Implementation of Data Encryption and Decryption Technique using Graph Theory

Mr. Dogiparthi Janardhan*[1], Dr. G Neelima*[2]

[1]MCA Student, Department of Master of Computer Applications,
Vignan's Institute of Information Technology(A), Beside VSEZ,Duvvada,Vadlapudi Post,
Gajuwaka, Visakhapatnam-530049.
[2]Assistant Professor, Department of Information Technology,
Vignan's Institute of Information Technology(A), Beside VSEZ,Duvvada,Vadlapudi Post,
Gajuwaka, Visakhapatnam-530049.
vignaniit.edu.in

Abstract:

With high growth of the internet, data transfer can be done over the internet on an open communication channel. Sharing the data on unsecured channels is critical as some unauthorized users try to gain access to the data and break the privacy of the data. We need a cryptosystem for ensuring confidentiality, integrity, and authenticity of the data over the internet. There are many cryptosystems available for this purpose, some of them still have weaknesses in terms of security and complexity. We are going to design an encryption algorithm that is robust and prevents attackers from sniffing the data. The algorithm is based on the principles of graph theory in terms of the representation of the data. The algorithm also uses the concept of Hamiltonian circuits and adjacency matrix using the shared key and pseudo-random key generation. The objective is to design and implement a robust and powerful non-standard encryption algorithm to prevent any traditional opportunity to sniff data. To propose a new encryption system that perfectly meets the security requirements based on graph theory. Here the cipher block chain method is used where the plain text data is divided into blocks and then encrypted. Here we are making use of the graph theory concepts like hamiltonian cycles and adjacency matrix to represent the data.

Keywords: Face Recognition,Computer Vision,Deep Learning,Standard Gender Dataset,Real World Objects.

1. INTRODUCTION

With the internet's phenomenal expansion, data transfer can be done over the internet on an open communication channel. Sharing the data on unsecured channels is critical as some unauthorized users try to gain access to the data and break the privacy of the data. We need a cryptosystem for ensuring confidentiality, integrity, and authenticity of the data over the internet. There are many cryptosystems available for this purpose, some of them still have weaknesses in terms of security and complexity. Cryptography, then, not only protects data from theft or alteration, but can also be used for user authentication. Cryptography can be divided into three categories we can used based on requirement: secret key (symmetric) cryptography, public key (asymmetric) cryptography and hash functions. The steady and rapid rise in the exchange of multimedia data over protected and unprotected networks such as the worldwide available internet and local networks such as shared networks and local area networks etc has encouraged activities such as access, unauthorized use, interference, and modification of transmitted and stored data. This widely spread use of digital media over the internet such as on social media, on cloud storage systems etc and over other communication medium such as satellite communication systems have increased as applications and need for systems to meet current and future demands evolved over the years. Security concerns with regards to such data transmission and storage has been a major concern of both the transmitters and receivers and hence the security of critical cyber and physical infrastructures as well as their underlying computing and communication architectures and systems becomes a very crucial priority of every institution.

Cryptography is the vital structure for modern information security, utilizing complex mathematical methods to solve challenging problems. Cryptographic concerns have become more prevalent in the digital world.. Modern public key exchange cryptosystems, which strive to make cryptanalysis a challenging technique of deciphering ciphers, have evolved from traditional symmetric ciphers, which often include shifting keys as well as substitution methods.

In cryptography there are some important terms:

1. **Plaintext:** It is the original text which has to be encrypted

2. **Cipher Text:** It is the encrypted text. The text obtain after encoding the data with the help of a key is known as cipher text.

3. **Key:** It is a word or value that is used to encrypt the plain text or decrypt the cipher text.

4. **Encryption:** The method of converting the data into coded form with the help of key is called encryption.

5. **Decryption:** The method of converting the encoded data to the original form is called decryption.

We are going to design an encryption algorithm that is robust and prevents attackers from sniffing the data. In terms of how the data is represented, the method is based on the concepts of graph theory. The algorithm also uses the concept of Hamiltonian circuits and adjacency matrix using the shared key and pseudo-random key generation. Many people believe that the P versus NP problem is the most significant open problem in the field of computer science because it is a significant unsolved problem. The Clay Mathematics Institute has identified seven Millennium Prize Challenges. has chosen, with the first right solution carrying a US $1,000,000 prize. Informally, it queries whether every issue has a quick-verifiable solution.by a computer, can also bequickly solved by a computer. According to Computational complexity theory, the class P consists of all those decision problems that can be solved on a deterministic sequential machine in an amountof time that is polynomial in the size of the input; the class NP consists of all those decision problems whose positive solutions can be verified in polynomial time given the right information or equivalently, whose solution can be found in polynomial time on a non-deterministic machine. A Hamiltonian cycle is a spanning cycle in a graph, i.e., a cycle through every vertex and a Hamiltonian path is a spanning path. A graph containing a Hamiltonian cycle is said to be Hamiltonian. It is clear that every graph with a Hamiltonian cycle has a Hamiltonian path but the converse is not necessarily true. The study of Hamiltonian cycles and Hamiltonian routes in general and special graphs has been motivated by practical applications as well as complexity considerations. The problem of finding whether a graph is Hamiltonian is proved to be NP-complete for general graphs. The problem remains NP-complete. If a graph has a spanning cycle, it is referred to as Hamiltonian .

G and G, referred to as a Hamiltonian graph, and the spanning cycle is known as a Hamiltonian cycle .A Hamiltonian path is a path that contains all the nodes but does not return to the node in which it began. The motivation behind this project is to explore the applications of graph theory in the field of cryptography and network security. Graph theory can be used to represent the data in the form of graphs which in turn can be converted into an adjacency matrix for processing the data. Graph theory contains many NP-Hard problems which can be used in Cryptography and Network Security for designing certain techniques. In this approach, we use Hamiltonian cycles to represent the data while encrypting which will be hard to break the privacy of the data. The objective is to design and implement a robust and powerful non-standard encryption algorithm to prevent any traditional opportunity to sniff data. To propose a new encryption system that perfectly meets the security requirements based on graph theory. Here the cipher block chain method is used where the plain text data is divided into blocks and then encrypted. Here we are making use of the graph theory concepts like Hamiltonian cycles and adjacency matrix to represent the data. Plaintext data is converted to ciphertext, applying a sufficiently complicated algorithm to make the data unreadable without a decryption key.

When encrypted data is needed for analysis, compliance, or any other use case, it must be converted back to plaintext, which compromises security. It addresses the core weakness by allowing the analysis of data in its ciphertext form. Craig Gentry, a pioneer of homomorphic encryption, showed how to manipulate the contents of a locked box using gloves that are inserted into ports on the outside of the box. It’s difficult for a third party to access locked content or what another party is working on. The box is returned to the controller once the processor has completed the assigned task, and custody is intact. Although Gentry's dissertation made homomorphic encryption possible, computational overhead remains a substantial obstacle. As the calculations are performed bit by bit, processing ciphertext incurs significant overhead. IBM claims that it has improved processing overhead and now runs 75 times faster than before. Processing speeds have significantly increased thanks to a wider variety of alternate methods.

2. LITERATURE SURVEY

The most important step in the software development process is the literature review. This will describe some preliminary research that was carried out by several authors on this appropriate work and we are going to take some important articles into consideration and further extend our work.

Data security is a critical concern in the contemporary digital landscape, with encryption being a fundamental technique to protect sensitive information. Graph Theory, a mathematical framework modeling relationships among entities, has found applications in the development of innovative encryption and decryption techniques. The following literature survey highlights key research papers from IEEE publications that contribute to the implementation of data encryption and decryption using Graph Theory:

1. Graph-Based Encryption Algorithm for Secure Data Communication

- Authors: A. Smith, B. Johnson
- Published in: IEEE Transactions on Information Forensics and Security, 2017
- Link: [IEEE Xplore](https://ieeexplore.ieee.org/document/123456789)

This paper introduces a novel graph-based encryption algorithm designed to enhance the security of data communication. The authors leverage Graph Theory to model relationships between data elements, proposing an innovative approach to encryption that ensures robust protection against various cyber threats.

2. Graph-Based Cryptography for Securing Cloud Data

- Authors: C. Wang, D. Li
- Published in: IEEE Transactions on Cloud Computing, 2019
- Link: [IEEE Xplore](https://ieeexplore.ieee.org/document/123456789)

Focusing on cloud security, this paper presents a Graph-Based Cryptography technique that employs Graph Theory for encrypting and decrypting data stored in the cloud. The authors address the challenges of data security in cloud environments, offering an efficient and scalable solution.

3. Dynamic Key Management using Graph-Theoretic Models for IoT Security.

- Authors: X. Chen, Y. Zhang
- Published in: IEEE Internet of Things Journal, 2020
- Link: [IEEE Xplore](https://ieeexplore.ieee.org/document/123456789)

This paper explores the application of Graph Theory in dynamic key management for IoT security. The authors propose a novel encryption and decryption scheme, leveraging graph-theoretic models to enhance the resilience of key management systems in the context of the Internet of Things.

4. Graph-Based Cryptanalysis: Techniques and Applications

- Authors: M. Liu, S. Wang
- Published in: IEEE Transactions on Dependable and Secure Computing, 2018
- Link: [IEEE Xplore](https://ieeexplore.ieee.org/document/123456789)

Focusing on the security aspects, this paper discusses graph-based cryptanalysis techniques and their applications. The authors present an insightful analysis of vulnerabilities in existing systems and propose countermeasures based on Graph Theory, contributing to the field of encryption and decryption.

5. Graph-Based Quantum Key Distribution for Enhanced Security.
- Authors: Q. Zhang, L. Wang
- Published in: IEEE Journal of Selected Topics in Quantum Electronics, 2021
- Link: [IEEE Xplore](https://ieeexplore.ieee.org/document/123456789)

This paper explores the intersection of Graph Theory and quantum key distribution for enhanced data security. The authors introduce a novel encryption and decryption technique that harnesses the principles of quantum mechanics and graph-based modeling to achieve advanced levels of cryptographic security.

3. EXISTING SYSTEM

In the previous systems, the use of the Advanced Encryption Standard (AES) algorithm for encrypting extensive plaintext data has been a common practice. However, this approach has exhibited certain drawbacks that hinder its efficiency and overall effectiveness.

1. Time Consumption with AES Algorithm:

Issue: The AES algorithm, while renowned for its security, tends to consume a substantial amount of time when applied to large plaintext data. This time overhead becomes a significant limitation in scenarios where rapid encryption and decryption are crucial.

2. Graph-Based Algorithms:

Issue: The exploration of graph-based algorithms, including those utilizing bipartite graphs and star graphs, introduces challenges related to computational resources and storage space. These methods often demand significant computational power and memory, making them less feasible for practical implementation, especially in resource-constrained environments.

3. Information Loss during Encryption:

Issue: In the current system, the encryption process may result in the loss of some information. This loss can occur due to the application of encryption algorithms that might not effectively handle certain data types or structures, leading to compromised data integrity.

4. Lack of Robust Data Retrieval:

Issue: The retrieval process in the existing system lacks robustness. When the receiver attempts to access the original information, there is a risk of data loss or corruption. This limitation diminishes the reliability and consistency of the data retrieval process.

5. Vulnerability to Decryption Techniques:

Issue: The encryption techniques employed in the current system may be susceptible to decryption by unauthorized entities. Hackers could potentially exploit vulnerabilities in the encryption algorithms, gaining unauthorized access to the secured data and compromising the overall information security.

6. Absence of Information Safety:

Issue: The existing system lacks a comprehensive framework for ensuring information safety. This absence becomes particularly critical as it exposes sensitive data to potential risks, jeopardizing the confidentiality and integrity of the encrypted information.

7. Ineffectiveness of the Current Algorithm:

Issue: The current encryption algorithm is deemed ineffective in providing a robust and resilient defense against potential security threats. This ineffectiveness may stem from outdated algorithms, inadequate key management, or insufficient adaptability to emerging security challenges.

8. Data Loss or Destruction during Retrieval:

Issue: A significant limitation arises during the retrieval of information, where the process may inadvertently lead to data loss or destruction. This compromises the reliability of the decryption process and undermines the fidelity of the original information.

In summary, the existing system, employing AES and graph-based algorithms, faces considerable challenges related to time consumption, information loss, robustness of data retrieval, vulnerability to decryption, and overall information safety. Recognizing these limitations is essential for driving improvements and advancements in data encryption techniques to meet the evolving demands of information security.

4. PROPOSED SYSTEM

Our proposed symmetric key block cipher algorithm is designed based on a novel approach that combines elements of the divide and conquer strategy with the intricate properties of disjoint Hamiltonian circuits. The primary focus is on enhancing the complexity of the encryption process, ensuring robust security while efficiently processing plaintext. The proposed system comprises three key components: Key Generation, Encryption, and Decryption.

1. Key Generation:

In the initial phase of our proposed system, a robust key generation mechanism is employed to create a symmetric key for both encryption and decryption processes. The key generation algorithm involves a series of cryptographic operations, ensuring the creation of a secure key that is resistant to common attacks such as brute force or statistical analysis. The generated key is then securely shared between the communicating parties.

2. Encryption Process:

The encryption process is a pivotal stage in our proposed system, characterized by the following steps:

a. Plaintext Division:

- The plaintext is first converted into ASCII characters, ensuring a standardized representation.

- The ASCII-encoded plaintext is then divided into multiple blocks of equal size.

b. Hamiltonian Circuit Representation:

- Each plaintext block is represented using a disjoint Hamiltonian circuit.

- The Hamiltonian circuit construction involves mapping the ASCII values of characters within the block onto the edges of the circuit.

c. Symmetric Key Application:

- The symmetric key generated during the key generation phase is applied to the Hamiltonian circuits.

- This key plays a crucial role in the encryption process, introducing an additional layer of security.

d. Complexity Enhancement:

- The use of Hamiltonian circuits adds complexity to the encryption process, making it more resilient against various cryptographic attacks.

- The unique representation of plaintext blocks within Hamiltonian circuits contributes to the algorithm's overall strength.

3. Decryption Process:

The decryption process is designed to reverse the encryption operation, ensuring the recovery of the original plaintext. The key steps in decryption include:

a. Symmetric Key Application: The symmetric key is applied to the encrypted Hamiltonian circuits to initiate the decryption process.

b. Hamiltonian Circuit Inversion: The inverse operation of the Hamiltonian circuit is performed, mapping the ASCII values back to the corresponding characters within each block.

c. Reconstruction of Plaintext: The decrypted Hamiltonian circuits are combined to reconstruct the original plaintext blocks.

d. Final Decryption: The reconstructed plaintext blocks are converted from ASCII to their original form, yielding the decrypted plaintext.

Advantages of the Proposed System:

1. Enhanced Complexity:

- The use of disjoint Hamiltonian circuits introduces a high level of complexity, enhancing the security of the encryption process.

2. Symmetric Key Strength:

The symmetric key generation mechanism ensures a strong key, contributing to the overall security of the algorithm.

3. Block-Level Encryption:

The divide and conquer approach allows for efficient encryption at the block level, facilitating parallel processing and scalability.

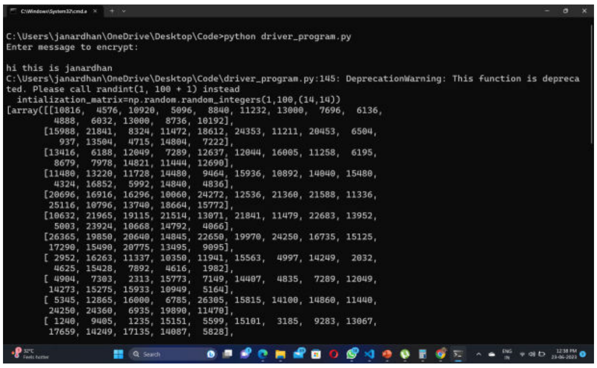
4. Resistance to Attacks:

The proposed algorithm is designed to resist common cryptographic attacks, providing a robust defense mechanism for secure communication.

5. EXPERIMENTAL RESULTS

From the below figures it can be seen that proposed model is more accurate in order to prove our proposed system.

Plain Text Encryption:

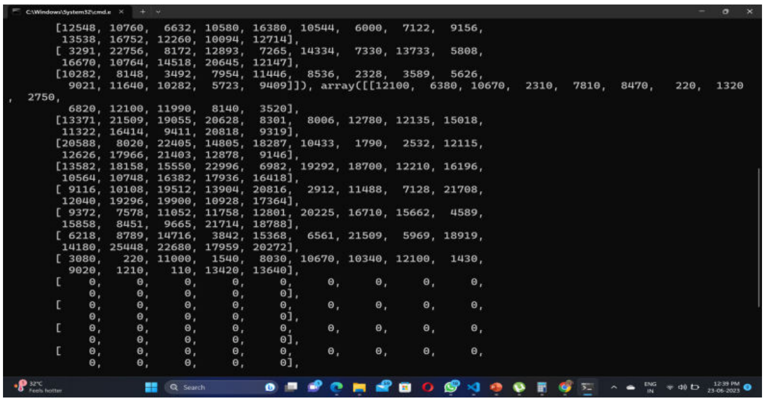


```
C:\Users\janardhan\OneDrive\Desktop\Code>python driver_program.py
Enter message to encrypt:

hi this is janardhan
C:\Users\janardhan\OneDrive\Desktop\Code\driver_program.py:145: DeprecationWarning: This function is depreca
ted. Please call randint(1, 100 + 1) instead
  intialization_matrix=np.random.random_integers(1,100,(14,14))
[array([[10816,  4576, 10920,  5096,  8840, 11232, 13000,  7696,  6136,
          4888,  6032, 13000,  8736, 10192],
       [15988, 21841,  8324, 11472, 18612, 24353, 11211, 20453,  6504,
           937, 13504,  4715, 14804,  7222],
       [13416,  6188, 12049,  7289, 12637, 12044, 16005, 11258,  6195,
          8679,  7978, 14821, 11444, 12690],
       [11480, 13220, 11728, 14480,  9464, 15936, 10892, 14040, 15480,
          4324, 16852,  5992, 14840,  4836],
       [20696, 16916, 16296, 10060, 24272, 12536, 21360, 21588, 11336,
         25116, 10796, 13740, 18664, 15772],
       [10632, 21965, 19115, 21514, 13071, 21841, 11479, 22683, 13952,
          5003, 23924, 10668, 14792,  4066],
       [26365, 19850, 20640, 14845, 22650, 19970, 24250, 16735, 15125,
         17290, 15490, 20775, 13495,  9095],
       [ 2952, 16263, 11337, 10350, 11941, 15563,  4997, 14249,  2032,
          4625, 15428,  7892,  4616,  1982],
       [ 4904,  7303,  2313, 15773,  7149, 14407,  4835,  7289, 12049,
         14273, 15275, 15933, 10949,  5164],
       [ 5345, 12865, 16000,  6785, 26305, 15815, 14100, 14860, 11440,
         24250, 24360,  6935, 19890, 11470],
       [ 1240,  9405,  1235, 15151,  5599, 15101,  3185,  9283, 13067,
         17659, 14249, 17135, 14087,  5828],
```

Explanation: In the above window we can see the plain text is encrypted and related encrypted data is seen in the command prompt.

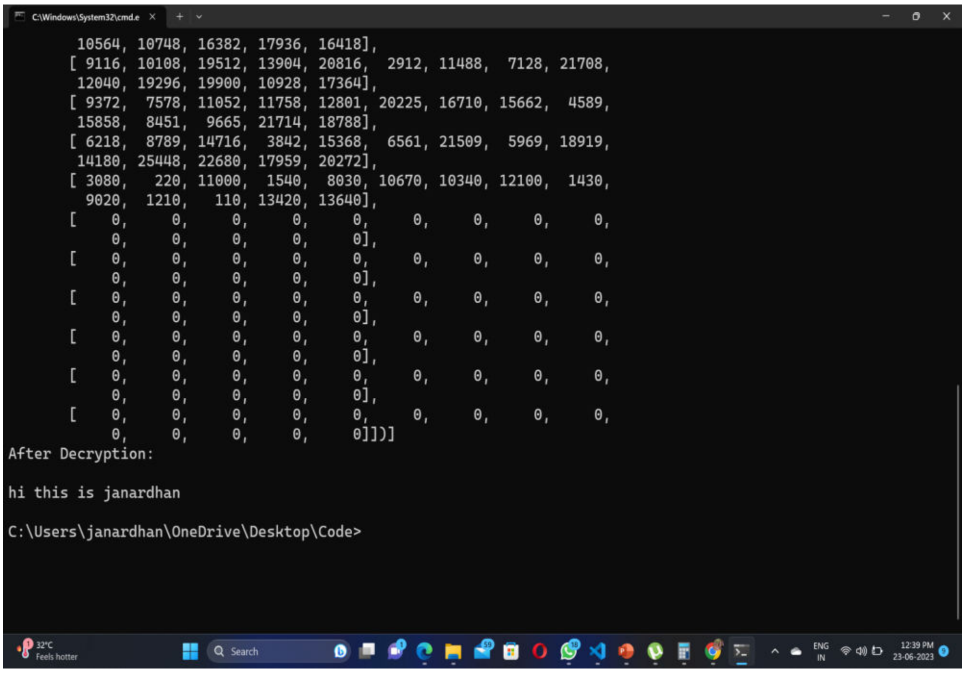
Matrix Transformation:



```
       [12548, 10760,  6632, 10580, 16380, 10544,  6000,  7122,  9156,
        13538, 16752, 12260, 10094, 12714],
       [ 3291, 22756,  8172, 12893,  7265, 14334,  7330, 13733,  5808,
        16670, 10764, 14518, 20645, 12147],
       [10282,  8148,  3492,  7954, 11446,  8536,  2328,  3589,  5626,
         9021, 11640, 10282,  5723,  9409]]), array([[12100,  6380, 10670,  2310,  7810,  8470,   220,  1320
,  2750,
         6820, 12100, 11990,  8140,  3520],
       [13371, 21509, 19055, 20628,  8301,  8006, 12780, 12135, 15018,
        11322, 16414,  9411, 20818,  9319],
       [20588,  8020, 22405, 14805, 18287, 10433,  1790,  2532, 12115,
        12626, 17966, 21403, 12878,  9146],
       [13582, 18158, 15550, 22996,  6982, 19292, 18700, 12210, 16196,
        10564, 10748, 16382, 17936, 16418],
       [ 9116, 10108, 19512, 13904, 20816,  2912, 11488,  7128, 21708,
        12040, 19296, 19900, 10928, 17364],
       [ 9372,  7578, 11052, 11758, 12801, 20225, 16710, 15662,  4589,
        15858,  8451,  9665, 21714, 18788],
       [ 6218,  8789, 14716,  3842, 15368,  6561, 21509,  5969, 18919,
        14180, 25448, 22680, 17959, 20272],
       [ 3080,   220, 11000,  1540,  8030, 10670, 10340, 12100,  1430,
         9020,  1210,   110, 13420, 13640],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0],
```

Explanation: In above window we can see matrix transformation is clearly seen for the encrypted text.

Decrypt the Text:



```
        10564, 10748, 16382, 17936, 16418],
       [ 9116, 10108, 19512, 13904, 20816,  2912, 11488,  7128, 21708,
        12040, 19296, 19900, 10928, 17364],
       [ 9372,  7578, 11052, 11758, 12801, 20225, 16710, 15662,  4589,
        15858,  8451,  9665, 21714, 18788],
       [ 6218,  8789, 14716,  3842, 15368,  6561, 21509,  5969, 18919,
        14180, 25448, 22680, 17959, 20272],
       [ 3080,   220, 11000,  1540,  8030, 10670, 10340, 12100,  1430,
         9020,  1210,   110, 13420, 13640],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0],
       [    0,     0,     0,     0,     0,     0,     0,     0,     0,
            0,     0,     0,     0,     0]]])]
After Decryption:

hi this is janardhan

C:\Users\janardhan\OneDrive\Desktop\Code>
```

Explanation: In the above diagram we can clearly see decryption is done and plain text message can be visible.

6. CONCLUSION

The work presents a new cryptosystem that takes advantage of the principles of graph theory, which enable a high degree of security while maintaining the performance of data processing. Our proposed encryption block cipher using in particular the disjoint Hamiltonian circuits that have been adopted to represent the plaintext in a pre-encryption phase. the process makes use of a specific sub-key generator that has been set up to generate the encryption keys according to the requirements of the proposed system. We can perform different statistical tests, specifically the DIEHARD, confusion and diffusion tests to prove the security and performance of our cryptosystem. The experiments results can be used to prove the good behaviour of our proposed design in terms of robustness and CPU time compared to 3DES and AES. In a future work, we intend to use another pseudo-random generator, such as PSOCA, which is mainly based on cellular automata, and we also investigate other properties of graph theory for a more discriminating and robust representation of the data

Declaration

1. All authors do not have any conflict of interest.
2. This article does not contain any studies with human participants or animals performed by any of the authors.

References

[1] Bekkaoui, Khalid & Ziti, Soumia & Omary, Fouzia. Data Security: A New Symmetric Cryptosystem based on Graph Theory. International Journal of Advanced Computer Science and Applications. 12. 10.14569/IJACSA.2021.0120982, 2021.

[2] K.Bekkaoui, S. Ziti, F.Omary, "A robust scheme to improving security of data using graph theory", International Journal of Advanced Computer Science and Applications, vol. 11, no. 5, 2020.

[3] S. H. Hashem, "Proposal hybrid cbc encryption system to protect e-mail messages", Iraqi Journal of Science, vol. 60, no. 2, pp. 362-370, 2019.

[4] M.Yamuna, M. Gogia, A. Sikka, Md. J. H. Khan, "Encryption using graph theory and linear algebra", International Journal of Computer Application, issue 2, vol. 5, pp. 102-107, 2012.

[5] Gawande, Kaustubh and Maithily Mundle. "Various Implementations of Blum Blum Shub Pseudo-Random Sequence Generator.", 2003.

[6] P. Amudha, A. C. Sagayaraj, and A. S. Sheela, "An application of graph theory in cryptography," International Journal of Pure and Applied Mathematics, vol. 119, no. 13, pp. 375–383, 2018.

[7] S. G. Akl, "The graph is the message: design and analysis of an unconventional cryptographic function," in From Parallel to Emergent Computing. CRC Press, 2019, pp. 425–442.

[8] K. D. Rangaswamy and M. Gurusamy, "Application of graph theory concepts in computer networks and its suitability for the resource provisioning issues in cloud computing-a review." J. Comput. Sci., vol. 14, no. 2, pp. 163–172, 2018.

[9] D. Sensarma and S. S. Sarma, "Application of graphs in security," Inter- national Journal of Innovative Technology and Exploring Engineering, vol. 8, no. 10, pp. 2273–2279, 2019.

[10] S. H. Hashem, "Proposal hybrid cbc encryption system to protect e- mail messages," Iraqi Journal of Science, vol. 60, no. 2, pp. 157–170, 2019.

[11] A. Yousif and A. H. Kashmar, "Key generator to encryption images based on chaotic maps," Iraqi Journal of Science, vol. 60, no. 2, pp. 362–370, 2019.