International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Rubik's Cube Encryption Algorithm-Based Technique for Information Hiding During Data Transmission in Sensor-Based Networks

Mahendra Balkrishna Salunke, Parikshit Narendra Mahalle, Gitanjali Rahul Shinde

Submitted: 06/06/2022 **Accepted:** 10/09/2022

Abstract-The rise of the Internet of Things has created new opportunities for individuals and businesses in various industries such as healthcare, transportation, and manufacturing. However, it is important that the security of the data collected by these devices is taken seriously. To ensure that the data is secure, various types of cryptographic primitives have been used in various security solutions. These primitives are generally not designed to preserve the privacy or support advanced functionalities. Here we used Rubik's cube encryption principle for secure data transmission. The original image is then scrambled using the algorithm of the famous cube, which is known as the XOR operator. Two secret keys are then used to encrypt the rows and columns of the image. The proposed system can be used to achieve strong encryption and flawless concealment, but can also withstand exhaustive, statistical, and differential attacks. Information security is currently the most significant concern. Whether the data is image or word, it must be protected. This study proposes a method for the secure transmission of data. The paper proposes a method that involves scrambling a three-dimensional color image matrix using a combination of operations.

Keywords- Encryption, Decryption, IoT, Rubik's cube, AES

[1]INTRODUCTION

Due to the rapid emergence and evolution of new technologies, such as Internet and artificial intelligence, information security has become a major concern for society. One of the most common sources of information that people send and receive is images. Due to the nature of images, they contain various sensitive information, such as trade secrets and personal privacy. An encryption technology is therefore needed to prevent these types of images from being stolen or leaked. The rise of the Internet of Things (IoT) has changed the way people think about the Internet. It allows devices to connect and interact with each other, which has made it more practical to use than the traditional Internet. However, it is still very vulnerable to security attacks due to its ability to communicate over a network. The various features of IoT include self-configuration, environment sensing, smart decision making, ad hoc networking, and autonomous reacting. Due to the increasing number of devices, it is estimated that the number of IoT gadgets will reach around 50 billion by 2020. When it comes to developing Internet of Things (IoT) devices, manufacturers need to consider their affordability. They should also consider having a secure model that can protect their data. This can be done through the use of codes, which are designed to only be used by those who are authorized to access the information.

The Internet of Things (IoT) is rapidly turning into an indispensable component of a wide variety of applications across a variety of domains, including consumer, industrial, and other fields. The Internet of Things makes possible in large part the realisation of concepts such as smart business, smart transportation, and smart planet. In its most fundamental form, the Internet of Things (IoT) is anchored by a collection of interconnected devices, including such sensors and actuators, that work together to offer a necessary service. Users' safety and privacy protection is one of the most critical requirements that the vast majority of Internet of Things apps are required to fulfil. The concepts of secrecy, integrity, and privacy are often attained through the application of cryptographic encryption strategies. Security is an umbrella phrase that incorporates all of these concepts and more.

Private Key management technique in Wireless and IoT Networks consists of a single key and data that is encoded as well as decrypted with the assistance of this key using Private Key Encryption/Decryption techniques (Procedures) such as Data Encryption Standards (DES), Advanced Encryption Standards (AES), etc. Private Key management technique in Wireless and IoT Networks

[1]Research Scholar, Department of Computer Engineering, Smt. Kashibai Navale College of Engineering, SPPU, Pune, India.
[2].Professor, Department of Artificial Intelligence and Data Science, Vishwakarma Institute of Information Technology, SPPU, Pune, India.
[3].Associate Professor, Department of Computer Engineering, Vishwakarma Institute of Information Technology, SPPU, Pune, India.

1. msalunke@gmail.com
2. aalborg.pnm@gmail.com
3. gr83gita@gmail.com

consists of a single key and data that is encrypted as well as decrypted with the help

It is necessary for the key to be stored within in the interacting nodes in order for information to be sent between them. This approach is straightforward to build, administer, and keep up to date because it just requires a single key. Encryption techniques are also significantly quicker than other methods for the maintenance of private keys. However, it is important to note that this method has a number of flaws, the most significant of which is the fact that an adversary may readily obtain the cryptographic keys through the process of node capture. During the time that the concerned nodes are communicating with one another, the session that they share needs to be extremely safe.

It is guaranteed that such legal user can access a consistent and trusted experience, which results in relatively insignificant legitimate security threats, in this paper the automated system interactively produces a new key at both ends, i.e. at the transmitter and receiver, and this ensures that there are negligible legitimate security threats. Along with incorporating the idea of a hop counter into the design of the architecture, one of our goals was to reduce the amount of key exchange over the network to an absolute minimum. The Key Generation Algorithm, which is essentially a Dynamic Key Generation Algorithm and whose design is of the cubic kind, is what we will be utilising in this particular piece of work. Also, the Key Distribution Algorithm presented in this study is comparable to the process of key distribution in a trusted base system; nevertheless, there are significant distinctions between the two.

The cube is scrambled in order to produce a key that resembles the shuffling that is done in a Rubik's cube. This is done in order to solve the puzzle. The purpose of encrypting the data is nonetheless the most important possible point of divergence in this work. In Rubik's cube, the cube is scrambled in order to solve it and to bring it in the desired form. However, in this paper, the Key Generation Algorithm scrambles the cube in order to generate private keys and not in order to solve it for any purpose. This is in contrast to how the cube is scrambled in Rubik's cube. As a result, the idea of scrambling a cube is derived from the notion of the Rubik's cube. Wireless networks may be used for many different things, including transportation, education, stock trading, the military, and the delivery of packages, disaster recovery, medical emergency treatment, and many other things

Due to the increasing number of devices and the complexity of the data they collect, the traditional methods of image encryption are gradually being replaced by newer technologies. Some of these include the data encryption standard DES, the advanced encryption standard AES, and the security standard for computers, the RSA. Here this paper describes the implementation of Rubik's cube encryption model in data transmission. Various studies have been conducted on the security of image encryption. It has been found that many schemes that use permutations are vulnerable to attacks that are mainly caused by the high information redundancy. This is understandable since recovering the secret permutations can be done by comparing the different types of encryptions. Although chaos-based image encryption is commonly used, it requires a high computational cost. Unlike other types of encryptions, it is not always possible to achieve a one-dimensional chaotic cryptosystem due to its weak security and small key spaces. The concept of the concept of the cube is used to scramble the original image's pixels. This method only affects the position of the pixels. In order to perform this, two random keys are used to apply the bitwise XOR to the odd rows and columns.

RELATED WORK

The following section discusses the various existing technologies that utilize the AES algorithm for encryption. In [1] the authors have pressed on the importance of safeguarding the information that is in transit in a Smart City environment where various Information and Communication technologies have been used for the betterment of human lives. Protecting the data from known ill-effects such as eavesdropping, tampering and many more needs to be taken care with effective encryption standards and the one used here is the S-box Advanced Encryption Standard (AES) using the combinational logic.

The authors[2] have performed a logical rearrangement in the S-box (byte substitution) module which reduces the number of gates in the critical. This addresses the most important aspects that need to be fulfilled by an encryption algorithm such as flexibility and low cost (for implementation) along with a resistance to brute-force and cryptanalytic attacks. In the authors[3] have adopted the AES algorithm in order to address the very specific need for a hardware accelerator in cloud servers that are extensively utilized for data storage and computation. The hardware accelerator thereby reduces the additional overhead on the cloud server caused due to handling encryption of information. The AES algorithm has been chosen for the various characteristics such as security, flexibility and ease of use.

In the authors[4] have presented a high throughput efficient hardware implementation of the Advanced Encryption Standard (AES) algorithm for high speed network applications. The authors have also introduced a 5-stage pipeline S-box design utilizing combinational logic to improve the speed and achieve maximum operational frequency. In [5] the authors have presented a lightweight masked implementation of the AES algorithm to provide

security in an IoT environment against the CPA (Correlation Power Analysis) attacks. The implementation helps in protecting the stored secret key from leaking via the substitution box. The intermediate data (highly related to the false key) is hidden during the reconstruction stage using the Wave Dynamic Differential Logic (WDDL) based XOR gates. This technique reduces the overhead in terms of power, area and performance to a greater extent.

In the authors[6] have presented a low-cost AES-128 Implementation for the resource constrained edge devices in IoT applications. The edge devices rely upon cryptographic algorithms in order to encrypt the data that is collected from the surrounding sensors. This has led the authors to present a novel mechanism using the AES algorithm with time-multiplexed architectures. A maximum operational frequency of 1.053GHz is obtained in the proposed scheme by following a four-fold mechanism on the AES hardware by including the resource sharing mechanism with a modified S-box. The authors [7]have introduced a Low-Power AES Data Encryption Architecture for a LoRaWAN. The LoRaWAN is the most suitable network application for an IoT environment due to its various advantages such as low power communication, star-of-stars topology, well defined MAC layer protocol and three communication modes. However, the AES encryption mechanism still consumes more power in the IoT end devices powered by batteries, due to the power involved in several cycles of repetition. To address this, the Low-Power AES Data Encryption Architecture (LPADA), an AES encryption architecture with low power consumption for data encryption has been introduced. A low powered S-box, power gating technique and power management technique has been utilized to achieve the goal.

The authors[8] have presented an area-efficient Nano-AES Implementation for the resource constrained Internet of Things devices. The 256-bit key AES algorithm offers enough security for the various levels of IoT applications and protocols. However, the software implementation of the AES algorithm has some disadvantages such as higher latency involved in data processing and transportation and increased power consumption. Therefore, many applications have adopted the hardware implementation due to its suitability for the IoT applications and the associated resource constrained devices. However, the AES serial implantation architecture with 8-, 16-, and 32- bit datapaths provide low throughput which has led the authors towards the introduction of the 8- bit datapath architecture for the resources constrained devices which occupies lesser internal wires. An attempt has also been made to reduce the number of blocks, utilizing low area design and merging of functions.

The following set of papers gives a brief overview of the various approaches and works done in line with the key management techniques. The authors[9] have introduced the Rubik's cube-based algorithm that is used for key management in wireless network. The algorithm is built upon the private key management technique in-order to ensure a safe and a dynamic key generation policy. A major advantage of the Rubik's cube mechanism is that the keys are not exchanged between the participating nodes thereby protecting the keys from being captured and interpreted in transit. The algorithm also has a mechanism called the hop count check which makes provides an alert when a node is being compromised, in turn making it difficult to acquire the key.

The authors'[10] have mainly focused on reducing the delay in communication between multiple-input-multiple-output IoT devices, which is caused to the key establishment process. A notable aspect of the proposed scheme is that it does not pause the ongoing communication between the participating devices during the process of key establishment. The overhead on the IoT networks is greatly reduced as the system requires only one device to send pilot signals to the other device. The entire system and the keys are protected from eavesdroppers and other IoT users as the system does not impose the key quantization and reconciliation phases in the public channel.

The authors[11] have introduced a system that ensures a secured exchange of information by encrypting an image using the Advanced Encryption Standard (AES) and a Rubik's cube based novel algorithm. Raspberry Pi is used as server and a storage device in which the encrypted image is stored. A user is then able to access the image and decrypt it with the appropriate secret key shared by the sender of the encrypted image.

The authors[12] have proposed a scheme to ease the key update scheme in the SM9 identity-based cryptography system. The authors have introduced a decentralized scheme for identity authentication and key management using the block chain technology. Here, the system requires the users to use the Identity Generator Center (IGC) only once for the generation of the key and identifier, after which the key can be automatically updated by the user. The identity authentication process becomes much easier in this scheme as the identifier stays the same as the public key. The only things that keep updating are the private key and parameters. Every key update is treated as a transaction and is logged in the chain blocks. This ensures that the key updates are not tampered with, which enhances the integrity and reliability of the process.

The authors[13] have focused on addressing the security and privacy risks in a wireless communication system. The system proposes to generate symmetric keys between the participating entities, by practically tracking

the users/human movement in the house or by recording the movements of the antenna. The system aims at enhancing the quality of communication (reduce the effect of noise) by incorporating the adaptive optimization method that helps in increasing the bit generation and bit matching rates. The randomness of quantization is improved by with the proposed D-gray coding mechanism.

The authors[14] have focused on addressing the security threats that exists in online medical services (medical IoT). Firstly, the authors have proposed an algorithm for key management called the Chaotic Whale Optimization (CWO) algorithm. The algorithm helps to reduce the number of iterations thereby reducing the costs incurred in hardware. The authors also have proposed a flexible design of modified ULBC algorithm which helps in realizing a power, area, delay efficient and attack free system.

DATA TRANSMISSION IN INTERNET OF THINGS

The various protocols used by the Internet of Things (IoT) devices allow them to communicate with each other and with other systems. They can do so in different ways depending on their location, their requirements, and their environment. There is no optimal protocol for connecting all of these devices. The optimal protocol for each application depends on its requirements and the overall architecture of the system as shown in fig.1. Although there are many different types of systems that can be used with the IoT, the majority of them come with a set of communication components. Objects that can be connected to the Internet of Things include sensors. Examples of these include the smallest temperature sensor and the largest industrial robot[15].

The various types of communication components that can be used by the IoT devices include local communications and gateway. Local communications allows the gadget to communicate with the other devices in its vicinity. Gateways are used to route the data to the Internet. A network server is a type of device that is usually housed inside a data center. It is responsible for handling the transmission and reception of data that the IoT devices generate. Developers can create cloud-based applications that allow people to access and use the data collected by the IoT devices. These applications can then be used to make changes to the information that the devices provide[16].

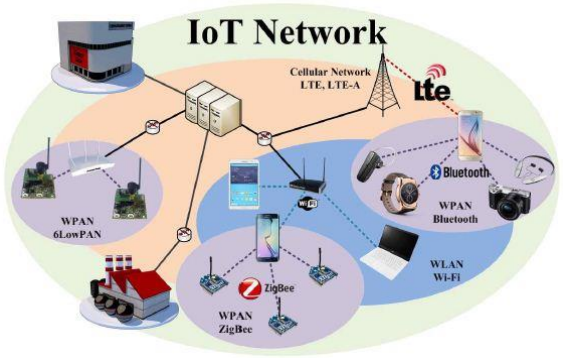

Figure. 1 Data transmission in IOT network (src- opentechdiary)

PUBLIC AND PRIVATE KEY CRYPTOSYSTEM

In information security, encryption algorithms widely available are categorized into private (symmetric) and public (asymmetric) keys encryption algorithms. They are used in the process of encryption and decryption of sensitive information.

Private Key cryptosystem refers to the process where the same key is used for both encryption and decryption of the information[17]. It is also sometimes referred to as symmetric key encryption as the key is common to both the participating parties (sender and receiver of the encrypted message). A few well known symmetric key algorithms are Digital Encryption Algorithm (DES), Triple DES, Blowfish etc. A few things to be considered while using the private key cryptosystem are as follows:

The secret key that is used for encryption and decryption must be exchanged between the participating parties before the actual exchange of data (communication) takes place.

As the secret key is common and exchanged over a common channel between the participating entities, the key is highly prone to attacks in transit. This requires the necessity of having a robust mechanism in place to facilitate an efficient and secure key exchange process. Also, the keys need to be changed regularly in order to prevent any attacks.

The number of keys required for a communication between two parties in a group of n people is n(n-1)/2.

Since the length of the keys used in the private key cryptosystem is smaller, the encryption and decryption process become faster and the processing power that is required for the same is lesser.

Public key cryptosystem is a process where two different keys are used for encryption and decryption of the data. A few of the public key cryptographic

algorithms are referred to as asymmetric key cryptographic algorithms. Every participant has a pair of keys, namely, the public key and the private key which are dissimilar but are mathematically related, thereby making it possible to decrypt the cipher text with a key that is different from the key that was used to encrypt the data[18]. A few notable features of the public key cryptosystem are as follows.

As the name indicates, the public keys reside in a public repository common to all the participating entities and the secret key is well guarded and kept as a secret with the respective participants.

The strength of the public key cryptosystem lies in the fact that although the public and private keys are mathematically related it is impossible to compute and find one key from another known key.

Suppose there are two entities (s1(sender1) and r1(reciever1)) who are participating in communication and s1 wants to send data to r1, then, s1 obtains the public key of r1 from the public repository. S1 then encrypts the data with r1's public key and upon receiving the encrypted message r1 uses its own secret private key to decrypt the data.

As opposed to the private key cryptosystem, the length of the keys used in the public key cryptosystem is larger which makes the encryption and decryption process slower and the processing power that is required for the same is higher.

Main challenges of private key cryptosystems.

Since the sharing of the private key between the participant entities is a mandatory phase, without a robust key exchange mechanism in place the system is prone to attacks.

The private key needs to be changed often in order to ensure security which creates an additional overhead in the system, where time and effort is spent more on key exchange instead of on the actual data.

A secure key establishment mechanism is required to facilitate the process where both the sender and receiver agree upon a shared secret key before the start of any communication.

There needs to be a robust mechanism for trust establishment between the sender and the receiver. This 'trust establishment' is very crucial in an IoT scenario where different users and devices communicate with many other unknown devices at various instances. For example, when a sender has lost a key to an attacker and the receiver has not been informed about it, then the system has been compromised without the knowledge of the receiver who is also a vital part of the system.

Although the private key cryptosystem is faster and requires much lesser processing power than that of the public key cryptosystem, in an IoT scenarios where resource constrained devices are in use largely, the private key cryptosystem may also be hard to implement. Another challenge is where the power required to keep the devices up and running comes into picture. Since the system will be performing additional tasks of exchanging the keys at regular intervals and securing the channel for transmission of keys, there Is a possibility where the algorithm might be using up most of the power in the device. Frequent identification of battery depletion and change of batteries of such devices can be a challenge if they are deployed in remote and unreachable areas.

AES ALGORITHM

The AES (Advanced Encryption Standard) aka Rijndael algorithm is a symmetric block cipher known as the AES. It takes 128 bits of plain text and converts them into 256, 128, and 192 bits of ciphertext using the keys of 128, 192, and 256. Since it is secure, it is considered to be the world standard. The AES uses a substitution-permutation network, which produces multiple rounds of ciphertext depending on the size of the key. For instance, a 128-bit key size dictates the number of rounds, while a 192-bit key size is used for 12 and a 256-bit key for 14. Since only one key is used in the algorithm, the key needed to expand to get the keys for each round is included in the calculation.

The Advanced Encryption Standard (AES) was established by the NIST in 2001. It is widely used for the secure transmission of electronic data. Compared to other methods, such as triple DES and DES, it is more powerful. The key size of an AES block cipher is 128/192.256 bits. It takes 128 bits of plain text and forwards 128 bits of encrypted data as output. It utilizes a series of linked operations to perform a substitution-permutation network. This method involves replacing and shifting the input data.

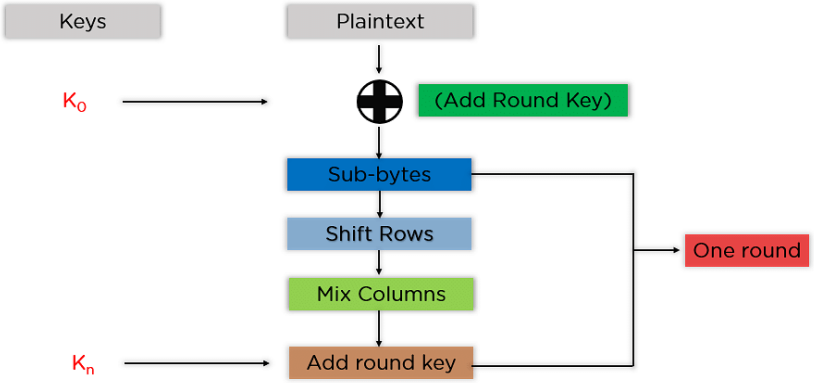

Figure. **2** AES steps

Each block should be followed step by step as shown in fig. . After successfully encrypting all of them, the final ciphertext will be formed.

- Add Round Key: The first key generated through an XOR function is used to pass the block data to the next step. This process returns the state array as input.
- Sub-Bytes: The following step takes the state array's bytes and converts them into a pair of equal parts. The columns and rows are mapped with a substitution box. This box will then generate new values for the final array.
- Shift Rows: The first row is skipped. The second row is moved one position to the left, while the third row moves two positions to the left. The last row is also shifted to the left.
- Mix Columns: The following procedure takes a constant matrix and adds a new column to the state array. This step is not performed in the last round. The next step involves getting the state array with all the columns multiplied.
- Add Round Key: The key for the round is XOR which is the result of the previous step. If the last round is the last one, then the state array will become the ciphertext for that block. On the other hand, if this is the first round, then the new state array will be used for the next one.

RUBIK'S CUBE ALGORITHM

Cryptographic cryptography is a type of security that is used to protect and secure data. It can be used to prevent unauthorized individuals from accessing it. Two of its main functions are encryption and decryption as shown in fig.2. With encryption, the original data is converted into an unreadable format. The new version of a message is different from the old one. Because of this, a hacker cannot access the data because the senders use an encryption method. This type of encryption is usually performed using key algorithms to protect the data. However, many companies protect their trade secrets by encrypting their data. The process of decrypting data involves taking the encoded or encrypted data and converting it into a form that can be accessed by a computer or human. This can be done manually or using the data's encryption keys.

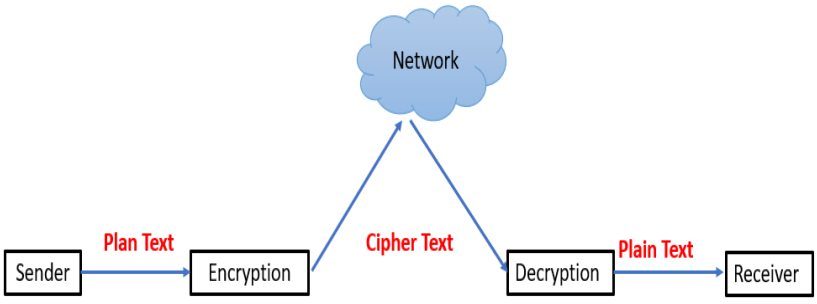

Figure. 3 Encryption and Decryption process

The concept of transforming the cube's design is inspired by the toy's design, which involves rotating it to remove the patterns on its surface. This paper presents a method that takes into consideration the cube's third-order model[20]. The special cube, which has 26 sub-blocks, can be rotated across the axis as shown in fig.3. In this paper,

we will first consider the various colors of the faces of the cube. We then use the third-order model to represent these faces. We then mark the various sub-blocks of the cube. Based on the third-order model, the top side of the cube is labeled as U, while the right and the front sides are represented as F, while the bottom side is labeled as R. The left side, on the other hand, is labeled as L, and the back and bottom sides are respectively B and D. Due to the relative position of the surface of the cube, we only consider its three layers: U, F, and R. The state of the cube can be seen in Figure 2 when the first layer of U is rotated 90 degrees to the left as shown in fig.4.

When the middle layer of a cube is rotated 90 degrees, its changes in the other surfaces, such as those shown in the R, B, and D, are cyclical. On the flip side, the changes in the D and U surfaces do not change. This principle can be utilized to piece the cubes together or modify their appearance. The paper presents the concept of image encryption, which takes into account the transformations of the Rubik's Cube. The subblocks of the cube are considered as pixels, and their position changes to generate an irregular image, which can be used to generate various patterns. The middle layer of the cube's rotation also affects the other surfaces' pixels' position, which can be used to generate various patterns. An encrypted image can be decrypted using the obtained key, which allows the recipient to retrieve its original content. Doing so helps improve the privacy and security of the information that's been transmitted through the process[21,22].

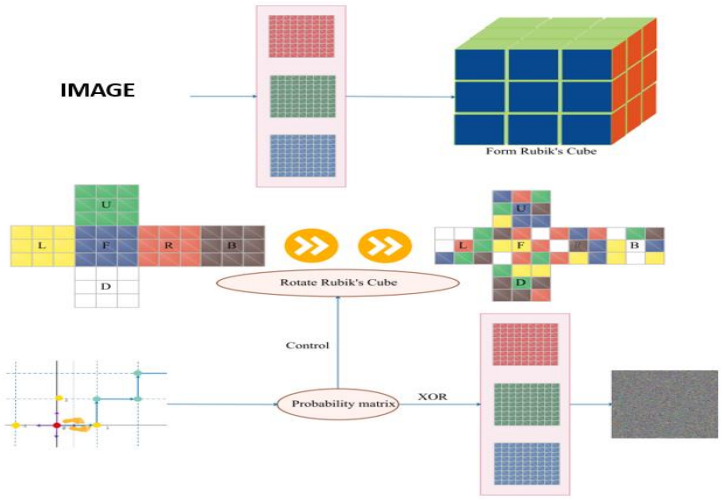

Figure. 4 Encryption flowchart using Rubik's cube

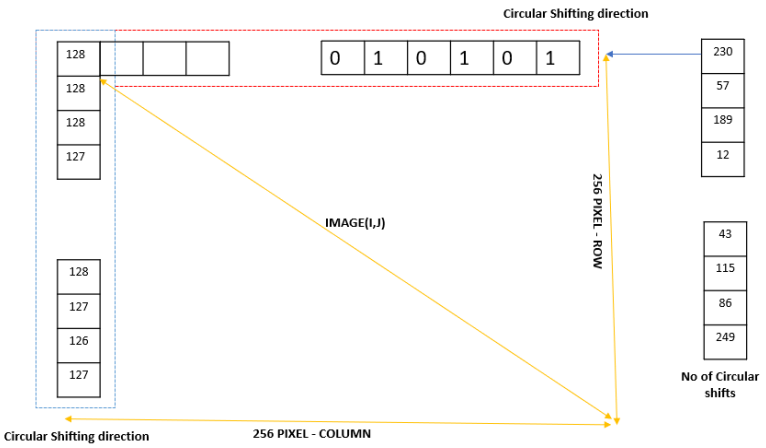

Figure. 5 Calculation of no. of circular shift

Algorithm 1 : Image Encryption based on Rubik's cube principle

1 *INPUT: Create two vectors, E_R and E_C, of lengths P and Q, at random.*
set A = (0, 1, 2,... 2^{α}-1) ← E_R(i) and E_C(j) each take a random value from the E_R and E_C cannot have constant values.

2 *Determine the maximum number of iterations (ITR),*
ITR_{max}, and set the counter ITR → 0.

3 *ITR ← ITR +1*

4 *Image I_o → row i, for every image*

i. *Computer the summation of row i →α(i)*
α(i) ← $\sum_{j=1}^{N} I_o(i,j)$, i= 1,2,....M

ii. *Calculate modulo of 2 ← α(i), depicted by $M_{\alpha(i)}$*

iii. *Row i is circularly shifted → right/ left by ER(i) positions*
If $M_{\alpha(i)=0}$ →Right Rotatory Shift
Else →Left Rotatory Shift

5 *Image I_o → column j, for every image*

Computer the summation of column →β(j)
β(j) ← $\sum_{i=1}^{M} I_o(i,j)$, i= 1,2,....N
Calculate modulo of 2 ← β(j), depicted by $M_{\beta(j)}$

Column j is circularly shifted → up or down by EC(i) positions
(Image pixels are shifted ER(i) positions to the left or right, and the first pixel moves to the last pixel).
If $M_{\beta(j)=0}$ → Rotatory Shift upward
Else →Rotatory Shift downward
Step 4 & step 5 → scrambled image (I_{SCRB})

6 *Vector EC applies the bitwise XOR operator to each row of scrambled image I_{SCRB} using these expressions:*
$I_1(2i-1,j) = I_{SCRB}(2i-1),j) \oplus Ec(j)$; $\oplus = XOR$
$I_1(2i-j) = I_{SCRB}(2i,j) \oplus$ *rot 180 (Ec(j)) ; rot 180 = spinning of Ec*

7 *Vector ER applies the bitwise XOR operator to each column of scrambled image I_1 using these expressions:*
$I_{encr}(i,\ 2j-1) = I_1(i,2j-1),j) \oplus E_R(j)$; $\oplus = XOR$
$I_{encr}(i,2j) = I_1(i,2j) \oplus$ *rot 180 (E_R(j)) ; rot 180 = spinning of E_R*
If ITR = ITR_{max} →I_{encr} created and encryption stage completed; else repeat from step 3

Algorithm 2 : Image Decryption based on Rubik's cube principle

1 *Initialize →ITR=0*

2 *ITR=ITR + 1*

3 *Implement XOR (⊕) operator → E_R encrypted image I_{encr}*
$I_1(i,\ 2j-1) = I_{encr}(i,2j-1),j) \oplus E_R(j)$; $\oplus = XOR$
$I_1(i,2j) = I_{encr}(i,2j) \oplus$ *rot 180 (E_R(j)) ; rot 180 = spinning of E_R*

4 *Using E_C vector and bitwise XOR operator → every image row I_1*
$I_{SCRB}(2i-1,\ j) = I_1(2-1,j) \oplus E_C(j)$; $\oplus = XOR$
$I_{SCRB}(2i,j) = I_1(2i,j) \oplus$ *rot 180 (E_C(j)) ; rot 180 = spinning of E_R*

5 *Image I_{SCRB} → column j, for every image*

i. *Computer the summation of column j* → $\beta_{SCRB(j)}$
$\beta_{SCRB(j)}=\sum_{i=1}^{M} I_{SCRB}(i,j)$, *j=1,2,....N*

ii. *Calculate modulo of 2* ← β_{SCRB}, *depicted by* $M_{\beta(SCRB)}$

iii. *Column j is circularly shifted* → *up or down by* $E_C(i)$ *positions*
If $M\ \beta_{SCRB}(j)= 0$ → *Rotatory Shift upward*
Else → *Rotatory Shift downward*

6 *Image* I_o → *row i, for every image*

i. $\alpha_{SRCB}(i) \leftarrow \sum_{j=1}^{N} I_{SCRB}(i,j)$, *i= 1,2,....M*

ii. *Calculate modulo of 2* ← $\alpha_{SCRB}(j)$, *depicted by* $M_{\alpha SCRB(j)}$

iii. *Row i is circularly shifted* → *left or right by* $E_R(i)$ *positions*
If $M_{\alpha SCRB(i)} = 0$ → *Right Rotatory Shift*
Else → *Left Rotatory Shift*

7 *If* $ITR=ITR_{max}$ → I_{ENCR} → *Decrypted*
Process completed;
Else Repeat → *step 2*

8 *Process Done*

RESULTS AND CONCLUSION

We use NPCR, UACI and histography calculation respectively, to test the number of pixels that have changed in the intensity of the ciphertext images when the difference between them is subtle.

Number of pixels change rate (NPCR)- The number of pixels change rate is the rate at which the number of pixels in an image changes when only one of its components is modified.

$$NPCR = \frac{\sum_{i,j} D(i,j)}{Width * Height} \text{ x } 100\%$$

= **NPCR = 0.995921921921922**

The unified average changing intensity (UACI) This measure compares the average intensity of the differences between the plain and ciphered images.

$$UACI = \frac{1}{Width * Height}\left[\sum_{i,j}\frac{|C_1(i,j) - C_2(i,j)|}{255}\right] \text{ x } 100\%$$

UACI = 0.15156175783626766

KEY GENERATION TIME: 0.0747990608215332

Encrypted Image

Encryption Time: 4.729905605316162

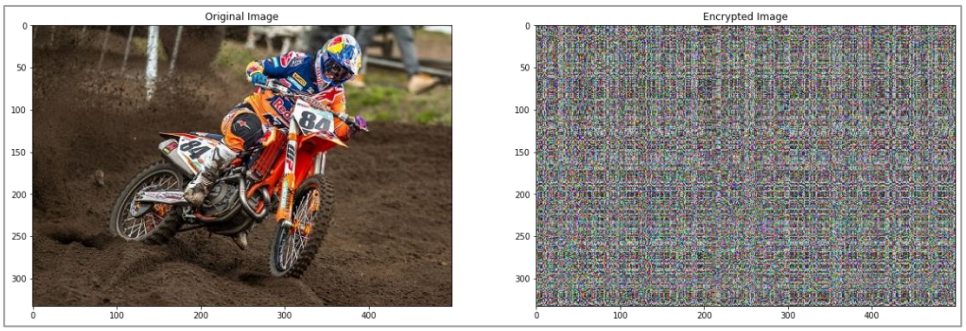

Figure. 6 original image to encryption

Decrypted Image

Decryption Time: 4.7363786697387695

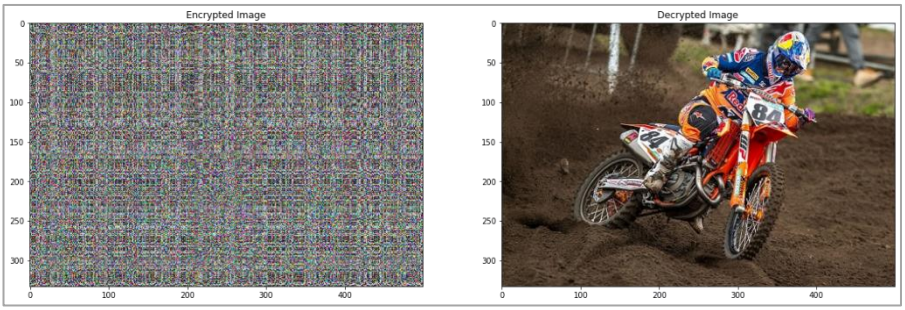

Figure. 7 Decrypted to original image

Histogram Calculation

One of the most widely used methods of illustrating the quality of an image's encryption is the ciphertext image histogram. Since a good encryption method tends to send a random-like image, it is desirable to have a uniformly distributed distribution of the data. Fig . 7,8 and 9 shows the various histogram for original and encrypted image as well.

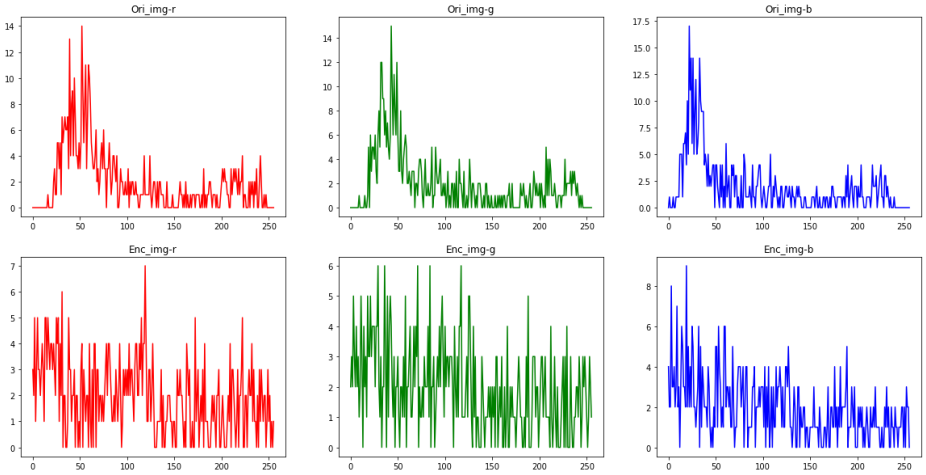

Figure. 8 Various histography

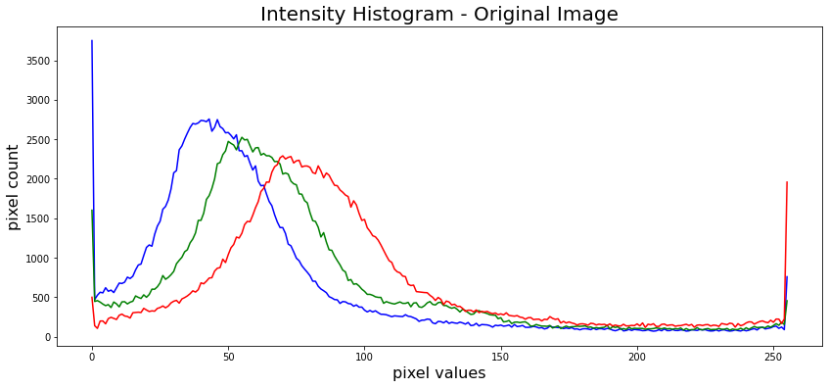

Figure. 9 Intensity histogram - original image

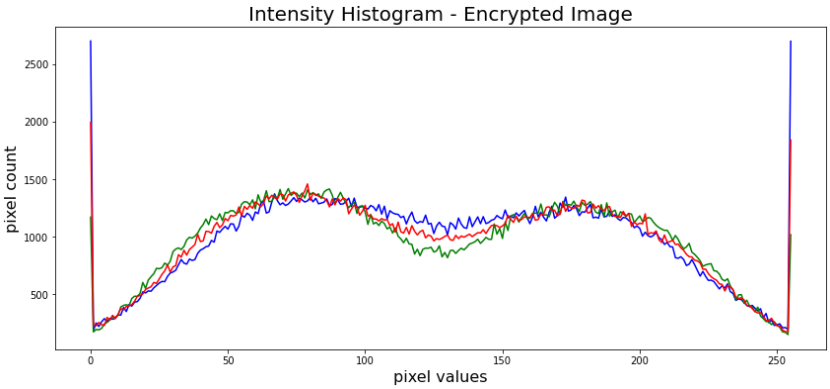

Figure 10 Intensity Histogram - Encrypted Image

CONCLUSION

Using the idea of a Rubik's Cube, we have developed a novel method that may be used for private key management in wireless networks inside this particular piece of research. This strategy makes use of a two-layered architecture, which is comprised of a Command Node and a cluster of Network Nodes. This system was developed in such a way that keys are produced dynamically for each and every sub transaction; as a result, it has the ability to significantly improve the security of wireless networks. This idea offers protection against a wide variety of potential security hazards. The generation of the private key in our architecture occurs through the use of a one-of-a-kind cubic design known as Rubik's cube, and it does so without the necessity of key dissemination through the network. Our architecture ensures that secure key management and a limited amount of key storage are used, both of which are essential components for achieving high levels of security in wireless networks which attains 99.59 as NPCR and 0.15 as UACI. The next step in developing this strategy is to find a more effective way to evaluate this solution in relation to various adversary and threat models. Keeping this model as a prototype while adding far more advanced features and capabilities would also make it feasible to develop a 'n' x 'n' cubic model, where 'n' is a rather big number. This model would then be capable of producing more profound and effective levels of protection.

ACKNOWLEDGEMENT

I[1] extend my sincere gratitude towards the Pimpri Chinchwad Education Trust's Pimpri Chinchwad College of Engineering and Research, Pune, India for their support to complete this research work.

REFERENCES

[1] S. Oukili, S. Bri, and A.V.S. Kumar, Colloq. Inf. Sci. Technol. Cist **0**, 901 (2016).

[2] V.K. Sharma, S. Kumar, and K.K. Mahapatra, J. Circuits, Syst. Comput. **25**, (2016).

[3] M. Baby Chellam and R. Natarajan, Arab. J. Sci. Eng. **43**, 6873 (2018).

[4] S. Oukili and S. Bri, J. Circuits, Syst. Comput. **26**, 1 (2017).

[5] Y. Weize and S. Kose, IEEE Trans. Circuits Syst. I Regul. Pap. **64**, 2934 (2017).

[6] S. Shanthi Rekha and P. Saravanan, J. Circuits, Syst. Comput. **28**, (2019).

[7] K.L. Tsai, F.Y. Leu, I. You, S.W. Chang, S.J. Hu, and H. Park, IEEE Access **7**, 146348 (2019).

[8] K. Shahbazi, S. Ko, and S. Member, **29**, 136 (2021).

[9] S.C. Naik and P.N. Mahalle, 2013 15th Int. Conf. Adv. Comput. Technol. ICACT 2013 (2013).

[10] J. Tang, A. Xu, Y. Jiang, Y. Zhang, H. Wen, and T. Zhang, Proc. 2020 IEEE Int. Conf. Artif. Intell. Inf. Syst. ICAIIS 2020 225 (2020).

[11] 2T MADHAVI KUMARI K LILY, 24 (2018).

[12] Y. Tu, J. Gan, Y. Hu, R. Jin, Z. Yang, and M. Liu, 2019 3rd IEEE Conf. Energy Internet Energy Syst. Integr. Ubiquitous Energy Netw. Connect. Everything, EI2 2019 2697 (2019).

[13] L. Wang, H. An, H. Zhu, and W. Liu, IEEE Internet Things J. **7**, 7590 (2020).

[14] M.B. Salunke, Rev. Gestão Inovação e Tecnol. **11**, 236 (2021).

[15] T.A. Shaikh, W.A. Mir, T. Rasool, and S. Sofi, *Machine Learning for Smart Agriculture and Precision Farming: Towards Making the Fields Talk* (Springer Netherlands, 2022).

[16] P. Radanliev, D. De Roure, M. Van Kleek, O. Santos, and U. Ani, AI Soc. **36**, 783 (2021).

[17] R. Bhandari and V.B. Kirubanand, Int. J. Electr. Comput. Eng. **9**, 3732 (2019).

[18] M. Jalasri and L. Lakshmanan, Cluster Comput. **2**, (2022).(n.d.).

[19] K. Loukhaoukha, J.Y. Chouinard, and A. Berdai, J. Electr. Comput. Eng. **2012**, (2012).

[20] P. Oza, V. Kathrecha, and P. Malvi, Third Int. Conf. Multidiscip. Res. Pract. IJRSI **4**, 239 (2016).

[21] R.V. Mudduluri, A. Golla, S. Raghava, and T.J. Sai, Int. J. Eng. Adv. Technol. **8958**, 24 (2022).