INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Attack Analysis on Hybrid-SIMON-SPECKey Lightweight Cryptographic Algorithm for IoT Applications

Rahul P Neve[1]*, **Rajesh Bansode**[2]

1 Research Scholar, Department of Information Technology, Thakur College of Engineering and Technology, Mumbai, Maharashtra, India
2 Professor, Department of Information Technology, Thakur College of Engineering and Technology, Mumbai, Maharashtra, India

OPEN ACCESS

Received: 06-11-2023

Accepted: 12-02-2024

Published: 27-02-2024

Citation: Neve RP, Bansode R (2024) Attack Analysis on Hybrid-SIMON-SPECKey Lightweight Cryptographic Algorithm for IoT Applications. Indian Journal of Science and Technology 17(10): 932-940. https://doi.org/10.17485/IJST/v17i10.2811

* **Corresponding author**.

rahulneve@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objective: To perform attack analysis on new developed hybrid-SIMON-SPECKey lightweight cryptographic algorithms and compare its strength with existing SIMON and SPECK Lightweight cryptographic algorithm. **Methods:** A hybrid-SIMON-SPECKey algorithm is the combination of round function of SIMON and key scheduling of SPECK algorithm. Both SIOMN & SPECK algorithm are used for securing resource constrained devices. In this research work, avalanche effect method is used to analyze attack resistance property of algorithm. **Findings:** Newly developed Hybrid algorithm shows better results in terms of execution time and memory consumption. As compared to SIMON, hybrid version of algorithm consumes 50% less time and 20% less memory, which makes it efficient. Strict Avalanche criteria for SIMON is 89%, that of SPECK is 90% and in case of hybrid algorithm, it is 90% at start position but when the character is flipped or changed at the end position of plain text then SAC is more (87%) in case of hybrid algorithm as compared as SIMON and SPECK algorithms. Hence, newly developed algorithm showed improved results with equally resistance to the attack as compared to SIMON & SPECK. **Novelty and applications:** The novelty lies in the creation of a hybrid lightweight cryptographic algorithm that combines the feistel structure of SIMON with the key scheduling function of SPECK. This hybrid approach aims to leverage the strengths of both algorithms, potentially providing a more robust and efficient solution for resource-constrained IoT devices. In section 3.1 comparative analysis is done which show that hybrid algorithm outperforms in term of time and memory consumption as well a strength of newly developed hybrid algorithm is evaluated using avalanche effect which shows that it is at par with base algorithms.

Keywords: Attack Analysis; Cipher Code; Decryption; Encryption; Lightweight Cryptography; Iot Devices; And Resource Constraint Devices

1 Introduction

In an era where the Internet of Things (IoT) is being redefined by lightweight cryptography, the symbiotic relationship between IoT devices and lightweight cryptography is observed as a critical cornerstone of the hyper connected world. Lightweight cryptography is served as the guardian of this new digital frontier, offering a delicate balance between the fortification of security in IoT ecosystems and the preservation of the essential agility and efficiency demanded by these devices(1). "Lightweight cryptography" denotes the development and application of cryptographic methods tailored to operate efficiently in resource-constrained settings, such as embedded systems, sensor networks, and low-power devices. These devices typically grapple with limited computational power, storage capacity, and energy resources, rendering conventional cryptographic techniques impractical. Traditional cryptographic algorithms, known for their energy and memory-intensive nature, aren't suitable for power-constrained devices, and their time-consuming nature adds to the challenge(2).

The diverse array of lightweight cryptographic methods created thus far comes with their own strengths and weaknesses. Given the varying requirements across applications, it's challenging to identify a universally applicable solution. Present lightweight cryptography solutions often encounter issues related to key size and encryption/decryption speed(3).

Developing novel strategies and techniques is essential to devise computationally efficient algorithms that demand minimal storage, power, and time. Equally important is ensuring that these lightweight algorithms maintain robust security despite their limited resources. Striking a delicate balance between security levels and resource demands is the key. Furthermore, lightweight cryptography underscores usability and compatibility with existing protocols and systems, contributing to its significance in the realm of secure communication for resource-constrained environments(4–6) .

SIMON and SPECK are lightweight block ciphers designed specifically for resource-constrained devices like IoT sensors and wearable's(7). These encryption algorithms are optimized for efficiency, offering robust security while demanding minimal computational resources. SIMON and SPECK strike a balance between low overhead and cryptographic strength, making them ideal choices for devices with limited processing power and memory(8).

IoT devices often operate in challenging environments and are exposed to various security threats. Therefore, it is crucial to ensure the robustness and resilience of cryptographic algorithms specifically designed for these platforms. Attack analysis in the context of lightweight cryptography focuses on assessing the security and vulnerability of cryptographic algorithms when deployed in these constrained environments. Such analysis is essential to identify potential weaknesses and susceptibilities that adversaries might exploit(9,10). By understanding the attack vectors and the extent to which an algorithm can withstand various forms of attacks, researchers and practitioners can make informed decisions about the suitability of a cryptographic solution for their specific use cases(11).

2 Methodology

In this research paper, hybrid SIMON SPECK algorithm explained which is designed using feistel structure of SIMON and key scheduling function of SPECK algorithm. Experimentally hybrid algorithm shows better performance in terms of execute time and memory consumption. All three algorithms (SIMON, SPECK, hybrid-SIMON-SPECKey) were tested for avalanche effect to observe the attack resistance property(12).

2.1 Design of proposed Salted-Hybrid-SIMON-SPECKey

Designing a hybrid lightweight cryptography scheme by adding cryptographic random salt with plain text and combining the round function of the SIMON algorithm with the key scheduling of the SPECK algorithm involves integrating the respective components of each algorithm. Below is a high-level description of the hybrid design, considering the round function of SIMON and the key scheduling of SPECK. In newly developed technique the plain text is appended with randomize cryptographic salt function and new plain text generated as shown in Figure 1 . New plain text is divided into block of 64-bit each and each block is further divided into two halves as Xi and Yi.

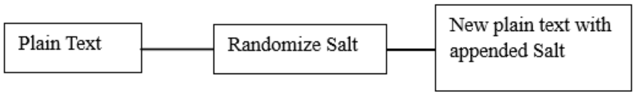


Fig 1. Plain text appended with cryptographic random number

A salt is a random value that is unique to each instance of data being processed, and it helps prevent the use of precomputed tables (rainbow tables) for attacks like dictionary attacks and rainbow table attacks.

The hybrid SIMONSPECK key supports block size and key size as shown in Table 1 . In this paper we implement the hybrid algorithm with block size of 64 -bit and key size 128 bits and number of rounds are 32.

Table 1. Blocksize and key size in bits for hybrid algorithm

Block Size	Key Size
128	120 , 192,256
96	96, 144
64	96, 128
48	72, 96
32	64

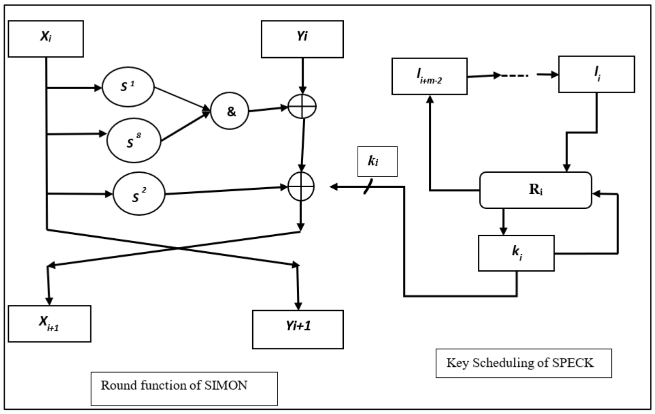


Fig 2. Hybrid Cipher module

2.2 Round function of SIMON

In each encryption round of SIMON, a series of operations are performed on the input data to introduce confusion and diffusion, which are fundamental principles in modern symmetric encryption. The round function typically includes bitwise operations such as XOR, AND, OR, and rotations to shuffle and mix the bits of the data. This process helps to ensure that even small changes in the input data result in significant changes in the output, enhancing the security of the algorithm(13,14).

The round function also involves the use of round keys, which are derived from the original encryption key and are used to modify the data at each round. These round keys add an additional layer of complexity and security to the encryption process.

Given a 64-bit block of data X and a round key K, the SIMON round function can be represented as:

XOR X with a left-circular shift of X by 3 bits:

Step 1: X XOR (X <<< 3)

Perform a bitwise AND between the result of step 1 and a left-circular shift of the result by 1 bit:

Step 2: (Step 1) AND ((Step 1) <<< 1)

XOR the result of step 2 with the round key K:

Step 3: (Step 2) XOR K

Finally, XOR the result of step 3 with a left-circular shift of X by 1 bit:

SIMON round result: (Step 3) XOR (X <<< 1)

2.3 Key scheduling of SPECK

The key scheduling algorithm in SPECK block ciphers involves dividing the user-provided secret key into key words, applying bitwise rotations to these key words that are determined by the round number, mixing them using bitwise logical operations, and generating a set of round keys. The specific number of rounds and key words, as well as the rotation and mixing operations, depend on the chosen SPECK variant and block size. The key scheduling algorithm is relatively simple, making SPECK suitable for lightweight, resource-constrained devices, with the overall security of the cipher dependent on the specific variant's design parameters, the strength of the user-provided key, and proper key management practices [15,16] .

The SPECK key scheduling algorithm is relatively simple and can be expressed mathematically as follows:

Let 'K' be the original key, and 'n' and 'm' be the block size and key size parameters, respectively.

Divide the original key 'K' into 'n'-bit words, denoted as 'K0', 'K1', 'K2', ..., 'Kn-1', where 'n' is typically half of the block size 'm'.

Initialize an array of round keys, denoted as 'RoundKeys', with 'RoundKeys[0]' to 'RoundKeys[T-1]', where 'T' is the total number of rounds.

For round 'i' from 0 to 'T-1', calculate the round keys using the following recursive formula:

For 'i= 0', use the original key: 'RoundKeys[i] = K'.

For 'i > 0', calculate the round key as follows:

'RoundKeys[i] = (RoundKeys[i-1] + ROR(RoundKeys[i-1], A)) XOR i XOR K[i % n]'

Where:

'+' denotes bitwise addition (mod 2^n),

'ROR' represents a right-circular bitwise rotation operation,

'A' is a constant (typically 7 for SPECK32/64 and 8 for SPECK64/128),

'i' is the round number, and

'K [i % n]' is the appropriate word from the original key, cycling through the words in a circular manner.

The round keys are generated for each round, incorporating the original key and using a circular rotation and bitwise XOR operations based on the round number and the original key words. The round keys can then be used in the encryption or decryption process during each round [16,17] .

2.4 Attack resistance analysis using avalanche effect

Attack analysis of lightweight cryptography is a critical component of ensuring the security of cryptographic algorithms and systems designed for resource-constrained environments. Lightweight cryptography is specifically tailored to provide efficient encryption and decryption on devices with limited computational power, memory, and energy resources. However, the constrained nature of these devices also makes them potential targets for adversaries seeking to compromise their security. Therefore, comprehensive attack analysis is indispensable to assess the robustness of lightweight cryptographic schemes [18,19] . Attack analysis involves a systematic evaluation of a cryptographic algorithm's vulnerability to a wide range of potential threats and exploits. Common categories of attacks include:

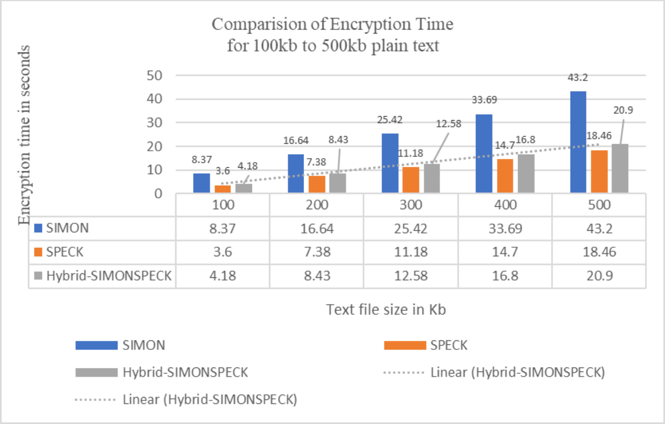
- **Differential and Linear Cryptanalysis:** These are analytical techniques that aim to discover patterns or relationships between the plaintext, ciphertext, and key. Attackers look for specific differentials or linear equations that may reveal key bits or reduce the security of the encryption [20,21] .
- **Side-Channel Attacks:** These attacks exploit information that is inadvertently leaked during the cryptographic operation, such as power consumption, electromagnetic radiation, or timing. Analyzing these side-channel leaks can help attackers recover sensitive information like encryption keys [22].
- **Fault Attacks:** Fault attacks involve intentionally introducing errors or faults during the execution of a cryptographic algorithm. By analyzing how these faults affect the algorithm's behavior, attackers can gain insights into its operation and potentially compromise its security.
- **Known-Plaintext and Chosen-Plaintext Attacks:** In these attacks, attackers have access to either known plaintext-ciphertext pairs or the ability to choose plaintexts and observe their corresponding ciphertexts. This information can be used to deduce the encryption key.

A comprehensive attack analysis assesses a lightweight cryptographic algorithm's resistance to these and other threats, helping cryptographers and security experts identify vulnerabilities and develop countermeasures to mitigate them. It is essential for ensuring the algorithm's real-world security, as resource-constrained devices often handle sensitive data in critical applications, such as the Internet of Things (IoT), smart cards, and embedded systems. By conducting thorough attack analysis, the cryptographic community can continue to enhance the reliability and security of lightweight cryptography in an ever-evolving threat landscape [23,24].

3 Results and Discussion

3.1 Comparative analysis of SIMON, SPECK & Hybrid Algorithm

Experiment is carried on raspberry pi model b controller with 1 GB of RAM. Plain text file of 100kb to 500kb are passed through all three algorithms (SIMON, SPECK, Hybrid) and experimental results are noted down.

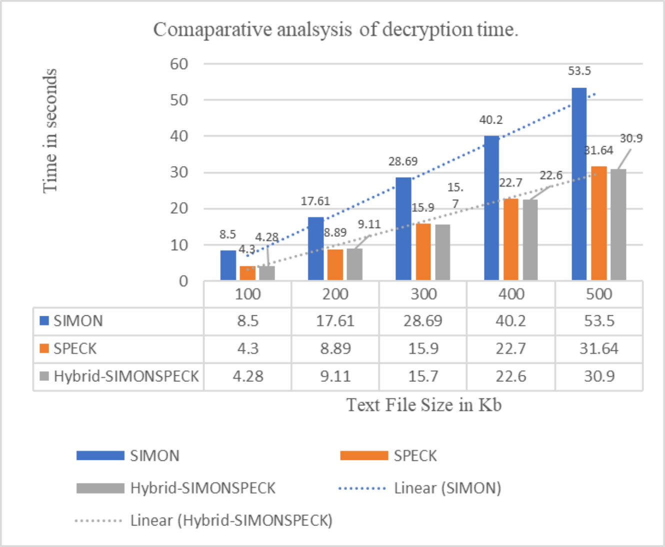


	100	200	300	400	500
SIMON	8.37	16.64	25.42	33.69	43.2
SPECK	3.6	7.38	11.18	14.7	18.46
Hybrid-SIMONSPECK	4.18	8.43	12.58	16.8	20.9

Fig 3. Execution time of Encryption function in seconds

From Figure 3 data, it can be seen that SPECK and Hybrid-SIMONSPECK shows 50 % less time consumption as compared to SIMON. Additionally, as the data size increases, the encryption time also increases for all three algorithms, which is expected as larger data requires more processing time for encryption.

Figure 4 depicts decryption time in seconds. For hybrid algorithm it is almost half of SIMON algorithm. Hence hybrid algorithm outperforms the SIMON algorithm.



	100	200	300	400	500
SIMON	8.5	17.61	28.69	40.2	53.5
SPECK	4.3	8.89	15.9	22.7	31.64
Hybrid-SIMONSPECK	4.28	9.11	15.7	22.6	30.9

Fig 4. Execution time of Decryption function in seconds

Figure 3 shows that SPECK algorithm consume almost 20% more memory during encryption process as compare to SIMON & hybrid algorithm. Hence, comparing Figures 3, 4 and 5 , it can be said that hybrid algorithm shows improved performance. Now its time to check strength of hybrid algorithm. For this purpose, avalanche effect is used in this research work.

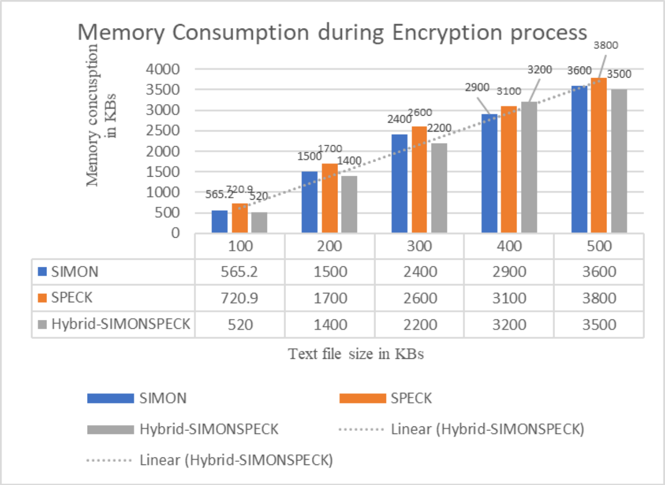


Fig 5. Memory usage in KBs during encryption process

3.2 Attack Analysis by using Strict Avalanche Criterion (SAC) test for the SIMON, SPECK & Hybrid algorithm

In cryptography, the avalanche effect is a fundamental concept that describes how a small change in the input data or key of an encryption algorithm should result in a significantly different and unpredictable output (ciphertext). It implies that even a minor alteration in the input should cause a major and chaotic change in the encrypted output, making it challenging for an attacker to discern any patterns or relationships between the original and encrypted data.

The avalanche effect is a crucial property because it ensures the security of encryption algorithms by preventing attackers from exploiting similarities or regularities between different inputs and their corresponding ciphertexts. It adds a layer of complexity and randomness to the encryption process, making it difficult to reverse-engineer the key or deduce the original message from the encrypted data[25,26].

1 kb sample file is tested by changing 1st character. Plan Text file is passed through SIMON , SPECK and HybridSIMONSPECKey algorithm and calculated avalanche effect. Below is the sample file data.

● Lightweight Cryptographic algorithm is one of the latest research topic. This file will be tested for avalanche effect on the various algorithms such as SIMON, SPECK , HybridSIMONSPECKey , SALTED-HybridSIMONSPECKey. Test Case will be Replacing the first charatcter by adjustent alphabet.Certainly! Here's a brief overview of lightweight cryptography:Lightweight cryptography refers to a subfield of cryptography that focuses on designing cryptographic algorithms and protocols that are optimized for resource-constrained devices, such as low-power microcontrollers, RFID tags, and IoT (Internet of Things) devices. These devices often have limited processing power, memory, and energy resources, making traditional cryptographic algorithms less suitable for their use.Key characteristics of lightweight cryptography include:Low Resource Usage: Lightweight cryptographic algorithms are designed to minimize the consumption of computational resources, memory, and power. They aim to provide security with minimal overhead...

To satisfy the SAC, a change in any single bit of the input (plaintext or key) should, on average, result in exactly half of the output bits changing with a probability of 0.5. This ensures that a small modification in the input has a highly unpredictable and diffusive impact on the output, contributing to the algorithm's resistance against differential and linear cryptanalysis.

Stage 1: Above sample data of 1kB is passed through SIMON, SPECK and Salted HybridSIMONSPECKey and cipher text is obtained respectively.

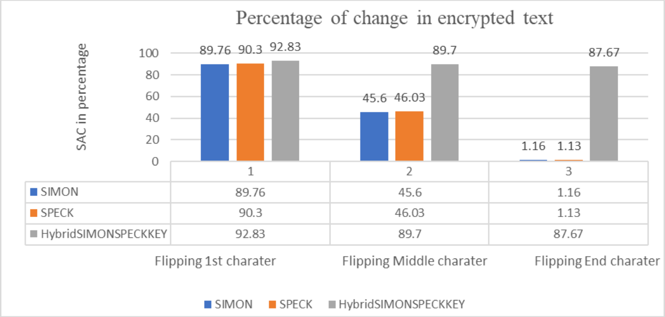
Stage 2: First letter 'L' of above sample data is replaced by 'K' and again passed through SIMON, SPECK & SaltedHybridSIMONSPECKey , now second set of cipher text is obtained.

Stage 3: Compare cipher text obtained in stage 1 with the cipher text obtained in stage 2 of respective algorithms using hamming distance code

The SAC for the LWC algorithms formally expressed as follows. Let X and X' be input pair that differs only in one bit and let Y and Y' be the corresponding output pairs. The SAC is satisfied if, for every pair of differing input bits i:

$$Pr\left(Y_i \oplus Y_i' = 1\right) = \frac{1}{2}$$

In other words, the probability that the i-th bit of the output changes due to a one-bit difference in the input is 0.5.



	1	2	3
SIMON	89.76	45.6	1.16
SPECK	90.3	46.03	1.13
HybridSIMONSPECKKEY	92.83	89.7	87.67

Fig 6. Comparative analysis of avalanche effect on SIMON,SPECK, Hybrid Algorithm

Figure 6 shows Strict avalanche effect by calculating hamming distance between cipher text of SIMON , SPECK & Hybrid and cipher text obtained after changing first character of plain text.

hybrid-SIMON-SPECKey algorithm shows improved performance in terms of execution time and memory usage and also fulfill the requirement of recourse constrained devices. As performance is improved so it become necessity to check whether security parameter of algorithm. Avalanche effect is one of the techniques to check the strength of cryptographic algorithm by calculating hamming distance. In this research work 1kb text file is passed through SIMON, SPECK & hybrid algorithm and cipher text is generated respectively. Then first character of plain text is replaced by adjacent alphabet and cipher text is generated by passing this plain text file to all algorithm respectively. After this first cipher text is compared with newly generated cipher text and change in the character is noted.

- **Start Position:**

SIMON (89.76%): The start position for SIMON shows that changing one character results in a 89.76% change in the output. This indicates a relatively high avalanche effect, which aligns with the desirable property of SAC.

SPECK (90.3%): Similarly, SPECK exhibits a high avalanche effect at the start position with a 90.3% change in the output, showing that the algorithm is responsive to changes in the input.

HybridSIMONSPECKKEY (92.83%): The HybridSIMONSPECKKEY demonstrates the highest avalanche effect at the start position, with a 92.83% change in the output. This suggests that the combination of SIMON and SPECK, or additional keying, enhances the strict avalanche behavior.

- **Center Position:**

SIMON (45.6%): At the center position, SIMON exhibits a lower avalanche effect compared to the start position, with a 45.6% change in the output. This might indicate a decrease in sensitivity to changes in this scenario.

SPECK (46.03%): SPECK also shows a reduced avalanche effect at the center position, with a 46.03% change in the output. The decrease in percentage suggests a similar trend to SIMON.

HybridSIMONSPECKKEY (89.7%): Interestingly, the HybridSIMONSPECKKEY exhibits a higher avalanche effect at the center position compared to SIMON and SPECK individually. This might suggest that the hybrid combination has specific characteristics in this scenario.

- **End Position:**

SIMON (1.16%): At the end position, SIMON shows a significantly lower avalanche effect, with only a 1.16% change in the output. This could indicate that changes at the end position have a minimal impact on the output.

SPECK (1.13%): Similarly, SPECK exhibits a low avalanche effect at the end position, with a 1.13% change in the output. This suggests reduced sensitivity to changes in this context.

HybridSIMONSPECKKEY (87.67%): The HybridSIMONSPECKKEY, in contrast, demonstrates a very high avalanche effect at the end position, with an 87.67% change in the output. This is a notable deviation from SIMON and SPECK individually, suggesting a unique behavior in this scenario.

Strong diffusion of hybridSIMONSPECKey, as indicated by a high avalanche effect, contribute to the resistance of following cryptographic attacks.

- **Resistance to Differential Cryptanalysis:** A high avalanche effect makes it more challenging for an attacker to exploit differential characteristics in the algorithm, enhancing resistance to differential cryptanalysis.
- **Resistance to Statistical Attacks:** A high avalanche effect enhances the statistical properties of the algorithm, making it more resistant to attacks that rely on detecting patterns or biases in the output.
- **Resistance to Known-Plaintext Attacks**: Strong diffusion properties can resist known-plaintext attacks by ensuring that even small changes in the plaintext result in extensive changes in the ciphertext.
- **Protection Against Birthday Attacks**: The avalanche effect contributes to the strength of the algorithm against birthday attacks, where an attacker tries to find two different inputs producing the same output.
- **Robustness Against Side-Channel Attacks:** While not a direct resistance, a high avalanche effect can indirectly contribute to the robustness of an algorithm against certain side-channel attacks by minimizing information leakage through side channels.
- **Non-linearity and Algebraic Attacks:** The high avalanche effect often correlates with strong non-linearity, making the algorithm more resistant to algebraic attacks that aim to exploit mathematical relationships.

4 Conclusion

The research introduces a novel Hybrid Lightweight Cryptographic (LWC) algorithm by combining the key scheduling technique of SPECK with the rounds of SIMON. This hybrid approach exhibits improved performance in terms of time, energy, and memory consumption. The claim of consuming 50% less time and energy compared to individual algorithms (SIMON and SPECK) suggests a noteworthy advancement in the design of lightweight cryptographic solutions for resource-constrained environments.

Additionally, the research highlights the importance of the avalanche effect for security against various attacks, including differentials. The observation that the Hybrid algorithm maintains an avalanche effect strength equivalent to SIMON and SPECK, with a 90% hamming distance, further contributes to the novelty of the proposed algorithm. The promising results of the Hybrid LWC algorithm open up prospects for its application in real-world scenarios, particularly in IoT devices where resource constraints are prevalent. The improved efficiency in terms of time, energy, and memory consumption positions the algorithm as a potential candidate for secure communication in lightweight and energy-efficient IoT applications.

The strong avalanche effect observed in the hybrid algorithm enhances its resistance against differential attacks, adding to its potential for securing sensitive data in a variety of cryptographic applications.

1. New Hybrid Algorithm is being developed by using key scheduling of SPECK and Rounds of SIMON. This new Hybrid LWC algorithm shows good results in terms of time, energy and memory consumption. It consumes 50% less time as well as energy as compared.
2. The provided percentages align with the SAC criteria, showcasing strong avalanche effects, particularly at the start position, for all three algorithms. The hybrid combination, HybridSIMONSPECKKEY, demonstrates enhanced sensitivity, potentially providing increased diffusion and security.

References

1) Neve R, Bansode R, Kaul V. Novel Lightweight Approach to Perform Cryptography for Data Security & Privacy in IoT Mobile Devices. *International Journal of Intelligent Systems and Applications in Engineering*. 2012;11(9s):822–830. Available from: https://ijisae.org/index.php/IJISAE/article/view/3270.
2) Rana M, Mamun Q, Islam R. Lightweight cryptography in IoT networks: A survey. *Future Generation Computer Systems*. 2022;129:77–89. Available from: https://doi.org/10.1016/j.future.2021.11.011.
3) Enriquez M, Garcia DW, Arboleda E. Enhanced Hybrid Algorithm of Secure and Fast Chaos-based, AES, RSA and ElGamal Cryptosystems. *Indian Journal of Science and Technology*. 2017;10(27):1–14. Available from: https://doi.org/10.17485/ijst/2017/v10i27/105001.
4) Caraveo-Cacep MA, Vázquez-Medina R, Zavala AH. A survey on low-cost development boards for applying cryptography in IoT systems. *Internet of Things*. 2023;22. Available from: https://doi.org/10.1016/j.iot.2023.100743.
5) Chaudhary RRK, Chatterjee K. A lightweight security framework for electronic healthcare system. *International Journal of Information Technology*. 2022;14(6):3109–3121. Available from: https://doi.org/10.1007/s41870-022-01034-4.
6) Kaur J, Kumar KRR. Analysis of Avalanche effect in Cryptographic Algorithms. In: 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO). IEEE. 2022;p. 1–4. Available from: https://doi.org/10.1109/ICRITO56286.2022.9965127.
7) Neve R, Bansode R. Review of Lightweight Cryptography for Secure Data Transmission in Resource Constraint Environment of IoT. 2022. Available from: https://doi.org/10.1201/9781003277217.
8) Loai, Tawalbeh, Alicea M, Alsmadi I. New and Efficient Lightweight Cryptography Algorithm for Mobile and Web Applications. *Procedia Computer Science*. 2022;203:111–118. Available from: https://doi.org/10.1016/j.procs.2022.07.016.

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

10) Tabash FK, Izharuddin M, Tabash M. Encryption techniques for H.264/AVC videos: A literature review. *Journal of Information Security and Applications*. 2019;45:20–34. Available from: https://doi.org/10.1016/j.jisa.2019.01.001.

11) Alhirabi N, Rana O, Perera C. Security and Privacy Requirements for the Internet of Things. *ACM Transactions on Internet of Things*. 2021;2(1):1–37. Available from: https://doi.org/10.1145/3437537.

12) Sall S, Bansode R. Lightweight Cryptography Using Pairwise Key Generation and Malicious Node Detection in Large Wireless Sensor Network. *Indian Journal Of Science And Technology*. 2023;16(36):3002–3008. Available from: https://doi.org/10.17485/IJST/v16i36.2503.

13) Sanap SD, More V. Performance Analysis of Encryption Techniques Based on Avalanche effect and Strict Avalanche Criterion. In: 2021 3rd International Conference on Signal Processing and Communication (ICPSC). IEEE. 2021;p. 676–679. Available from: https://doi.org/10.1109/ICSPC51351.2021.9451784.

14) Karie NM, Sahri NM, Yang W, Valli C, Kebande VR. A Review of Security Standards and Frameworks for IoT-Based Smart Environments. *IEEE Access*. 2021;9:121975–121995. Available from: https://doi.org/10.1109/ACCESS.2021.3109886.

15) Hassija V, Chamola V, Saxena V, Jain D, Goyal P, Sikdar B. A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures. *IEEE Access*. 2019;7:82721–82743. Available from: https://ieeexplore.ieee.org/document/8742551.

16) Jaigirdar FT, Tan B, Rudolph C, Bain C. Security-Aware Provenance for Transparency in IoT Data Propagation. *IEEE Access*. 2023;11:55677–55691. Available from: https://ieeexplore.ieee.org/document/10138384.

17) Caudhari A, Bansode R. Securing IoT Devices Generated Data Using Homomorphic Encryption. In: Patil, M, editors. Intelligent Computing and Networking;vol. 146. Springer Singapore. 2021;p. 219–226. Available from: https://doi.org/10.1007/978-981-15-7421-4_20.

18) Zitouni N, Sedrati M, Behaz A. LightWeight energy-efficient Block Cipher based on DNA cryptography to secure data in internet of medical things devices. *International Journal of Information Technology*. 2024;16(2):967–977. Available from: https://doi.org/10.1007/s41870-023-01580-5.

19) Li H, Yang G, Ming J, Zhou Y, Jin C. Transparency order versus confusion coefficient: a case study of NIST lightweight cryptography S-Boxes. *Cybersecurity*. 2021;4(1):35–35. Available from: https://doi.org/10.1186/s42400-021-00099-1.

20) Cherbal S, Zier A, Hebal S, Louail L, Annane B. Security in internet of things: a review on approaches based on blockchain, machine learning, cryptography, and quantum computing. *The Journal of Supercomputing*. 2024;80(3):3738–3816. Available from: https://doi.org/10.1007/s11227-023-05616-2.

21) Neve RP, Bansode R. Performance Evaluation of Lightweight ASCON-HASH Algorithm for IoT Devices. In: Balas, E V, Semwal, B V, Khandare, A, editors. Intelligent Computing and Networking;vol. 699. Springer Nature Singapore. 2023;p. 355–366. Available from: https://doi.org/10.1007/978-981-99-3177-4_25.

22) Liao B, Ali Y, Nazir S, He L, Khan HU. Security Analysis of IoT Devices by Using Mobile Computing: A Systematic Literature Review. *IEEE Access*. 2020;8:120331–120350. Available from: https://doi.org/10.1109/ACCESS.2020.3006358.

23) Cook J, Rehman SU, Khan MA. Security and Privacy for Low Power IoT Devices on 5G and Beyond Networks: Challenges and Future Directions. *IEEE Access*. 2023;11:39295–39317. Available from: https://arxiv.org/abs/2304.00713.

24) Dwivedi AD, Srivastava G. Security analysis of lightweight IoT encryption algorithms: SIMON and SIMECK. *Internet of Things*. 2023;21. Available from: https://doi.org/10.1016/j.iot.2022.100677.

25) Zhou L, Kang M, Chen W. Lightweight Security Transmission in Wireless Sensor Networks through Information Hiding and Data Flipping. *Sensors*. 2022;22(3):823–823. Available from: https://doi.org/10.3390/s22030823.

26) Jangra M, Singh B. Mod-k-Chained Variant of PRESENT and CLEFIA Lightweight Block Cipher for an Improved Security in Internet of Things. *SN Computer Science*. 2022;3(1):71–71. Available from: https://doi.org/10.1007/s42979-021-00941-w.