# Enhancement of NTSA Secure Communication with One-Time Pad (OTP) in IoT

Ali Hasan Aidaros Alattas[1], Mahmood A. Al-Shareeda[1], Selvakumar Manickam[1,*] and Murtaja Ali Saare[2]
[1]National Advanced IPv6 Centre, Universiti Sains Malaysia, 11800, Penang, Malaysia
[2]Department of Computer Technology Engineering, Shatt Al-Arab University College, Basrah, Iraq
E-mail: alshareeda022@usm.my, selva@usm.my, murtaja.a.sari@sa-uc.edu.iq
*Corresponding author

**Keywords:** Internet of Things (IoT), NTSA, One-Time Pad (OTP), lightweight cryptographic algorithms

**Received:** October 24, 2022

*Internet of Things (IoT) systems use interconnected devices with limited processing, memory, storage, and power availability. Designing the IoT system requires careful consideration of data security. IoT networks are used to collect, process, and transport data; as a result, it needs to be encrypted and secured. To ensure that the data of IoT systems are protected, a variety of lightweight encryption techniques have been developed. These algorithms are unable to carry out complicated or extensive computations. The current challenge facing lightweight cryptographic algorithms, such as NTSA, is how to combine the highest level of security with the least amount of negative influence on runtime speed and space. By applying the One-Time Pad (OTP) technique, the proposed mechanism can raise the security level and effectiveness of NTSA. The proposed mechanism must be put into practise and put to the test in order to demonstrate its effectiveness and capacity to satisfy the needs of the resource-constrained devices. Due to the benefits of the OTP, this suggested method would be beneficial for devices with minimal resources. The proposed technique offers a greater security level, 2134, than NTSA, 2128, after examining and evaluating the experimental data noticed throughout the tests. NTSA is slower than the suggested approach by 70% in terms of runtime speed. While NTSA uses 16% of SRAM, the proposed algorithm only uses 12%. NTSA uses 70% more energy than the suggested algorithm, with higher energy consumption results of 0.000388 Joules for the proposed algorithm and 0.001295 Joules for NTSA.*

*Povzetek: Predstavljena je nova metoda za šifriranje in varnostna vprašanja IoT omrežij, ki dosega boljše rezultate kot NTSA.*

## 1 Introduction

The Internet is a system architecture that has allowed communications to advance to connect devices via different networks all over the world. Any individual object that connects to one of its networks can access the Internet for nearly any purpose that requires information (1; 2). It enables access to digital information through human or machine-to-machine (M2M) communications (3). Each connected object in the Internet of Things has a unique identity and can connect to other connected objects (4). Medical equipment, monitoring equipment, machinery, automobiles, and buildings will all be upgraded to become intelligent objects that can interact with people or other IoT devices (5; 6). The digital transformation of many industries is what fuels the IoT's growth. IoT connections will increase from fifteen billion in 2015 to seventy-five billion by 2025, as stated in (7), see Figure 1.

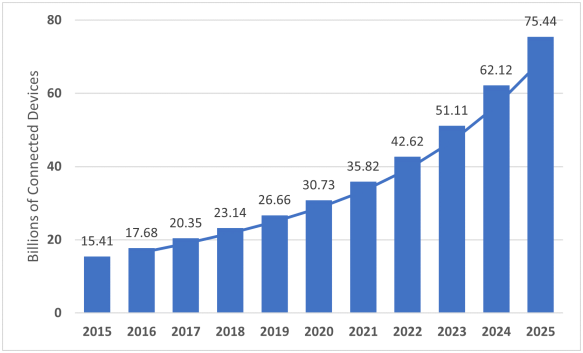


Figure 1: IoT connected devices in number.

The security issue is an afterthought because the resource-constrained networked device is meant to consume a little power to give all essential capabilities (8; 9). There are problems with IoT hardware, including the possibility of an attack on the device's encrypted data since some IoT devices are too small to support asymmetric cryptography algorithms. A gadget transmits or receives data that needs to be encrypted (10; 11). However, using cryptographic methods on devices with limited resources is difficult. The device itself, such as an 8-bit microcontroller with a 2KB RAM limit, performs the encryption operation (12; 13).

Traditional cryptography techniques cannot be implemented on such devices since they are expensive and inefficient. In order to address the security concerns on nodes with limited resources, lightweight ciphers have been developed. They are made to achieve cryptographic computational operation while adhering to the restrictions of microcontrollers, small-size RAM, and low power consumption (14; 15; 16).

By exploiting the benefits of the OTP technique, the suggested mechanism introduces a solution for both high security and higher performance. This paper focuses on symmetric encryption ciphers and the OTP approach as a foundation for lightweight cryptography. The advantages of block ciphers, which are simple to implement, the OTP technique, high security, and high performance can result in a dependable and robust system. The following are some of the research's contributions:

- This study will make it easier to deploy OTP in all areas of life and execute its secrecy into sensitive applications that require a high level of security with high performance because the OTP approach has shortcomings that have limited its adoption.
- One of OTP's flaws is the key exchange procedure. Therefore, this study will address this problem by creating a simple protocol for parties to exchange keys.

The rest of this paper is organized as follows. Section 2 reviews some related work. Section 3 introduces the background of this paper. Section 4 describes the general proposed mechanism's architecture. Section 5 and Section 6 provide a security analysis and results for the proposed mechanism, respectively. Lastly, Section 7 shows the conclusion and future work in this work.

# 2 Related work

Advance Encryption Standard (AES) was proven to be the best trusted and researched block cipher and still has to be subjected to more study to make it acceptable for resource-constrained devices, as indicated in (17; 18). While some lightweight cryptographic algorithms, like G-TBSA, are adequate for some factors like processing power and energy, they are not resistant to all types of attacks. Like G-TBSA, a number of lightweight cryptographic algorithms are adequate in some respects, such as computational power and energy, but are not resistant to other assaults.

None of the prominent modern lightweight block and stream ciphers is typical in offering the security, affordability, and performance for IoT devices with limited resources (19). It has been noted that the advancement of lightweight cryptography is still ongoing (20; 21).

However, developing an algorithm that satisfies the needs of lightweight cryptography for IoT devices is a considerable task. To accommodate various IoT device memory limits, the author has devised a simple encryption method that employs variable-sized keys and data blocks. This idea makes use of DNA sequences to produce random keys.

Many current LWC algorithms, according to (22), concentrate on lowering the cost of memory, computational power, physical area, and energy consumption and enhancing throughput and latency without paying attention to security vulnerabilities. In addition, the author claims that a successful encryption algorithm must strike a balance between three LWC design objectives (Security, Performance, Cost).

Banani et al. (23) employed the standard performance measures (memory occupied, execution time, and power consumption) to trade off among the various algorithms, including TEA. They did this by referring to the security and performance evaluation criteria. The avalanche effect attribute was utilised by the author to illustrate the security metrics in (24).

In order to summary the limitations of existing works, we list algorithms and attacks occurred as presented in Table 1. According to this leak, we enhance NTSA secure communication with OTP in IoT in order to raise the security level and effectiveness of NTSA. The proposed mechanism must be put into practise and put to the test in order to demonstrate its effectiveness and capacity to satisfy the needs of the resource-constrained devices (13).

# 3 Background

## 3.1 One-Time Pad technique

Similar to a stream cipher but not one that uses a random key generator is a One-Time Pad (OTP). It is a safe method for encrypting a message so that a cryptanalyst cannot decipher the message from the information (25). When encrypting and decrypting data, a random key must have a length equal to or greater than the message length produced by a genuine random generator. Then, it will be deleted so that a fresh new random key is used for the subsequent encryption and decryption procedures (26; 27).

OTP typically employs the XOR operation to encrypt plaintext by fusing the message and key bits, which is quick and appropriate for IoT devices. This increases security and makes OTP uncrackable under the following circumstances: (I) The key's unpredictability; (II) The length of the key must be at least as long as the plaintext; (IV) The key can be used just once; and (IV) The key has a very high level of confidentiality (28; 29; 30).

## 3.2 Lightweight Cryptographic Algorithm (LWC)

Designed for devices with limited resources, Lightweight Cryptographic Algorithm (LWC) is a branch of cryptography that seeks to offer solutions (31). The NIST started a lightweight cryptography project in 2013 to investigate how well the NIST-approved cryptographic standards function on restricted devices and to determine the demand for

Table 1: Different Attacks on Some Lightweight Cryptosystems in Related Work

| Algorithm | Attack | Cipher | Key Size | Structure |
|---|---|---|---|---|
| TEA | Related-key attack | Block | 128 bits | Feistel |
| XTEA | Related-key attack | Block | 128 bits | Feistel |
| HB-2 | Related-key attack | Hybrid | 128 bits | Hybrid |
| PRINTcipher | Related-key attack | Block | 80, 160 bits | SPN |
| PRESENT | Related-key attack | Block | 80, 128 bits | SPN |
| XXTEA | Chosen-Plaintext attack | Block | 128 bits | Feistel |
| AES | Biclique cryptanalysis | Block | 128, 192, 256 bits | SPN |
| LED | Biclique cryptanalysis | Block | 64, 80, 96, 128 bits | SPN |
| PRESENT | Biclique cryptanalysis | Block | 80, 128 bits | SPN |
| Grain | Key recovery attack | Stream | 80 bit | Stream |
| MICKEY | Differential fault attack | Stream | 80 bits | Stream |
| SIMON | Differential fault attack | Block | 64,72, 96,128, 144, 192, 256 bits | Feistel |
| SPECK | Differential fault attack | Block | 64,72, 96,128, 144, 192, 256 bits | ARX |
| PRESENT | Differential fault attack | Block | 80, 128 bits | SPN |
| PRESENT | Truncated differential attack | Block | 80, 128 bits | SPN |
| ChaCha | Truncated differential attack | Stream | 256 bits | ARX |

specific lightweight cryptography standards. The literature will go into detail about how lightweight encryption algorithms have been designed to meet the capabilities of resource-constrained devices to provide both a high level of security and high performance in terms of minimizing the runtime and space complexities as much as feasible (32).

### 3.3 TEA and NTSA algorithms

TEA uses 64 rounds spread over 32 cycles. Starting with dividing a 128-bit key into four 32-bit subkeys (k0, k1, k2, and k3), a 128-bit plaintext block is split into two blocks of 32 bits. Each set of four operations uses ADD, XOR, and left and right shift operations. In order to increase confusion during all rounds of encrypting a 64-bit plaintext block, NTSA, which is an upgrade to TEA, tries to generate dynamically changing subkeys derived from a 128-bit key (33; 34).

## 4 General proposed mechanism's architecture

The TEA algorithm performs well in LWC and is simple to implement in both hardware and software. It also uses less memory. However, it is susceptible to related-key assaults and has a flaw in the round function mixing. Based on the findings of the comparison analysis between TEA and NTSA, NTSA resolved the primary scheduling issue (35). It turns out that developing a system based on the NTSA and OTP will offer a reliable and lightweight cryptosystem for IoT devices with limited resources. As shown in Figure 2, the system uses block ciphers and OTP techniques along with two different forms of symmetric-key primitives (36).

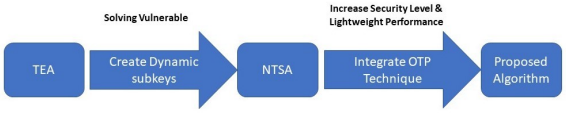


Figure 2: The proposed scheme's mechanism.

### 4.1 Random keys generation

Both in hardware and software, the TEA algorithm works well in LWC and is straightforward to implement. It consumes less memory as well. However, it has a vulnerability in the round function mixing and is vulnerable to related-key attacks. The fundamental scheduling issue was resolved by NTSA based on the results of the comparative analysis between TEA and NTSA (12).

To prevent noise bias between the axes, the Von Neumann extractor method extracts two bits from each axis. Then the desired value is generated by applying Equation (1) to a random byte.

$$RandByte = (x \leq 6) \oplus (z \leq 4) \oplus (y \leq 2) \oplus x \oplus (z \leq 2) \quad (1)$$

Additionally, XORing independent binary variables always minimizes bias, as the piling-up lemma in (37) shows. Let the random byte be made up of the values x, y, and z that are retrieved from the x, y, and z axes, respectively. The flowchart for the procedure that will produce a random byte is shown in Figure 3.

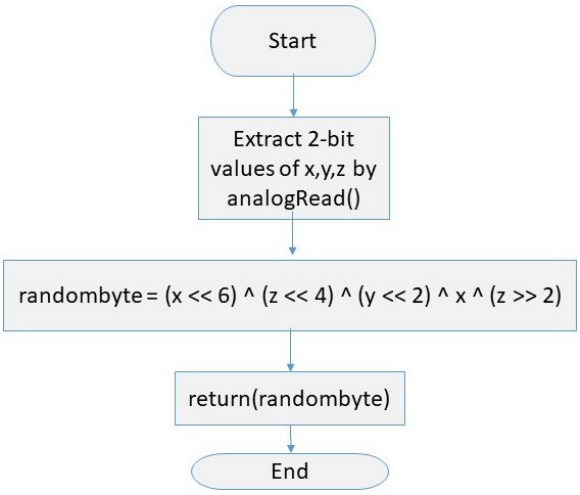


Figure 3: RNG flowchart.

## 4.2 Proposed mechanism

We examine three different instances of data transmission via the Internet of Things devices:

- Periodically, devices will send data; the transmission interval is determined by the application domain.
- The transmission interval is fixed, and devices will periodically deliver data.
- Data will be sent by devices when it is modified.

This approach is intended to be used in the third scenario, which involves passing information from the temperature sensor to an air conditioner. A key must be used once in the OTP approach before being discarded in order to generate a new key for use in the following encryption procedure. Therefore, sending data on a regular basis is not recommended, especially if the interval is small, like every second or even every hour (16).

### 4.2.1 Encryption algorithm

Regarding the first research issue, Figure 4 provides an illustration of the suggested algorithm. The Feistel structure, which uses the around function, is used by the proposed method since it employs the same round function that NTSA and TEA do. Data block P and subkey ki are two inputs that a round function accepts and returns one result.

The following conditions can be met by watching the encryption process's algorithm:

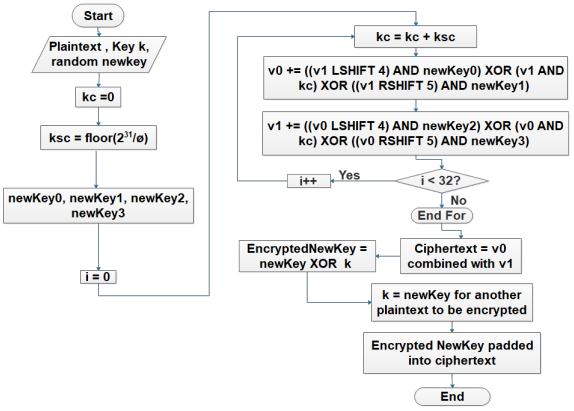


Figure 4: Encryption flowchart.

- single-use key.
- The key's confidentiality.
- True random secret keys for each encryption procedure and the high level of security provided by random keys contribute to increasing security.

### 4.2.2 Decryption algorithm

Figure 5 shows depicts the entire decryption process.

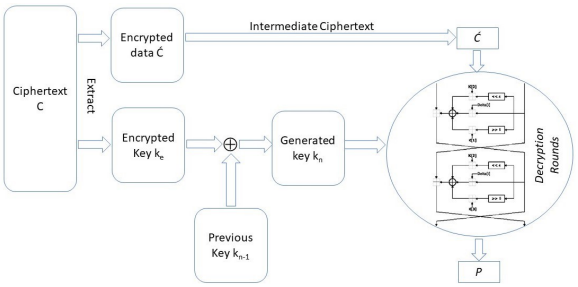


Figure 5: The entire decryption process.

## 4.3 Key padding protocol

When a new key is generated and utilised in each encryption procedure, the problem of key exchange between a sender and recipient arises. The sender and the recipient need to exchange this key. The problem must be taken into account to minimise the need for computationally intensive operations, as is the case with traditional cryptography like RSA. The complexity of key exchange problems grows as a result of the connectivity between IoT devices and machine-to-machine communication. The approach suggests padding the key for message encryption and decryption into the ciphertext after it has been encrypted using the previous key and the XOR operation, as shown in Figure 6.

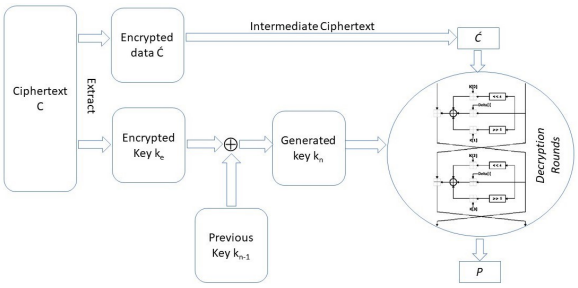


Figure 6: Process of encrypted key padding.

## 4.4 Key extraction protocol

Using (16), the intermediate ciphertext's bytes are first extracted from the final ciphertext C during the decryption process, D(cn, kn-1). Next, the bytes of the encrypted key ke is collected from the final ciphertext, as shown in Figure 7. Then kn is obtained by performing an XOR operation between kn-1 and ke in order to decrypt the message and the following newKey ke.

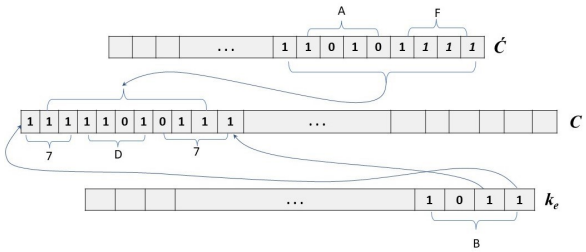


Figure 7: Final ciphertext bits.

## 4.5 Discussion

OTP is the suggested remedy, as discussed in the sections that came before it, to simplify the design complexity of lightweight encryption methods. If all of its requirements are met, it operates at a high-performance level with strong security. NTSA was chosen for this thesis because it fixed a flaw in the TEA algorithm, the most desirable lightweight encryption technique. However, NTSA continues to employ the same key, which is vulnerable to attack, across all encryption processes. IoT systems' limited-resource devices can use the proposed technique. The simplest and fastest computational processes, XOR, left and right shift, and modular addition arithmetic, which rely on bitwise operation, have been suggested as an effective mechanisms for implementing key exchange procedures.

# 5 Security analysis

Shift registers, Feistel structures, and substitution-permutation networks are a few examples of specific structures on which certain ciphers are based. The most frequent threats to Feistel-structure block ciphers, upon which this paper depends, will be covered.

### 5.0.1 Cipher-text only attack

In this type of attack, the attacker can capture ciphertext and attempt to decrypt it in order to learn more about the plaintext and, if possible, the key. To examine and decrypt ciphertexts, an attacker needs n of them. These attacks have not been successful against current ciphers.

### 5.0.2 Known-plaintext attack

Some ciphertext's plaintext can be deciphered by an attacker. The goal of this assault is to reveal and decrypt the remaining ciphertext blocks using already-known information, which may reveal the key.

### 5.0.3 Chosen-plaintext attack

Despite being the least factual, this form of attack is potent. With this technique, the key used to encrypt data is determined by measuring a change in the ciphertext.

### 5.0.4 Chosen-ciphertext attack

The chosen-Ciphertext attack also includes a chosen-plaintext assault, which decrypts ciphertexts with a particular key. If this type of attack is combined with a chosen-plaintext attack, it is not very practical.

### 5.0.5 Differential cryptanalysis

The typical technique for attacking cryptographic algorithms is this one. Since linear cryptanalysis uses a known-plaintext attack instead of the usual differential cryptanalysis method's chosen-plaintext attack, it is thought to be more practical in everyday life. Particularly, it examines ciphertext pairs. Pairs of ciphertexts with distinct plaintext differences and examine how these differences change as the plaintexts move through the encryption algorithm's rounds when they are encrypted with the same key. As long as the two plaintexts satisfy specific differences, they can be selected at random (with a fixed difference). Then, assign various probabilities to various keys based on the variations in the generated ciphertexts. One key will become more and more obvious as the most likely correct key as more and more ciphertexts are studied.

### 5.0.6 Related-Key attack

Comparable to differential cryptanalysis, but focused on key differences. Without knowing the actual keys, this approach focuses on the relationship between a pair of keys. It uses plaintext encryption using both the real key K and some derived keys, as well as a straightforward link between subkeys in neighboring rounds. The method for

changing the keys must be specified; it may involve flipping key bits while concealing the true key.

The TEA's issue, which is brought on by weak key scheduling and a weak mixing component of the round function, is resolved by the NTSA. The TEA technique can be broken by a related-key attack using 223 selected plaintexts, especially if the key is weak (38). The NTSA's defence against the related-key attack will then be clear see how it was created.

#### 5.0.7 Keys equivalence

If two keys, k1, and k2, produce the same ciphertext after encrypting the same plaintext, then they are equal. Ek1(P) = Ek2(P), where E is the encryption function, P is the plaintext, and k1 and k2 are separate keys K and K is the key space, illustrating this relationship. The connection between the classes that make up K is such that k1 and k2 are members of the same class. To make this argument more understandable, use mathematical equations and demonstrate the TEA's susceptibility as shown in (39).

#### 5.0.8 Resistance of NTSA against related-key attack

The NTSA uses the same round function as the TEA algorithm, with one modification to improve the key schedule procedure. In the NTSA, the extract() function is called after each round, and it dynamically returns a value from an array. Thirty-three separate 32-bit values that are obtained from the 128-bit key fill up this array

#### 5.0.9 The proposed mechanism security analysis against related-key attack

Great security level and high performance in terms of space and time complexity are coupled in the suggested method by incorporating the OTP technique. As long as its requirements are met, the OTP, a conventional but nonetheless powerful cipher, can withstand quantum computers (40). Despite using the same round function as the TEA, the suggested approach is more secure than NTSA.

# 6 Results

## 6.1 Execution time

This analysis will show the encryption and decryption process execution times for both algorithms, measured in milliseconds based on the number of cycles. The number of bits encrypted and decrypted using the 128-bit key serves as a measure of the data size, which is determined by each cycle's two rounds. Tables and bar charts are going to be used to show the results. With data blocks of 64, 128, 192, and 256 bits, encryption and decryption functions will be conducted throughout the number of cycles 8, 16, and 32 to reach the execution time tests. These several categories serve as illustrations of how the suggested mechanism and the performance of NTSA are affected by the number of rounds and size of the data block.

#### 6.1.1 Execution time of encryption process

The encryption function in the suggested technique requires three inputs: a 64-bit block of data, a previous key with a 128-bit size, and a 128-bit fresh key that is generated before each new encryption function begins. It has two parameters in NTSA: plaintext block with a 128-bit key and 64 bits. The execution time of NTSA increases by roughly 0.828 ms in Figure 8 and Table 2 for the same number of rounds, 16 rounds, and various block sizes. The suggested algorithm, however, is implemented more quickly than NTSA, and its runtime increases by about 0.544 ms for every increase in block size. In other words, the suggested algorithm outperforms the NTSA by 50%.

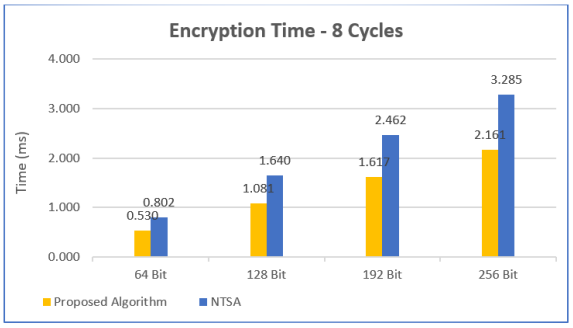


Figure 8: Encryption time for 8 cycles in ms.

Table 2: 8-cycle encryption process results in milliseconds

| Algorithm | 64 Bits | 128 Bits | 192 Bits | 256 Bits |
|---|---|---|---|---|
| NTSA | 0.802 | 1.640 | 2.462 | 3.285 |
| Proposed | 0.530 | 1.081 | 1.617 | 2.161 |

## 6.2 Execution time of decryption process

While the proposed algorithm's ciphertext contains both the encrypted data and the new key, the NTSA's decryption function requires 64-bit ciphertext and 128-bit key parameters. The execution time increment rate for both algorithms to complete the decryption function in 8 cycles, or 16 rounds, is shown in Figure 9 and Table 3. The proposed technique and the NTSA have slightly different runtimes for the encryption and decryption operations under a class of eight cycles.

## 6.3 Memory occupation

Memory occupation in Bytes: In this paper, memory usage is calculated using SRAM memory for execution time and

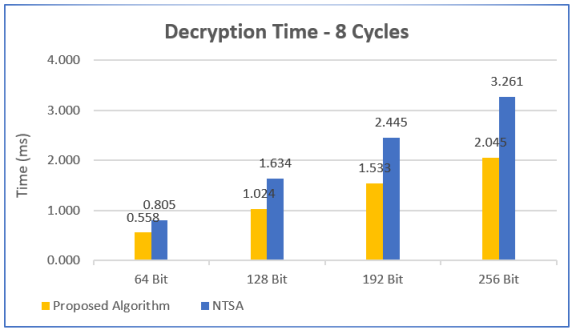


Figure 9: Decryption time for 8 cycles in ms.

Table 3: 8-cycle decryption process results in milliseconds

| Algorithm | 64 Bits | 128 Bits | 192 Bits | 256 Bits |
|---|---|---|---|---|
| NTSA | 0.805 | 1.634 | 2.445 | 3.261 |
| Proposed | 0.558 | 1.024 | 1.533 | 2.045 |

flash memory for storing code. To measure the amount of energy used by both algorithms, the Arduino Uno board is powered by a 9-volt battery in this experiment. The reading of the current passing through the Arduino Uno board was taken using the multimeter. There is a 5V voltage and a 20mA current (0.02A).

Figures 10 and 11 show that the NTSA uses 7% of flash memory to store the algorithm, which is 2546B, and 16%, or 340 bytes of 2KB, to store the global variables in SRAM for encrypting and decrypting 64-bit plaintext with a 128-bit key. Because the NTSA's code file has two routines—encryption and decryption—as well as one function to retrieve the array's contents, it uses less flash memory than the suggested approach. In contrast, the code file for the proposed approach contains the encryption and decryption procedures as well as the key generation function. In comparison, the NTSA employs an array to hold 33 32-bit subkeys during runtime, which requires more SRAM capacity. The suggested approach, in contrast, employs an array that holds six 32-bit values that constitute the final ciphertext.

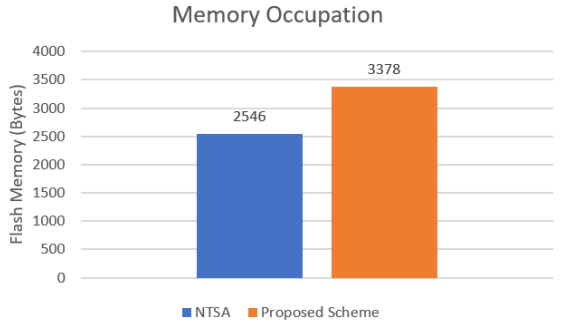


Figure 10: Memory occupations.

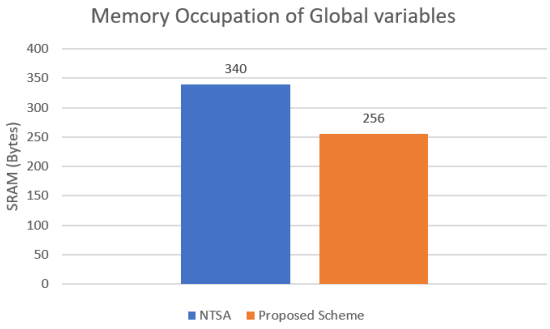


Figure 11: Global variable memory occupation.

## 6.4 Energy consumption

Energy consumption: a device with limited resources and low energy usage lasts longer on its battery. As shown in Figure 12, the following equipment should be available to conduct this experiment and assess the power consumption: an 8-bit microcontroller Arduino Uno board (MCU), the proposed algorithm, and its equivalent NTSA. (1) Multimeter to measure the voltage and the current; (2) Jumper wires; (3) Banana Plug to Crocodile Clip; (4) DC Barrel Jack Adapter – Female to screw terminals; and (5) Power Source whether 9V Battery with 9V Battery Connector to DC Jack Arduino or Wall Power Supply (5V- 2Am).

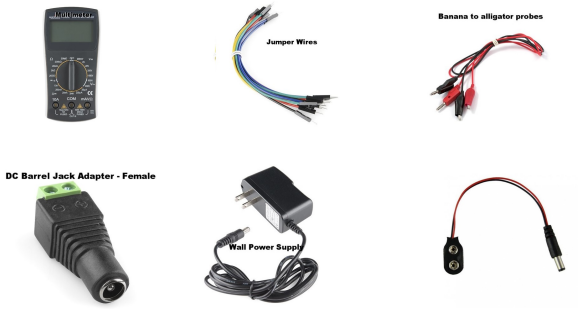


Figure 12: Tools used.

The procedures that follow explain how to set up the necessary equipment to begin this experiment's mechanism:

- Prepare the multimeter by inserting the red probe of the banana-crocodile cable into the mAV port to measure the voltage and the black probe of the cable into the COM port to measure the current.
- The multimeter's dial should be set to A for current and V for voltage.
- Connect the red probe from the multimeter to the (+) end of the power supply and the black probe to the Vin port on the Arduino Uno board using a DC Barrell Jack Adapter. Lastly, attach the (-) end of the power

supply to the GND port on the Arduino Uno board. This circuit has a series connection.

In this experiment, the Arduino Uno board is powered by a 9-volt battery to measure the energy required by both algorithms. Using the multimeter, the reading of the current flowing through the Arduino Uno board was captured. The voltage is 5V, and the current is equal to 20mA (0.02A). Table 4 shows the energy usage for the encryption procedure for various categories of data sizes with fewer than 64 rounds. The results in Table 4 demonstrate that the suggested method provides great optimization in terms of power usage compared to the NTSA.

Table 4: The energy consumption for encryption process

| Algorithm | 64 Bits | 128 Bits | 192 Bits | 256 Bits |
|---|---|---|---|---|
| NTSA | 0.0003192 | 0.00065 | 0.000972 | 0.001295 |
| Proposed | 0.000096 | 0.000193 | 0.00029 | 0.000388 |

# 7 Conclusion and future work

Traditional cryptographic algorithms are not suitable for IoT devices due to their inherent limitations in terms of processing power, memory, storage, and energy. However, the ongoing development of lightweight cryptography will continue to produce suitable lightweight cryptographic mechanisms to meet these requirements. Consequently, this research suggests a mechanism to incorporate the OTP technique into the NTSA in order to take advantage of the high-security level with the high performance offered by the OTP and easy implementation offered by block cipher and combine them into one mechanism to provide a lightweight cryptographic algorithm that can be implemented on IoT devices easily and effectively.

The first research goal was accomplished by integrating the OTP technique into NTSA in order to increase security. The data was encrypted using various new random keys generated by the MPU6050 sensor, and the final ciphertext was created by padding the bits in order to share the newly generated key. The experiments covered in Chapter 4 demonstrate that the suggested mechanism offers a greater security level and higher performance in terms of the complexity of speed, reduced memory utilization, and lower energy consumption. This is relevant to the second study objective. The encryption and decryption runtimes show that NTSA is 70% slower than the suggested technique. NTSA uses 16% of SRAM, compared to 12% for the suggested method. In terms of security, the proposed technique offers 2134 security complexity compared to 2128 security complexity offered by NTSA. The proposed algorithm uses 0.000388 Joules of energy, but NTSA uses 0.0013 Joules, meaning that NTSA uses 70% more energy than the proposed approach.

# References

[1] S. S. Oyewobi, K. Djouani, and A. M. Kurien, "Visible light communications for internet of things: Prospects and approaches, challenges, solutions and future directions," *Technologies*, vol. 10, no. 1, p. 28, 2022. [Online]. Available: https://doi.org/10.3390/technologies10010028

[2] M. A. Al-Shareeda, M. Anbar, I. H. Hasbullah, and S. Manickam, "Survey of authentication and privacy schemes in vehicular ad hoc networks," *IEEE Sensors Journal*, vol. 21, no. 2, pp. 2422–2433, 2020. [Online]. Available: https://doi.org/10.1109/JSEN.2020.3021731

[3] I. H. Sarker, A. I. Khan, Y. B. Abushark, and F. Alsolami, "Internet of things (iot) security intelligence: a comprehensive overview, machine learning solutions and research directions," *Mobile Networks and Applications*, pp. 1–17, 2022. [Online]. Available: https://doi.org/10.1007/s11036-022-01937-3

[4] A. E. Omolara, A. Alabdulatif, O. I. Abiodun, M. Alawida, A. Alabdulatif, H. Arshad *et al.*, "The internet of things security: A survey encompassing unexplored areas and new insights," *Computers & Security*, vol. 112, p. 102494, 2022. [Online]. Available: https://doi.org/10.1016/j.cose.2021.102494

[5] I. Ashraf, Y. Park, S. Hur, S. W. Kim, R. Alroobaea, Y. B. Zikria, and S. Nosheen, "A survey on cyber security threats in iot-enabled maritime industry," *IEEE Transactions on Intelligent Transportation Systems*, 2022. [Online]. Available: https://doi.org/10.1109/TITS.2022.3164678

[6] M. A. Al-Shareeda, M. Anbar, S. Manickam, and A. A. Yassin, "Vppcs: Vanet-based privacy-preserving communication scheme," *IEEE Access*, vol. 8, pp. 150 914–150 928, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.3017018

[7] T. Alam, "A reliable communication framework and its use in internet of things (iot)," *CSEIT1835111| Received*, vol. 10, pp. 450–456, 2018. [Online]. Available: https://doi.org/10.36227/TECHRXIV.12657158.V1

[8] M. A. Al-shareeda, M. Anbar, S. Manickam, I. H. Hasbullah, N. Abdullah, M. M. Hamdi, and A. S. Al-Hiti, "Ne-cppa: A new and efficient conditional privacy-preserving authentication scheme for vehicular ad hoc networks (vanets)," *Appl. Math*, vol. 14, no. 6, pp. 1–10, 2020. [Online]. Available: https://doi.org/10.3390/s21248206

[9] M. Salimitari, M. Chatterjee, and Y. P. Fallah, "A survey on consensus methods in blockchain

for resource-constrained iot networks," *Internet of Things*, vol. 11, p. 100212, 2020. [Online]. Available: https://doi.org/10.36227/techrxiv.12152142

[10] S. Misra, A. Mukherjee, A. Roy, N. Saurabh, Y. Rahulamathavan, and M. Rajarajan, "Blockchain at the edge: Performance of resource-constrained iot networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 1, pp. 174–183, 2020. [Online]. Available: https://doi.org/10.1109/TPDS.2020.3013892

[11] V. Tambe, G. Bansod, S. Khurana, and S. Khandekar, "Reliability and availability of iot devices in resource constrained environments," *International Journal of Quality & Reliability Management*, 2022. [Online]. Available: https://doi.org/10.1108/IJQRM-09-2021-0334

[12] M. A. Al-Shareeda, M. Anbar, S. Manickam, and I. H. Hasbullah, "Password-guessing attack-aware authentication scheme based on chinese remainder theorem for 5g-enabled vehicular networks," *Applied Sciences*, vol. 12, no. 3, p. 1383, 2022. [Online]. Available: https://doi.org/10.3390/app12031383

[13] M. A. Al-shareeda, M. A. Alazzawi, M. Anbar, S. Manickam, and A. K. Al-Ani, "A comprehensive survey on vehicular ad hoc networks (vanets)," in *2021 International Conference on Advanced Computer Applications (ACA)*. IEEE, 2021, pp. 156–160. [Online]. Available: http://doi.org/10.1109/ACA52198.2021.9626779

[14] P. Panahi, C. Bayılmış, U. Çavuşoğlu, and S. Kaçar, "Performance evaluation of lightweight encryption algorithms for iot-based applications," *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 4015–4037, 2021. [Online]. Available: https://doi.org/10.1007/s13369-021-05358-4

[15] M. A. Al-Shareeda, M. Anbar, I. H. Hasbullah, S. Manickam, and S. M. Hanshi, "Efficient conditional privacy preservation with mutual authentication in vehicular ad hoc networks," *IEEE Access*, vol. 8, pp. 144 957–144 968, 2020. [Online]. Available: https://doi.org/10.3390/su14169961

[16] M. A. Al-Shareeda, S. Manickam, M. A. Saare, and N. C. Arjuman, "Proposed security mechanism for preventing fake router advertisement attack in ipv6 link-local network," *Indones. J. Electr. Eng. Comput. Sci*, vol. 2023, no. 29, pp. 518–526, 2023. [Online]. Available: https://doi.org/10.11591/ijeecs.v29.i1.pp518-526

[17] I. K. Dutta, B. Ghosh, and M. Bayoumi, "Lightweight cryptography for internet of insecure things: A survey," in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2019, pp. 0475–0481. [Online]. Available: https://doi.org/10.1109/CCWC.2019.8666557

[18] M. A. Al-Shareeda, S. Manickam, B. A. Mohammed, Z. G. Al-Mekhlafi, A. Qtaish, A. J. Alzahrani, G. Alshammari, A. A. Sallam, and K. Almekhlafi, "Provably secure with efficient data sharing scheme for fifth-generation (5g)-enabled vehicular networks without road-side unit (rsu)," *Sustainability*, vol. 14, no. 16, p. 9961, 2022. [Online]. Available: https://doi.org/10.3390/su14169961

[19] M. Rana, Q. Mamun, and R. Islam, "Current lightweight cryptography in iot security: A survey," in *Extended Abstracts*. Charles Sturt University, 2020, p. 27. [Online]. Available: https://researchoutput.csu.edu.au/ws/portalfiles/portal/100690557/SCM_HDR_Booklet_2020.pdf#page=27

[20] M. A. F. Al-Husainy, B. Al-Shargabi, and S. Aljawarneh, "Lightweight cryptography system for iot devices using dna," *Computers and Electrical Engineering*, vol. 95, p. 107418, 2021. [Online]. Available: https://doi.org/10.1016/j.compeleceng.2021.107418

[21] M. A. Al-Shareeda and S. Manickam, "Man-in-the-middle attacks in mobile ad hoc networks (manets): Analysis and evaluation," *Symmetry*, vol. 14, no. 8, p. 1543, 2022. [Online]. Available: https://doi.org/10.3390/sym14081543

[22] V. A. Thakor, M. A. Razzaque, and M. R. Khandaker, "Lightweight cryptography algorithms for resource-constrained iot devices: A review, comparison and research opportunities," *IEEE Access*, vol. 9, pp. 28 177–28 193, 2021. [Online]. Available: https://doi.org/10.1109/ACCESS.2021.3052867

[23] S. Banani, S. Thiemjarus, K. Wongthavarawat, and N. Ounanong, "A dynamic light-weight symmetric encryption algorithm for secure data transmission via ble beacons," *Journal of Sensor and Actuator Networks*, vol. 11, no. 1, p. 2, 2021. [Online]. Available: https://doi.org/10.3390/jsan11010002

[24] W. Diaztary, D. Atmajaya, F. Umar, S. M. Abdullah *et al.*, "Tiny encryption algorithm on discrete cosine transform watermarking," in *2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT)*. IEEE, 2021, pp. 415–420. [Online]. Available: https://doi.org/10.1109/EIConCIT50028.2021.9431930

[25] F. Ramadhani, U. Ramadhani, and L. Basit, "Combination of hybrid cryptography in one time pad (otp) algorithm and keyed-hash message authentication code (hmac) in securing the whatsapp communication application," *Journal of Computer Science, Information Technology and Telecommunication Engineering*, vol. 1, no. 1, pp. 31–36, 2020. [Online]. Available: https://doi.org/10.30596/jcositte.v1i1.4359

[26] A. Sarkar, S. R. Chatterjee, and M. Chakraborty, "Role of cryptography in network security," in *The" Essence" of Network Security: An End-to-End Panorama*. Springer, 2021, pp. 103–143. [Online]. Available: https://doi.org/10.1007/978-981-15-9317-8_5

[27] V. B. Savant and R. D. Kasar, "A review on network security and cryptography," *Research Journal of Engineering and Technology*, vol. 12, no. 4, pp. 110–114, 2021. [Online]. Available: https://doi.org/10.12691/iteces-3-1-1

[28] S. Bourougaa-Tria, F. Mokhati, H. Tria, and O. Bouziane, "Spubbin: Smart public bin based on deep learning waste classification an iot system for smart environment in algeria," *Informatica*, vol. 46, no. 8, 2022. [Online]. Available: https://doi.org/10.31449/inf.v46i8.4331

[29] M. A. Al-Shareeda, S. Manickam, B. A. Mohammed, Z. G. Al-Mekhlafi, A. Qtaish, A. J. Alzahrani, G. Alshammari, A. A. Sallam, and K. Almekhlafi, "Cm-cppa: Chaotic map-based conditional privacy-preserving authentication scheme in 5g-enabled vehicular networks," *Sensors*, vol. 22, no. 13, p. 5026, 2022. [Online]. Available: https://doi.org/10.3390/s22135026

[30] H. Kaur and A. Kaur, "An empirical study of aging related bug prediction using cross project in cloud oriented software," *Informatica*, vol. 46, no. 8, 2022. [Online]. Available: https://doi.org/10.31449/inf.v46i8.4197

[31] K. McKay, L. Bassham, M. Sönmez Turan, and N. Mouha, "Report on lightweight cryptography," National Institute of Standards and Technology, Tech. Rep., 2016. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/ir/2017/NIST.IR.8114.pdf

[32] M. A. Al-Shareeda, S. Manickam, B. A. Mohammed, Z. G. Al-Mekhlafi, A. Qtaish, A. J. Alzahrani, G. Alshammari, A. A. Sallam, and K. Almekhlafi, "Chebyshev polynomial-based scheme for resisting side-channel attacks in 5g-enabled vehicular networks," *Applied Sciences*, vol. 12, no. 12, p. 5939, 2022. [Online]. Available: https://doi.org/10.3390/app12125939

[33] S. Rajesh, V. Paul, V. G. Menon, and M. R. Khosravi, "A secure and efficient lightweight symmetric encryption scheme for transfer of text files between embedded iot devices," *Symmetry*, vol. 11, no. 2, p. 293, 2019. [Online]. Available: https://doi.org/10.3390/sym11020293

[34] M. A. Al-Shareeda, S. Manickam, B. A. Mohammed, Z. G. Al-Mekhlafi, A. Qtaish, A. J. Alzahrani, G. Alshammari, A. A. Sallam, and K. Almekhlafi, "Cm-cppa: Chaotic map-based conditional privacy-preserving authentication scheme in 5g-enabled vehicular networks," *Sensors*, vol. 22, no. 13, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/13/5026

[35] H. Ran, "Methodology for interval-valued intuitionistic fuzzy multiple attribute decision making and applications to performance evaluation of sustainable microfinance groups lending," *Informatica*, vol. 46, no. 8, 2022. [Online]. Available: https://doi.org/10.31449/inf.v46i8.4355

[36] S. Nie, "Evaluation of innovative design of clothing image elements using image processing," *Informatica*, vol. 46, no. 8, 2022. [Online]. Available: https://doi.org/10.31449/inf.v46i8.4250

[37] M. Matsui, "Linear cryptanalysis method for des cipher," in *Workshop on the Theory and Application of of Cryptographic Techniques*. Springer, 1993, pp. 386–397. [Online]. Available: https://doi.org/10.1007/3-540-48285-7_33

[38] M. Shoeb and V. K. Gupta, "A crypt analysis of the tiny encryption algorithm in key generation," *International Journal of communication and computer Technologies*, vol. 1, no. 1, pp. 9–9, 2019. [Online]. Available: https://www.bibliomed.org/?mno=302643835

[39] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, and L. Wingers, "Simon and speck: Block ciphers for the internet of things," *Cryptology ePrint Archive*, 2015. [Online]. Available: https://csrc.nist.gov/csrc/media/events/lightweight-cryptography-workshop-2015/documents/papers/session1-shors-paper.pdf

[40] B. Singh, G. Athithan, and R. Pillai, "On extensions of the one-time-pad," *Cryptology ePrint Archive*, 2021. [Online]. Available: https://eprint.iacr.org/2021/298.pdf