, Middle Technical University, Baghdad, Iraq*
[*]Corresponding Author's Email: bbc0071@mtu.edu.iq

(Received 22 June 2024; Revised 24 August 2024; Accepted 23 September 2024; Published 1 March 2025)
https://doi.org/10.22153/kej.2025.09.002

Abstract

The integration of the Internet of Things (IoT) in healthcare has transformed the monitoring of physiological parameters, significantly enhancing the capabilities of medical diagnostics and patient care. This study explores the use of advanced cryptographic techniques to secure IoT healthcare systems, particularly through block chain (BC) technology. Moreover, this study evaluates the effectiveness of two cryptographic key generators—bee swarm key generator and snake key generator—in enhancing data security within a BC-based framework for remote patient monitoring. Our study conducts a series of NIST randomness tests to compare the performance of these key generators in terms of randomness, resilience to cryptographic attacks and computational efficiency. Results show that the snake key generator outperforms the bee swarm key generator, demonstrating higher randomness in key generation and better secure coverage of the orientation space. This finding is crucial for securing important healthcare data, especially given the complexity of IoT devices, which transmit vital patient health information. Furthermore, this study examines the double-layer encryption architecture used to secure data in at rest and during transmission from a sensor's gathering location to long-term storage. This approach is essential to preserve the privacy and integrity of patient data, which is critical for maintaining trust in IoT healthcare systems. This study aims to provide a comprehensive and comparative analysis to support the continued evolution of secure, efficient and scalable cryptographic solutions within IoT healthcare, demonstrating the importance of selecting the most effective key generation technique to address the unique security challenges of this new field.

Keywords: *IoT healthcare; blockchain technology; cryptographic key generation; data security; NIST randomness*

1. Introduction

An Internet of Things (IoT) variant integrates the human body with numerous sensor-equipped devices to measure various physiological parameters and create a set of linked IoT entities that are analysed by decision-support algorithms to forecast health-related results [1-4]. IoT technologies improve disease diagnosis and patient activity monitoring through biosensors that assess environmental and physiological conditions, enabling physicians to make informed medical decisions. Patients' wearable gadgets are the focus of patient-specific and customised nutrition care, with data security playing a crucial role. Security in remote patient monitoring systems involves the use of secure methods to encrypt patient information, transmit data through secure channels and limit access. Privacy and data integrity issues can be addressed through the implementation of a block chain (BC) as it creates a decentralised ledger to enhance electronic health records (EHRs) [5]. The authors proposed and implemented a BC framework, which focuses on the aspects of remote patient monitoring and data management, in another paper to support IoT healthcare [5]. Privacy issues arise from access control measures [6], whilst the application of BC in healthcare ensures patient data

integrity in remote treatment procedures [7]. IEEE-32-bit BCZ is recommended as a defense layer for transporting health records at the sensor level of Internet of Medical Things (IoMT), protecting the data packets sent from the sensor to the receiver [8]. This study evaluates the effectiveness of the bee swarm key generator and compares it to the snake key generator used to secure BC-based LPWMNs for RPM to mitigate the attack risks and add security layers within each block to protect data privacy [9].

The goal of this study is to develop a reliable system for accurately measuring and monitoring vital patient metrics whilst enhancing healthcare system security through an innovative BC approach. This approach involves using either the bee swarm or snake key generator, each offering different levels of encryption security. The key generator must ensure the protection of sensitive data by incorporating measures to anonymise identities linked to each BC block to further safeguard patient privacy. This comparative analysis aims to determine an effective cryptographic key generator for securing healthcare data within an IoT framework [10].

This study introduces a new two-layer encryption architecture designed for IoT healthcare applications, utilising the bee swarm key generator and the Speck cipher to protect patient data from initial capture to prolonged storage. This dual encryption strategy ensures utmost confidentiality and security of patient information, validated through several NIST randomness tests that confirm the unpredictability and robustness of the cryptographic techniques used.

The key contributions of this study are as follows:

- Development of a robust framework for IoT healthcare: Ensuring the secure confinement of critical medical data and maintaining data integrity during sensor data transmission;
- Enhancement of BC security within the public ledger: Anonymising sensor data before cloud storage using lightweight block encryption, acting as a privacy and confidentiality filter, thereby maintaining the anonymity of the block's owner;
- Introduction of an innovative secure random key generator inspired by bee swarm behaviour: This generator significantly enhances the randomness and security of cryptographic keys, drawing parallels with snake optimisation (SO) techniques to further optimise key generation processes;
- Hybrid key generation approach using bee swarm algorithms and SO techniques: This method constructs secure and random cryptographic keys leveraging the innate efficiency of bee swarm structures and the effective adaptability of snake optimisation, ensuring robust security measures for IoT healthcare data transmission and storage.

2. Related Work

Current IoT-related medical research focuses on disease identification, patient health status and medication administration whilst ensuring information privacy in patient records. Ogundokun et al. [11] proposed a crypto-stegno method for protecting healthcare data in IoT, which effectively prevents data leakage. However, this method lacks the integration of BC learning parameters. Abdellatif et al. [12] advocated for the development of an emergency response and remote monitoring BC solution with secure non-harmonised vital sign data transmission; however, it struggles with cybersecurity challenges. Pandey et al. [13] proposed a healthcare cybersecurity architecture based on BC, deep belief networks (DBN) and residual networks (ResNet) for disease detection and classification. Veeramakali et al. [14] presented a cryptographic IoT security authentication scheme based on optimal homomorphic encryption (OHE) and a deep learning neural network (DNN) for data encryption. Abd El-Latif et al. [15] proposed a novel secure e-healthcare architecture based on BC, utilising the orthogonal particle swarm optimisation (OPSO) for concealed transmission and the optimised DNN (ODNN) for disease classification. Quantum computer threats are emphasised for BC's role in cybersecurity, whilst quantum walks are considered for fresh cryptographic algorithms. Cao et al. [16] proposed an authentication and encryption system based on the QIWQ for BC data transfer. Ciphers developed by Shankar et al. [17] featured a BC-based ICO for cybersecurity, leveraging the random neural network model and unique encryption. Mathews et al. [18] introduced an enhanced Two Arch2 solution, which claimed to be more scalable than the existing one and closer to achieving BC time in the optimisation of model metrics.

3. IoT Ecosystem

The IoT framework allows for the integration of smart devices—such as mobile phones, sensor devices and Raspberry Pi—into a network that senses, monitors and responds to various environmental stimuli. The goal is to expand

connectivity by simultaneously linking multiple devices to the internet [20,21,22]. This connectivity enables human-to-machine and machine-to-machine interactions. For instance, humans can engage with machines to conduct medical procedures, remotely monitor patients at home and monitor real-time conditions within hospital infrastructure [23,24,25,26,27]. Moreover, machines can autonomously exchange data, with sensors, for example, relaying information to a cloudlet system for temporary data storage [28, 29]. A representative model of an IoT-enabled healthcare system is illustrated in Fig. 1.

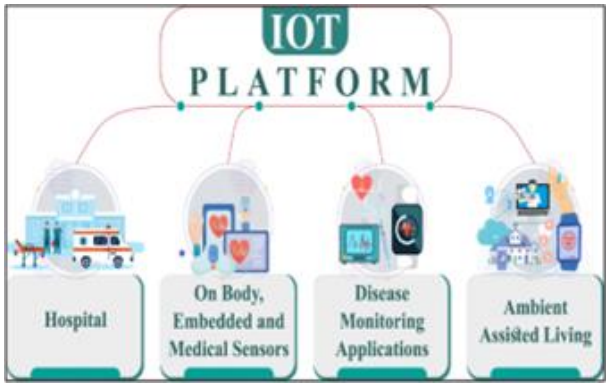

Fig. 1. IoT and its associated healthcare platform

The architecture of a traditional IoT-enabled healthcare system consists of four fundamental components. The first component is the hospital environment, which serves as a hub for healthcare providers and accommodates inpatients and outpatients [30, 31]. IoT technologies enable hospital administrators to effectively track the health and recovery of outpatients in their homes through real-time monitoring. This monitoring is facilitated by wearable or embedded medical body sensors that collect data, which are then processed and interpreted into actionable insights using machine learning algorithms and specialised software. This setup aids medical personnel in remotely assessing the current health conditions of outpatients [32].

Meanwhile, outpatients who are no longer under intensive monitoring, or those who have recuperated, are equipped with IoT-enabled ambient assistive living devices. These devices help individuals in independently managing their health by providing real-time feedback on various health metrics, such as caloric expenditure, blood sugar levels and heart rate. Consequently, individuals are highly equipped to proactively address potential health issues with the information supplied by these IoT devices. This proactive health management enhances overall patient welfare and equips caregivers with the essential data and tools required for delivering high-quality health services to the public, an approach commonly referred to as smart healthcare.

4. Block Chain

Block Chain (BC), an innovation with multiple uses, is now considered a mainstay in healthcare. BC can be described as a linked chain of embedded data blocks with records that have been made secure through the use of cryptographic hash algorithms [33]. Every peer in the decentralised system has a copy of the BC and acts according to a consensus mechanism, increasing the system's reliability and capability for continuous operation [34-36]. BC applications ensure the safety of medical records, the supply chain of drugs and the secure transfer of EHRs. The BC components, such as cryptography, immutability and decentralisation, enhance the security levels by making data modification without permission difficult [37-38]. Nevertheless, cyber threats are not foreign to BC; vulnerabilities persist, making cryptographic methods essential [39].

Electronic health records, once stored in digital format, now have a safer, distributed home in block-based chains [40]. HE is achieved through the collective effort of funding authorities, medical researchers and health departments. Some BC use cases face unique risks and constraints, such as 51% attacks, where miners controlling more than half of the network's hash power can alter the BC. The suggested resistance measures include optimising hash rates, proper monitoring of mining pools and evaluating Proof of Work (PoW) consensus algorithms [41]. Phishing attempts deceive users into divulging secure login information; the immediate protective measures include increasing browser security, using anti-malware and avoiding suspicious links. Routing attacks involve intercepting data transmitted to internet service providers; thus, safe routing protocols, data encryption and strong passwords are essential to prevent such attacks. Although signing with keys is important, they can still be forged, making the protection of private keys crucial [41].

Insecure perimeters of BC networks can collect user credentials, highlighting the importance of prohibiting the storage of BC keys in plain text and regularly reviewing system access. Therefore, this extensive literature review on BC security in the healthcare context underscores the necessity for further enhancement of protected health information [42].

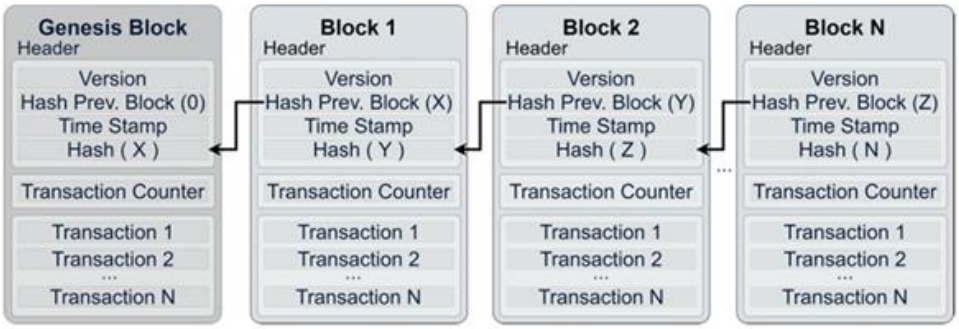

Fig. 2. General representation of the structure of a BC

5. IoT Security

The real world consists of numerous physical objects connected through the IoT environment, necessitating encryption to protect against eavesdropping for clique, chip ID and chip integrity and authenticity. Data collected in the sensor node itself must be protected, and firewall and IDS are crucial to protect data in the intermediate networks from the internet to the WSN [43]. Most IoTs are resource-constrained, making a single approach to security impractical. AES and DES are not feasible to implement in IoT devices because they require numerous gates and a high-power consumption [43]. Lightweight cryptography is suitable for IoT devices partly due to the low quantities of RAM or ROM/Flash for storage, processing capability and shorter clock cycles. Lightweight ciphers can be categorise based on software or hardware implementation. Software implementation offer low cost and flexibility. Daemen et al. asserted that the key features of an effective lightweight cipher are memory efficiency, resource efficiency, security strength, low implementation complexity and high throughput with short processing time. The model should efficiently work with minimal memory requirements, effectively perform on low-power devices. Moreover, the model should consume minimal energy to prevent battery drain during use. These ciphers should provide security against linear, differential and other attacks, such as biclique, zero correlation, MITM and algebraic attacks. Considering such requirements and security considerations, lightweight ciphers are appropriate for securing communication in constrained environments [43].

5.1. Speck Cryptography

Speck is a lightweight block cipher used in the cryptographic field, designed by the United States National Security Agency and publicly released in 2013 June, belongs to this family. Speck shows flexibility concerning key and block sizes, featuring an ARX design (Fig. 3). Moreover, Speck has several variations, including Speck32/64, Speck48/72 and Speck64/128, where the first number represents the block size in bits, and the second denotes the key size in bits.

In the configuration shown in Fig. 3, the cipher is referred to as Speck2n/wn, where 2n represents the block size in bits, and wn denotes the key size in bits. Speck's underlying round function involves right rotations of n-bit words and their addition:) The operations are bitwise XOR, modular addition and left and right circular shift. Together, these operations form an efficient and secure cipher for resource-constrained, power-efficient environments [44].

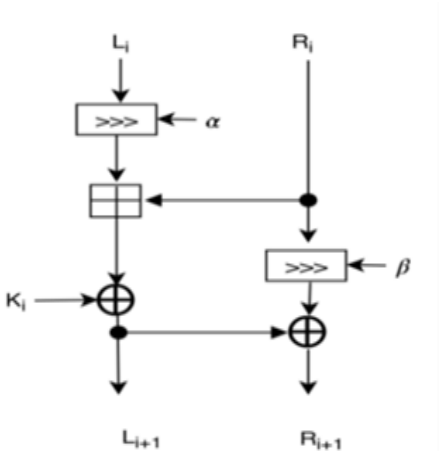

Fig. 3. Speck round function

5.2. Bee Swarm Optimiser

Bee Swarm Optimisation (BSO) [45] refers to an optimisation algorithm, as it is a nature-inspired optimisation algorithm that simulates the foraging

behaviour of honey bees. BSO uses the collective intelligence and swarm behaviour of bees to determine optimal solutions for complicated algorithms. The BSO algorithm is divided into two principal search phases: exploration and exploitation. In the exploration phase, scout bees search for new food sources (possible solutions to the problem in the problem space) by running in a random-by-choice way. During the exploitation phase, working bees exploit the most successful feeding sites, improving the quality of the solutions. Furthermore, the honeybees act as onlookers, evaluating the shared food sources of foragers and stochastically selecting which to further exploit. This mechanism allows the search process to dynamically balance exploration in new-candidate areas and exploitation of the best solutions. Given that BSO has been successful on a variety of optimisation challenges, such as engineering design, scheduling and machine learning, it can converge to high-quality solutions and maintains diversity throughout the swarm.

BSO is a novel nature-inspired algorithm that has gained significant attention in recent years for key generation in cryptographic applications, mainly due to its ability to provide robustness and efficiency. Strong, random keys are essential for ensuring security and protecting systems from cryptographic attacks.

Furthermore, BSO can efficiently search across a large search space and converge towards optimal solutions, making it well-matched for this. During the exploration phase, the algorithm evaluates a wide range of potential keys, significantly enhancing their randomness and unpredictability. In the exploitation phase, these potential keys are fine-tined to ensure that they are cryptographically strong and complex. Research has shown that BSOs achieve better randomness in key generation and produce more secure keys than traditional key generation methods, which is an important factor when considering the sensitive data that can be accessed and the threats of cybercrime. The versatility of BSO also makes it highly adaptable for customisation to various cryptographic protocols and demands, positioning it as an effective and competitive cryptographic key generator [45].

Pseudocode for Bee Swarm Optimiser

1. Initialise the population of scout bees with random solutions.
2. Evaluate the fitness of each scout bee.
3. **Whilst** the stopping criterion is not met, **do**:

For each employed bee, **do**:

1. Select a solution from the neighbourhood of the current solution.
2. Evaluate the fitness of the new solution.

If the new solution is better than the current solution, **then**:

1. Replace the current solution with the new solution.
2. **For** each onlooker bee, **do**:
   1. Select an employed bee based on the fitness probability.
   2. Select a solution from the neighbourhood of the selected employed bee's solution.
   3. Evaluate the fitness of the new solution.
   4. **If** the new solution is better than the selected employed bee's solution, **then**:
      1. Replace the selected employed bee's solution with the new solution.
      2. For each scout bee, **do**:
      3. If the solution has not improved for a certain number of iterations, **then**:
      Replace the solution with a new random solution. Memorise the best solution found so far.
4. **End whilst**
5. **Return the best solution**

5.3. Snake Optimisation

Snake Optimisation (SO) [46] is a mating-inspired algorithm that algorithm simulates the common behaviours of snakes in searching for food and competing for mating. Mating occurs in sub-chilled areas, where food is available, but the temperature is considerably low. The algorithm has two main phases: exploration and exploitation. In the search state, snakes randomly forage for food during non-mating periods, mimicking their behaviour in the wild when food is scarce. This process involves generating a random population and dividing it into males and females. During the exploitation phase, snakes approach food when it is abundant, and many engage in mating or competing for mates if the climate permits. This process updates the positions of snakes based on food availability and temperature, selecting the best solutions for each snake. The algorithm balances exploration and exploitation through snake behaviour, where food quantity and temperature in each cell determine their movements. The SO algorithm initialises the generation of its random populations (step 1) and evenly dividing it into males and females. Food quantity and temperature influence snake behaviour. When food is scarce, snakes engage in exploration, randomly searching for food. When food is abundant, snakes

move towards it. If the conditions permit, snakes either compete for mates or engage in mating. The algorithm updates positions based on these phases, using equations to model the movement and behaviour of snakes. The effectiveness of the SO algorithm was tested on benchmark functions from CEC 2017, where it outperformed several state-of-the-art algorithms in terms of average results and standard deviation. This algorithm maintained a good balance between exploration and exploitation, demonstrating its effectiveness in solving optimisation problems. Additionally, the SO algorithm was tested on CEC 2020 functions, showing strong performance against other competitive algorithms. The results highlighted the algorithm's capability to adapt to various conditions and achieve high rankings across different functions. The effectiveness of the SO algorithm in key generation for cryptographic systems lies in its robust exploration and exploitation mechanisms. Inspired by snake behaviour, the algorithm thoroughly searches a wide range of possible keys (exploration) and refines the best candidates (exploitation). This approach ensures that the generated keys are highly random and secure, minimising the predictability and making them difficult to guess or brute-force. The SO algorithm's ability to balance these phases allows it to efficiently handle large search spaces, producing cryptographic keys that meet high-security standards. Overall, the comprehensive search strategy and high-quality solution refinement make the SO algorithm a powerful tool for secure key generation [46].

Algorithm: SO for Key Generation

1. Initialise problem setting
- a. Define the dimensions (Dim) of the problem.
- b. Set the upper bound (UB) and lower bound (LB) of the search space.
- c. Set the population size (Pop_Size) and maximum number of iterations (Max_Iter).

2. **Randomly initialise the population**
- d. **For** each individual i in the population:
- i. Xi = Xmin + rand() * (Xmax − Xmin)

3. Divide population N into two equal groups: males (Nm) and females (Nf)
- e. Nm ≈ N / 2
- f. Nf = N − Nm

4. Whilst (current iteration $t \leq Max\_Iter$) **do**
- g. Evaluate each group Nm and Nf
- h. Find the best male ($fbest, m$) and best female ($fbest, f$)
- i. Define temperature ($Temp$) using $Temp = exp(-t / T)$
- j. Define food $Quantity\ (Q) using\ Q = 0.5 * exp((t - T) / T)$

5. If ($Q < 0.25$), then // Exploration phase (no food)
- k. **For** each male Xi,m:
- i. $Xi, m(t + 1) = Xrand, m(t) \pm 0.05 * Am * ((Xmax - Xmin) * rand() + Xmin)$
- ii. $Am = exp(-frand, m / fi, m)$

6. For each female Xi,f:
- iii. $Xi, f(t + 1) = Xrand, f(t) \pm 0.05 * Af * ((Xmax - Xmin) * rand() + Xmin)$
- *iv. Af = exp(−frand,f / fi,f)*

7. Else if (Q > 0.6), **then** // Exploitation phase (food exists)

8. For each individual Xi,j (male or female):

$$Xi, j(t + 1) = Xfood \pm 2 * Temp * rand() * (Xfood - Xi, j(t))$$

Else // mating or fighting mode
1. If (rand() > 0.6), then // fight mode
2. For each male Xi,m:
3.Xi,m(t + 1) = Xi,m(t) + 2 * FM * rand() * (Q * Xbest,f − Xi,m(t))
4.*FM = exp(−fbest,f / fi)*

i.For each female Xi,f:
5.Xi,f(t + 1) = Xi,f(t + 1) + 2 * FF * rand() * (Q * Xbest,m − Xi,f(t + 1))

6. FF = exp(−fbest,m / fi)
- l. Else // mating mode
- *v. For each male Xi,m:*

7. Xi,m(t + 1) = Xi,m(t) + 2 * Mm * rand() * (Q * Xi,f(t) − Xi,m(t))

8. Mm = exp (−fi,f / fi,m)
- ii. For each female Xi,f:

9. Xi,f(t + 1) = Xi,f(t) + 2 * Mf * rand() * (Q * Xi,m(t) − Xi,f(t))

10. Mf = exp(−fi,m / fi,f)
- m. **If** egg hatches, then replace the worst male and female:
- i. Xworst,m = Xmin + rand() * (Xmax − Xmin)
- ii. Xworst,f = Xmin + rand() * (Xmax − Xmin)

11. Check the termination condition
- a. **If** t > Max_Iter, then terminate the process.

12. Return the best solution found.

6. Proposed Methodology

The proposed methodology (Fig. 4) aims to address the security challenges in IoT healthcare systems by integrating a robust dual-layer encryption architecture. This architecture utilises the bee swarm key generator and Speck cipher to secure patient data

from initial capture to prolonged storage. The primary focus is to ensure utmost confidentiality and security of patient information, validated through several NIST randomness tests that confirm the unpredictability and robustness of the cryptographic techniques utilised.

The methodology consists of real-time monitoring of different physiological parameters through intelligent sensors linked to IoT, including oxygen, temperature and pulse oximeter sensors. The sensors collect a bunch of critical patient data, which are then transmitted to a Raspberry Pi. The Raspberry Pi serves as the primary node for data aggregation and preliminary processing.

IoT sensor-level encryption using the Speck algorithm is used to encrypt the collected information to protect vulnerable healthcare data. Immediate encryption ensures that data sent by a sensor are securely communicated to a receiver, keeping all data safe from man-in-the-middle attacks and maintaining continuous privacy standards. Thereafter, these data are transmitted to the cloud, encrypted for safety and stored, making it available for healthcare providers when needed.

One of the key features of this proposed methodology is promoting the security and integrity of the stored data through BC technology. This mechanism uses a decentralised ledger based on the BC framework to provide durable data storage. Patient data are stored on a BC keychain, ensuring an additional level of data privacy encryption within each block. The research compares the effectiveness of two cryptographic key generation methods: the bee swarm key generation and the snake key generation, evaluating their efficiency and robustness in securing BC-based remote patient monitoring systems.

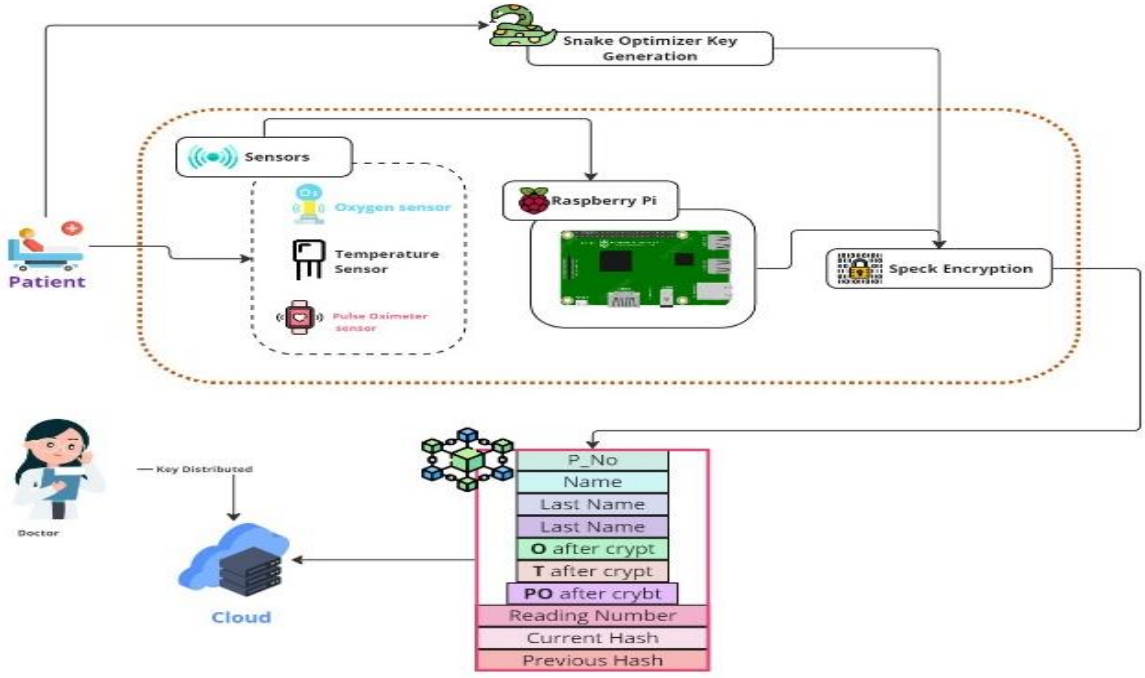

Fig. 4. Proposed Scheme

The concept of the mesoscale swarm key generator is inspired by the swarm behaviour of bees, which enhances randomness and strengthens the security of the crypto keys used. This generator is matched with the SO method, which works for key generation processes using the strategic flexibility of snake motion. The integration of these two methods aim to generate highly secure and random cryptographic keys, protecting IoT healthcare data communications and data-at-rest from unauthorised disclosure, thereby preventing privacy concern or breaches under applicable rules or regulations.

The methodology involves reinforcing access control to unify access permissions between sets of patients or families to enhance patients' and families' ownership of their data. This approach reduces the burden of access management, alleviates concerns about data leaks and prevents data from becoming obsolete due to malpractice. The approach suggests that each sensor data must be encrypted using lightweight block encryption and brought to the cloud as an anonymised version maintain confidentiality. This function serves as a privacy and confidentiality filter, ensuring that block ownership remains anonymous.

The goal of the study is to develop a robust and secure framework for gathering patient data, guaranteeing precise measurement and monitoring and improving healthcare delivery security by implementing a cutting-edge BC program. This dual-layer encryption, in addition to the new secure random key generator, is expected to significantly improve the security of patient data in an IoT healthcare environment.

7. Experimental Results

This section presents the results of testing the bee swarm key generator and snake key generator, both designed to enhance the security of BC-based remote patient monitoring systems. The experimental analysis had different themes, dealing with different sections of cryptographic security, including randomness, resistance to cryptographic attacks and computational efficiency. The results obtained from Table 1 show that both key generators passed the NIST randomness tests.

However, the snake key generator proved to be significantly faster in tests assessing resistance to modern cryptanalysis and computational efficiency. This generator demonstrated strong randomness properties, as demonstrated by the frequency (monotonicity) test and cumulative sums test, which resulted in the snake key generator producing unpredictably generated keys, unlike the bee swarm key generator. The resilience of some of these keys to various cryptographic attacks confirms that the snake key generator produced keys with significant resilience to key recovering attacks. These keys can effectively thwart numerous common varieties of attacks, including most types of brute force and cryptographic hash function attacks. This aspect is essential, especially when handling health data, as patient information must be securely managed and protected. The snake key generation is also computationally efficient, exhibiting the lowest computational cost amongst the three whilst ensuring security. This efficiency is an ideal fit for IoT scenarios, where resources are scarce, and processing power is highly valuable.

In summary, the key generators from the bee swarm and the snake present no practical flaws in terms of encryption safety. However, when comparing both, snake key generator outperforms in randomness, attack resilience and computational efficiency. The BC technology in question is considered the best fit for healthcare applications, encompassing nursing, telemonitoring, and patient-centric healthcare apps. This effectiveness lies in the technology's ability to encrypt health data in accordance with the necessary security and regulatory standards.

Table 1,
NIST randomness test results for speck cipher (Bee swarm vs snake)

	Snake optimiser		**Bee swarm**	
Test number and name	P-value	Conclusion	P-value	Conclusion
Frequency (monobit) test	0.3587953578869416	Random	0.2888443663464849	Random
Frequency test within a block	0.3587953578869416	Random	0.2888443663464849	Random
Discrete Fourier transform (spectral) test	0.24690699562474117	Random	0.7456027889274623	Random
Non-overlapping template Matching test	0.9999999999999729	Random	0.9999999556775546	Random
Approximate entropy test	1.0	Random	1.0	Random
Cumulative sums test (forward)	0.5998890627269671	Random	0.5754947715401479	Random

The NIST randomness test results (Table 2) for the Speck cipher, after optimisation with the bee swarm and snake algorithms, provide significant insights into the performance and reliability of these cryptographic methods under different conditions. The tests focus on various aspects of randomness, a crucial factor in determining the effectiveness of cryptographic ciphers.

In the frequency (monobit) test and the frequency test within a Block, both algorithms demonstrated random results with the Speck cipher, even though the P-values were higher for the bee swarm optimisation (approximately 0.65) compared with those after SO (around 0.42 and 0.48, respectively). This result suggests that both optimisers can maintain randomness, with the bee Swarm showing slightly higher randomness in these particular tests.

The runs test, which assesses the randomness of sequences produced by the cipher, presented a stark contrast between the two. The Speck cipher optimised with the bee swarm yielded a P-value

indicative of randomness (0.064), whereas the snake optimiser resulted in a P-value of 0.0, suggesting non-randomness and potential vulnerabilities in generating random sequences under the snake optimisation.

In the discrete Fourier transform (spectral) test, both optimisers produced random outcomes, although the bee swarm's P-value of 0.305 suggests a more robust randomness compared with the snake's lower P-value of 0.014. This result indicates a marginal but potentially significant difference in handling frequency components within the encrypted outputs.

The non-overlapping template matching test showed both algorithms yielding highly random P-values, virtually indistinguishable from each other and close to one. This notion implies that both methods are highly effective in ensuring randomness in this aspect, with negligible differences between them.

The cumulative sums tests (forward and backward) also demonstrated excellent performance for both algorithms, with P-values at or near the threshold. However, the results post-SO edged slightly closer to perfect randomness, indicating strong performance in ensuring that the sum of sequential random variables does not significantly deviate from the expected behaviours.

Overall, although the bee swarm and snake optimisers effectively enhance the randomness of the Speck cipher in several tests, the snake optimiser shows particular strengths in the cumulative sums tests. However, the snake optimiser also exhibits a critical weakness in the runs test, suggesting an area where the bee swarm optimiser maintains superior performance. This mixed result highlights the nuanced performance differences between these two optimisation techniques when applied to cryptographic key generation, underscoring the need for careful selection based on specific security requirements.

In Fig. 5, the image displays a user interface for a Firebase Realtime Database project titled 'saja333444'. This database is designed for storing and managing IoT results. The database entry for 'Patient ID: 56' includes various attributes, such as. the first and last names, room temperature, body temperature, Spo2 level and heartbeat rate. These numerical values correspond to patient health parameters that are monitored on a real-time basis. The entry also contains fields, such as 'current hash' and 'previous hash,' which are be used for data integrity or versioning. The dashboard provides a unified, real-time, single point of view of the data acquired from the IoT devices and presents an integrated approach to healthcare monitoring technologies and cloud-based data management.

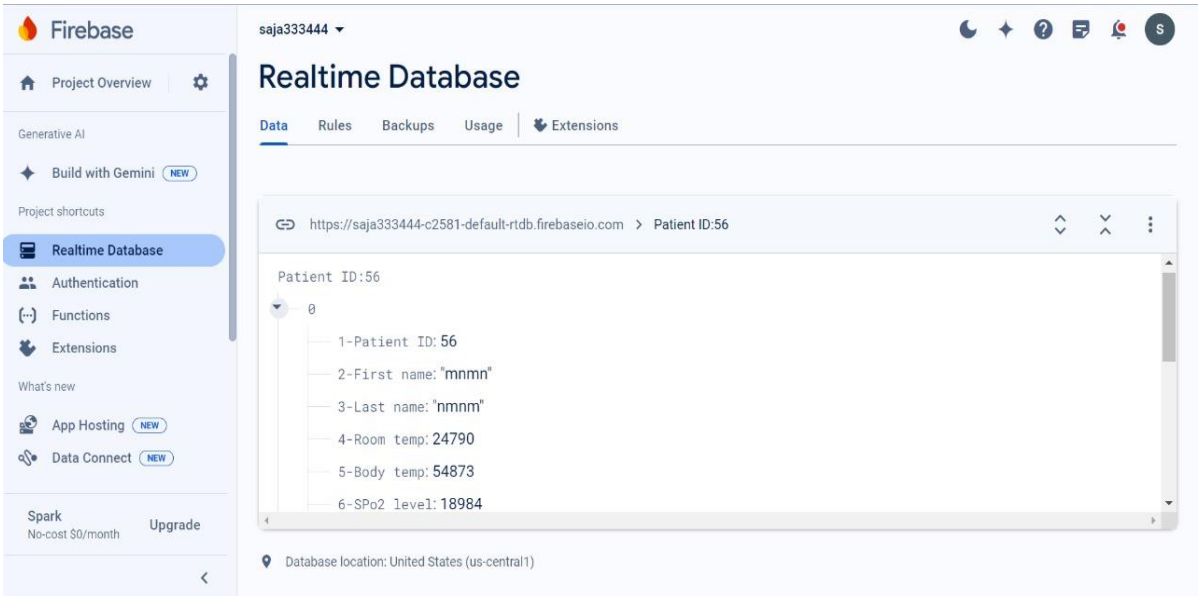

Fig. 5. Real-time database

Table 2,
Comparison with related work

Related work	Main focus	Cryptographic technique
Proposed Work	Secure and lightweight cipher for IoT healthcare using snake key generation	Snake key generator and bee swarm key generator
Ogundokun et al. [11]	Crypto-stegno framework for securing healthcare data in IoT	Crypto-stegno
Abdellatif et al. [12]	BC-based healthcare solution for emergency response and remote monitoring	BC
Pandey et al. [13]	Healthcare cybersecurity architecture using BC, DBN and ResNet	BC, DBN and ResNet
Veeramakali et al. [14]	Cryptographic IoT security authentication scheme using OHE and DNN	OHE and DNN
Abd El-Latif et al. [15]	Secure BC-based e-healthcare architecture using OPSO and ODNN	OPSO and ODNN
Cao et al. [16]	Authentication and encryption system using quantum-inspired quantum walks	Quantum-inspired quantum walks
Shankar et al. [17]	BC-focused ICO specialised in cybersecurity using a random neural network model	BC and random neural network model
Mathews et al. [18]	Improved Two Arch2 solution for scalability and faster BC time	Two Arch2

8. Conclusion

A detailed examination of IoT healthcare security, BC application and cryptographic robustness in multiple stages illustrates the improved data security achieved using novel optimisation methods. Specifically, the SO method surpasses the bee swarm technique in several critical aspects, making it a beneficial selection for BC-based healthcare applications focused on data privacy issues.

The snake key generator demonstrated improved performance in important NIST randomness tests, notably the cumulative sums tests (forward and reverse), which measure the sum of sequential random variables. These results showed how snake maintains high randomness and unpredictability, effectively securing sensitive patient data against various cryptographic attacks.

Furthermore, the SO strategy's resiliency has shown growing robustness against possible cryptographic attacks. The keys generated using this method were virtually impossible to break within a short period (even for those that are longer function only because, even with the efficiency of current algorithms, the quantity of operations required to break the key is out of reach, making them resistant to common attack methodologies, including brute force and attacks against current cryptographic hash functions). In healthcare, ensuring the confidentiality of highly sensitive data is paramount. Furthermore, the snake key generator is computationally efficient, providing for leaner and meaner key creation whilst maintaining security. This factor is important in the IoT landscape, where computational resources are often limited. The key generator implemented in the snake can work in constrained environments, making it a valuable tool for IoT security.

Although the snake key generator efficiently outperformed other routines in various tests, the runs test showed that its randomness is imperfect. This finding implies that even sophisticated iterations of the SO process may encounter problems. Addressing vulnerabilities is essential, as any security solution must be continuously tested against emerging threats and unforeseen scenarios.

References

[1] S. A. Butt, J. L. Diaz-Martinez, T. Jamal, A. Ali, E. De-La-Hoz-Franco, and M. Shoaib, "IoT Smart Health Security Threats," in 2019 19th International Conference on Computational Science and Its Applications (ICCSA) (IEEE, Saint Petersburg, Russia, 2019) pp. 26–31, doi: 10.1109/ICCSA.2019.000-8 .

[2] A. Frimpong, C. Barbosa, and R. A. Abd-Alhameed, "The impact of the internet of things (iot) on healthcare delivery: A systematic literature review." Journal of Techniques 5 (2023), doi: https://doi.org/10.51173/jt.v5i3.1433.

[3] H. F. Jawad, A. Al-Askery, and A. H. Ali, "Design and implementation of a healthcare monitoring system based on lora." Journal of Techniques 4 (2022), doi: https://doi.org/10.51173/jt.v4i4.792.

[4] S. Mohapatra, A. Sahoo, S. Mohanty, and D.

Singh, "IoT Enabled Ubiquitous Healthcare System using Predictive Analytics," Procedia Computer Science 218, 1581–1590 (2023), doi: 10.1016/j.procs.2023.01.136.

[5] Awotunde, J. B., Jimoh, R. G., Folorunso, S. O., Adeniyi, E. A., Abiodun, K. M., & Banjo, O. O. (2021). Privacy and security concerns in IoT-based healthcare systems. In The fusion of internet of things, artificial intelligence, and cloud computing in health care (pp. 105-134). Cham: Springer International Publishing doi: 10.1007/978-3-030-75220-0_6.

[6] Raparthi, M. (2021). Privacy-Preserving IoT Data Management with BC and AI-A Scholarly Examination of Decentralised Data Ownership and Access Control Mechanisms. Internet of Things and Edge Computing Journal, 1(2), 1-10 https://thesciencebrigade.com/iotecj/article/view/131/132.

[7] Abbas, K., Tawalbeh, L. A. A., Rafiq, A., Muthanna, A., Elgendy, I. A., & Abd El-Latif, A. A. (2021). Convergence of blockchain and IoT for secure transportation systems in smart cities. Security and Communication Networks, 2021, 1-13 doi:10.1155/2021/5597679.

[8] Rangappa, J. D., Manu, A. P., Kariyappa, S., Chinnababu, S. K., Lokesh, G. H., & Flammini, F. (2023). A Lightweight Blockchain to Secure Data Communication in IoT Network on Healthcare System. International Journal of Safety & Security Engineering, 13(6) DOI: https://doi.org/10.18280/ijsse.130604.

[9] Perwej, Y., Akhtar, N., Kulshrestha, N., & Mishra, P. (2022). A methodical analysis of medical internet of things (MIoT) security and privacy in current and future trends. Journal of Emerging Technologies and Innovative Research, 9(1), d346-d371 DOI: 10.6084/m9.figshare.JETIR2201346

[10] Popoola, O., Rodrigues, M., Marchang, J., Shenfield, A., Ikpehia, A., & Popoola, J. (2023). A critical literature review of security and privacy in smart home healthcare schemes adopting IoT & blockchain: problems, challenges and solutions. Blockchain: Research and Applications, 100178 doi: https://doi.org/10.1016/j.bcra.2023.100178.

[11] G. N. Nguyen, N. H. Le Viet, M. Elhoseny, K. Shankar, B. B. Gupta et al., "Secure blockchain enabled Cyber-physical systems in healthcare using deep belief network with ResNet model," Journal of Parallel and Distributed Computing, vol. 153, no. 9, pp. 150–160, 2021 doi: https://doi.org/10.1016/j.jpdc.2021.03.011.

[12] G. Kalyani and S. Chaudhari, "An efficient approach for enhancing security in internet of things using the optimum authentication key," International Journal of Computers and Applications, vol. 42, no. 3, pp. 306–314, 2020 doi: https://doi.org/10.1080/1206212X.2019.1619277.

[13] P. Pandey and R. Litoriya, "Securing and authenticating healthcare records through blockchain technology," Cryptology, vol. 44, no. 4, pp. 341–356, 2020 doi: https://doi.org/10.1080/01611194.2019.1706060.

[14] T. Veeramakali, R. Siva, B. Sivakumar, P. S. Mahesh, and N. Krishnaraj, "An intelligent internet of things-based secure healthcare framework using blockchain technology with an optimal deep learning model," The Journal of Supercomputing, vol. 4, no. 9, pp. 1–21, 2021 doi: https://doi.org/10.1007/s11227-021-03637-3

[15] A. A. Abd El-Latif, B. Abd-El-Atty, I. Mehmood, K. Muhammad, V. Andraca et al., "Quantum-inspired blockchain-based cybersecurity: Securing smart edge utilities in IoT-based smart cities," Information doi: https://doi.org/10.1016/j.ipm.2021.102549

[16] W. Serrano, "The blockchain random neural network for cybersecure IoT and 5G infrastructure in smart cities," Journal of Network and Computer Applications, vol. 175, no. 3, pp. 102909, 2021 doi: https://doi.org/10.1016/j.jnca.2020.102909.

[17] B. Cao, X. Wang, W. Zhang, H. Song, and Z. Lv, "A many-objective optimization model of industrial internet of things basedon private blockchain," IEEE Network, vol. 34, no. 5, pp. 78–83, 2020 DOI: 10.1109/MNET.011.1900536.

[18] K. Shankar, M. Elhoseny, E. Perumal, M. Ilayaraja, and K. S. Kumar, "An efficient image encryption scheme based on signcryption technique with adaptive elephant herding optimization," Cybersecurity and Secure Information Systems, vol. 12, no. 4, pp. 31–42, 2019 doi: 10.1007/978-3-030-16837-7_3.

[19] R. S. Mathews, A. N. Maadhuree, R. R. Justus, K. Vishnu, and C. R. Robin, "Fulcrum: Cognitive therapy system for stress relief by emotional perception using DNN," in Int. Conf. on Em doi: https://doi.org/10.1007/978-3-030-32150-5_120.

[20] L. Abualigah, R. A. Zitar, K. H. Almotairi, A. M. Hussein, M. Abd Elaziz, M. R. Nikoo, and A. H. Gandomi, "Wind, solar, and photovoltaic renewable energy systems with and without

energy storage optimization: a survey of advanced machine learning and deep learning techniques," Energies, vol. 15, no. 2, pp. 578, 2022 doi: https://doi.org/10.3390/en15020578.

[21] H. Ahmadi, G. Arji, L. Shahmoradi, R. Safdari, M. Nilashi, and M. Alizadeh, "The application of internet of things in healthcare: a systematic literature review and classification," Univ Access Inf Soc, vol. 18, no. 4, pp. 837–869, 2019 doi: https://doi.org/10.1007/s10209-018-0618-4.

[22] S. Mukherjee and G. P. Biswas, "Networking for IoT and applications using existing communication technology," Egypt Inform J, vol. 19, no. 2, pp. 107–127, 2018 doi: https://doi.org/10.1016/j.eij.2017.11.002.

[23] P. Chanak and I. Banerjee, "Internet-of-things-enabled smartvillages: an overview," IEEE Consum Electron Mag, vol. 10, no. 3, pp. 12–18, 2020 DOI: 10.1109/MCE.2020.3013244.

[24] L. Catarinucci, D. De Donno, L. Mainetti, L. Palano, L. Patrono, M. L. Stefanizzi, and L. Tarricone, "An IoT-aware architecture for smart healthcare systems," IEEE Internet Things J, vol. 2, no. 6, pp. 515–526, 2015 DOI: 10.1109/JIOT.2015.2417684.

[25] T. M. Ghazal, M. K. Hasan, M. T. Alshurideh, H. M. Alzoubi, M. Ahmad, and S. S. Akbar, "IoT for smart cities: machine learning approaches in smart healthcare—a review," Future Internet, vol. 13, no. 8, pp. 218, 2021 doi: https://doi.org/10.3390/fi13080218.

[26] S. Dash, S. K.Shakyawar, M. Sharma, and S. Kaushik, "Big data in healthcare: management, analysis and future prospects," J Big Data, vol. 6, no. 1, pp. 54, 2019 doi: https://doi.org/10.1186/s40537-019-0217-0.

[27] S. Zeadally, F. Siddiqui, Z. Baig, and A. Ibrahim, "Smart healthcare: challenges and potential solutions using internet of things (IoT) and big data analytics," PSU Res Rev, pp. 1–17, 2019 DOI 10.1108/PRR-08-2019-0027.

[28] S. K. Abujayyab, K. H. Almotairi, M. Alswaitti, S. S. A. Amr, A. F. Alkarkhi, E. Taşoğlu, and A. M. Hussein, "Effects of meteorological parameters on surface water loss in Burdur Lake, Turkey over 34 years Landsat Google earth engine time-series," Land, vol. 10, no. 12, pp. 1301, 2021 doi: https://doi.org/10.3390/land10121301.

[29] M. Otair, A. Alhmoud, H. Jia, M. Altalhi, A. M. Hussein, and L. Abualigah, "Optimized task scheduling in cloud computing using improved multi-verse optimizer," Clust Comput, 2022 doi: https://doi.org/10.1007/s10586-022-03650-y.

[30] M. M. Dhanvijay and S. C. Patil, "Internet of things: a survey of enabling technologies in healthcare and its applications," Comput Netw, vol. 153, pp. 113–131, 2019 doi: https://doi.org/10.1016/j.comnet.2019.03.006.

[31] J. T. Kelly, K. L. Campbell, E. Gong, and P. Scuffham, "The Internet of Things: Impact and implications for health care delivery," J Med Internet Res, vol. 22, no. 11, pp. e20135, 2020 doi: 10.2196/20135.

[32] A. Thamara, M. Elsersy, A. Sherif, H. Hassan, O. Abdelsalam, and K. H. Almotairi, "A novel classification of machine learning applications in healthcare," in 2021 3rd IEEE Middle East and North Africa COMMunications conference (MENACOMM), IEEE, pp. 80–85, 2021 DOI: 10.1109/MENACOMM50742.2021.9678232.

[33] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Decentralized Bus. Rev., vol. 5, p. 21260, Oct. 2008 https://www.researchgate.net/publication/228640975_Bitcoin_A_Peer-to-Peer_Electronic_Cash_System.

[34] Y. Wang, Z. Su, J. Ni, N. Zhang, and X. Shen, "Blockchain-empowered space-air-ground integrated networks: Opportunities, challenges, and solutions," IEEE Commun. Surveys Tuts., vol. 24, no. 1, pp. 160–209, 1st Quart., 2022 DOI: 10.1109/COMST.2021.3131711.

[35] J. Xie, H. Tang, T. Huang, F. R. Yu, R. Xie, J. Liu, and A. Liu, "A survey of blockchain technology applied to smart cities: Research issues and challenges," IEEE Commun. Surveys Tuts., vol. 21, no. 3, pp. 2794–2830, 3rd Quart., 2019 DOI: 10.1109/COMST.2019.2899617.

[36] R. Belchior, A. Vasconcelos, S. Guerreiro, and M. Correia, "A survey on blockchain interoperability: Past, present, and future trends," ACM Comput. Surv., vol. 54, no. 8, pp. 1–41, Nov. 2022 doi: https://doi.org/10.1145/3471140.

[37] G. Zyskind and O. Nathan, "Decentralizing privacy: Using blockchain to protect personal data," in Proceedings of the 2015 IEEE Security and Privacy Workshops, San Jose, CA, USA, 21–22 May 2015; IEEE: New York, NY, USA, 2015; pp. 180–184 doi: https://doi.org/10.1145/3471140.

[38] R. Hassan, A. Ahmed, and N. E. Osman, "Enhancing security for IPv6 neighbor discovery protocol using cryptography," Am. J. Appl. Sci., vol. 11, pp. 1472–1479, 2014 DOI: 10.3844/ajassp.2014.1472.1479.

[39] G. Aruna, M. K. Hasan, S. Islam, K. G. Mohan, P. Sharan, and R. Hassan, "Cloud to cloud data migration using self sovereign identity for 5G and beyond," Clust. Comput., vol. 25, pp. 2317–2331, 2021 doi: https://doi.org/10.1007/s10586-021-03461-7.

[40] S. Bhattacharya, A. Singh, and M. Hossain, "Strengthening public health surveillance through blockchain technology," AIMS Public Health, vol. 6, pp. 326–333, 2019 DOI:10.3934/publichealth.2019.3.326.

[41] R. MIslam, M. Rahman, M. Mahmud, M. Rahman, and M. H. S. Mohamad, "A Review on blockchain security issues and challenges," in Proceedings of the 2021 IEEE 12th Control and System Graduate Research Colloquium (ICSGRC), Shah Alam, Malaysia, 7 August 2021; IEEE: New York, NY, USA, 2021; pp. 227–232 DOI:10.1109/ICSGRC53186.2021.9515276.

[42] Sullivan, "Blockchain-based identity: The advantages and disadvantages," in Blockchain and the Public Sector; Springer: Cham, Switzerland, 2021; pp. 197–218 doi: https://doi.org/10.1007/978-3-030-55746-1.

[43] Daemen, J. and Rijmen, V., 1999. AES proposal: Rijndael https://www.researchgate.net/publication/2237728_AES_proposal_rijndael.

[44] Sleem, L., & Couturier, R. (2021). Speck-R: An ultra-light-weight cryptographic scheme for Internet of Things. Multimedia Tools and Applications, 80(11), 17067-17102 doi: https://doi.org/10.1007/s11042-020-09625-8

تطبيق شيفرة آمنة وخفيفة الوزن للتطبيقات الصحية في إنترنت الأشياء المقيدة بالموارد باستخدام توليد مفاتيح الأفعى

سجى كريم إسماعيل[1*]، محمد إبراهيم شجاع[2]، أحمد بهاء الدين عبد ألوهاب[3]
[1،2] كلية الهندسة الكهربائية التقنية، الجامعة التقنية الوسطى، بغداد، العراق
[3] الكلية التقنية للإدارة، الجامعة التقنية الوسطى، بغداد، العراق
[*]البريد الالكتروني: bbc0071@mtu.edu.iq

المستخلص

إن دمج إنترنت الأشياء في الرعاية الصحية قد غير الطريقة التي يتم بها مراقبة المعايير الفسيولوجية، مما يعزز بشكل كبير قدرات التشخيص الطبي ورعاية المرضى. يستكشف هذا البحث تطبيق تقنيات التشفير المتقدمة لتأمين أنظمة الرعاية الصحية لإنترنت الأشياء، وخاصة من خلال تقنية البلوك تشين. يركز البحث على تقييم فعالية مولدي المفاتيح التشفيرية - مولد مفتاح سرب النحل ومولد مفتاح الأفعى - في تعزيز أمان البيانات داخل إطار عمل قائم على البلوك تشين لمراقبة المرضى عن بعد. يجري دراستنا سلسلة من اختبارات العشوائية لـــ NIST لمقارنة أداء هذه المولدات من حيث العشوائية، والمقاومة للهجمات التشفيرية، والكفاءة الحسابية. تظهر النتائج أن مولد مفتاح الأفعى بشكل عام يقدم أداءً أفضل فيما يتعلق بزيادة العشوائية في توليد المفاتيح وتغطية أمان أفضل للمساحة التوجيهية. يتبين أن هذا مناسب جداً لتأمين بيانات الرعاية الصحية المهمة، وهو جانب رئيسي نظراً لتعقيد أجهزة إنترنت الأشياء، التي قد تنقل بيانات صحة المرضى المهمة. يغطي البحث أيضاً بنية التشفير ذات الطبقتين المستخدمة لتأمين البيانات أثناء السكون وأثناء الحركة بين النقطة التي يتم جمعها بواسطة المستشعر إلى التخزين طويل الأمد. هذا النهج ضروري للحفاظ على خصوصية وسلامة بيانات المرضى، وهو أمر بالغ الأهمية لضمان الثقة في أنظمة الرعاية الصحية لإنترنت الأشياء. يهدف هذا العمل إلى أن يكون دراسة شاملة ومقارنة لتسهيل التطور المستمر لحلول التشفير الآمنة والفعالة والقابلة للتوسع في مجال الرعاية الصحية لإنترنت الأشياء، مما يبرز أهمية اختيار تقنية توليد المفتاح الأقوى للتعامل مع التحديات الأمنية الفريدة لهذا المجال الجديد.