Bulletin of Electrical Engineering and Informatics
Vol. 14, No. 4, August 2025, pp. 3277~3288
ISSN: 2302-9285, DOI: 10.11591/eei.v14i4.9712

An enhanced key schedule mechanism to improve the security strength of the data encryption standard algorithm

Mareye Zeleke Mekonen[1], Komal Kumar Napa[2], Amogne Andulalem Ayalew[3,4], Bommy Manivannan[5], Tamilarasi Suresh[6], Janakiraman Senthil Murugan[7], Tsehay Admassu Assegie[8]
[1]Department of Information Technology, College of Engineering and Technology, Injibara University, Injibara, Ethiopia
[2]Department of Artificial Intelligence and Data Science, Saveetha Engineering College, Chennai, India
[3]Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, China
[4]University of Chinese Academy of Sciences, Beijing, China
[5]Department of Computer Science & Engineering, Madanapalle Institute of Technology & Science, Madanapalle, India
[6]Department of Information Technology, St. Peter's Institute of Higher Education and Research, Chennai, India
[7]Department of Computer Science and Engineering, Vel Tech High Tech Dr. Rangarajan Dr. Sakunthala Engineering College, Chennai, India
[8]School of Electronics Engineering, Department of Electronic and Electrical Engineering, Kyungpook National University, Daegu, Republic of Korea

Article Info

Article history:

Received Dec 15, 2024
Revised Jun 25, 2025
Accepted Jul 5, 2025

Keywords:

Avalanche effect
Data encryption standard
Hamming weight
Key scheduling algorithm
Symmetric key encryption

ABSTRACT

The rapid growth of internet accessibility requires strong data security measures, mainly for safeguarding sensitive information. Since many threats and attacks steal our private data. Data encryption standard (DES) is one of the cryptographic methods that uses a symmetric key encryption method to resist various types of cryptographic attacks. This work proposes an improved key scheduling algorithm (KSA) to enhance DES security. The modified KSA is evaluated using criteria such as frequency test, hamming weight, and bit difference to measure round key randomness and resilience. Moreover, the avalanche effect is evaluated to assess the diffusion and confusion character of the generated ciphertext. The final result indicates that the enhanced KSA attains better frequency distribution (0.89-1.0), increased hamming weight consistency (97.13%), and high bit transition rates compared to the original DES KSA. These enhancements demonstrate increased randomness and complexity, making the algorithm more resistant to brute-force and other cryptographic attacks. Our proposed work shows enhanced security capabilities, albeit with increased computational requirements, and establishes a foundation for future improvement in symmetric key cryptography.

Corresponding Author:

Mareye Zeleke Mekonen
Department of Information Technology, College of Engineering and Technology, Injibara University
Injibara, Ethiopia
Email: mareye132@gmail.com

1. INTRODUCTION

It is well known that web usage is rapidly increasing from time to time, and many people share public and personal information over the internet [1]. Through the network, massive amounts of datasets are transmitted digitally either through wired or wireless communication. While information passes from device to device, it is open for attackers to access in an unauthorized way. Thus, it requires security, as the data and information are sensitive and must be transmitted continuously. One of the most important techniques that can be very effective in securing confidential information is cryptography [2]-[4]. Cryptography can be

defined in various ways. It is the sense of encrypting and decrypting data using codes, or cryptography is a set of secure information and communication systems based on mathematical principles and algorithms. The algorithms play a vital role in providing data security against unauthorized users [5].

Cryptography algorithms can be categorized into two, namely symmetric(private) and asymmetric (public). Symmetric encryption and decryption require only one key, while asymmetric encryption and decryption require two keys, and hashing involves a fixed-length message digest [6], [7]. Even though these encryption standards are in use, denial of service attacks have become major security risks these days [8]. Data encryption standard (DES) is one of the symmetric encryption systems proposed by the National Security Agency (NSA). This type of cryptosystem has a faster encryption speed and lower computing costs. DES is still used as the main encryption core for two reasons: first, due to its complex algorithm structure, and second, because it is used in different practical applications. It is also one of the certified encryption standards and the safest technology [9]. However, how to strengthen the symmetric DES encryption system is still a very important issue [10]. Since DES has a long history, many scholars have questioned its security [11], [12]. The major questions include the insufficient length of DES and the insufficient resistance to brute-force attacks [13].

DES is a symmetric key block cipher encryption algorithm that takes a 64-bit plaintext, a 56-bit key (dropped parity bit), and produces a 64-bit ciphertext [14], [15]. In DES, splitting keys into two halves and swapping them might throw up the same result if they have continuous 1s and 0s short block size of the keys, and plain text. The primary challenge with DES lies in its fixed key schedule and predictable key shifting operations, which can be exploited by attackers to compromise encrypted data. Numerous works have been done in the enhancement of DES. For instance, [16], [17] had made efforts to enhance DES, focused on improving its security by increasing the key length and strengthening the weak round function to resist cryptographic attacks. However, these modifications still fall short of effectively countering brute force attacks, are challenging to implement, and require significant memory and processing resources. To address these limitations, we propose a novel enhanced key scheduling algorithm (KSA) for DES, aiming to increase algorithmic complexity and better resist brute force attempts since the security of the DES algorithm is directly affected by KSA [18]. Our approach is evaluated through rigorous testing, including frequency analysis, hamming weight measurement, and bit difference evaluation, to assess key randomness, the avalanche effect, and overall cryptographic strength. The results demonstrate that our enhanced KSA outperforms the classical DES KSA across all tested criteria, indicating improved security and resilience to unauthorized access. The remainder of this paper is organized as follows: section 2, the method that we followed while doing this work. Section 3 details the results and discussion of this study. Finally, section 4 concludes the paper and outlines directions for future research.

2. METHOD

In the proposed methodology, we introduce a novel transformation that incorporates a systematic approach to key manipulation through either left or right shifting. The direction of this shift is determined by the most significant bit (MSB) of the round key, which serves as a critical determinant in our encryption process. This transformation is further enhanced by a mixing operation designed to increase the diffusion of the key material, thus improving the overall security of the encryption algorithm. In addition to the shifting and mixing operations, we also implement up to three-bit shifting operations on the 56-bit keys. This multi-faceted approach allows for a more complex interplay between the bits of the key, thereby increasing the difficulty for potential attackers to reverse-engineer the encryption process. To ensure the robustness and reliability of our methodology, we conduct a thorough comparison of the bits generated by the round key at each stage of the transformation. This meticulous examination not only validates the correctness of our implementation but also provides insights into the behavior of the key under various conditions.

2.1. The proposed key divider of 56 bits- the bit key for enhancing the key schedule

In the classical DES encryption algorithm, the 56-bit key is divided into two halves, each containing 28 bits. These halves are designated for left and right shifts during the key scheduling process. This traditional approach has been effective; however, it limits the complexity and variability of the generated round keys. By maintaining only two halves, the key schedule can become predictable, which may compromise security in certain applications.

In our enhanced KSA, we come up with a novel method by splitting the key into four sub-keys, each 14 bits in length. This modification not only increases the number of possible combinations but also adds an extra layer of complexity to the generation of round keys, illustrated in Figure 1. By diversifying the key structure, we aim to improve the overall security of the encryption process, making it more resilient against potential attacks and enhancing its robustness in various cryptographic applications.

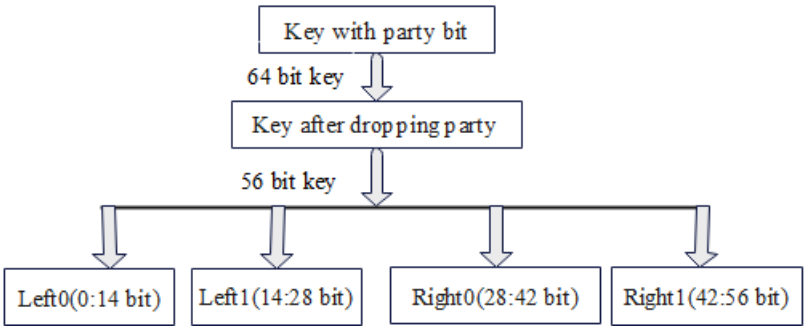


Figure 1. Enhanced DES key generation algorithm

2.2. The proposed most significant bit checker for enhancing the key schedule

In the traditional DES KSA, the subkeys undergo a circular left shift of either one or two bits, determined by the specific round being processed. In contrast, the main key is subjected solely to left shifts throughout the encryption process. This conventional method, while effective, can limit the variability and unpredictability of the generated round keys. In our proposed approach, we enhance this mechanism by allowing for circular shifts that can occur in either direction, left or right, based on the characteristics of the generated round keys. This flexibility produces an additional layer of complexity to the key scheduling process, as illustrated in Figure 2.

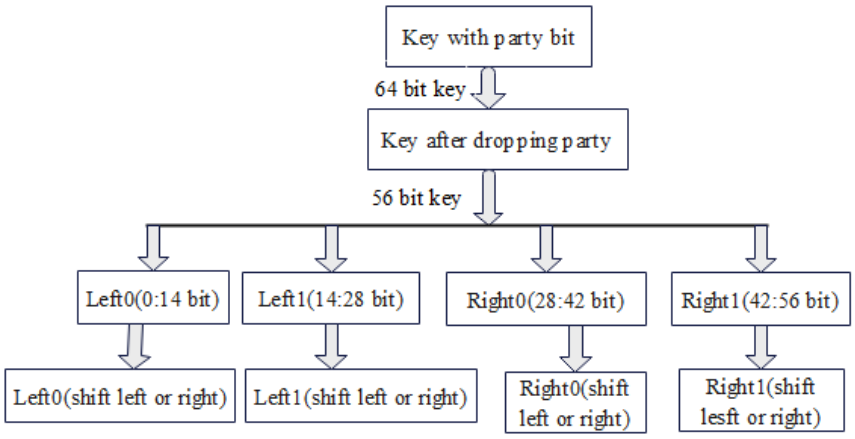


Figure 2. MSB checker

2.3. Proposed counter operation

After subdividing the keys into smaller segments, the next step involves utilizing a counter to accurately assess the distribution of ones and zeros within each segment. This counting process is crucial, as it provides valuable insights into the binary composition of the keys, highlighting areas of strength and potential vulnerability. By quantifying the number of ones and zeros, we can gain a deeper understanding of the key's structure, which can significantly influence the subsequent encryption process. This information serves as a foundational element for the next phase, where it is relayed to a comparator that evaluates the overall distribution of these binary values across the key segments.

2.4. The comparator for shifting one, two, or three bits

The comparator plays a crucial role in evaluating the distribution of zeros and ones within the round keys during each iteration. By analyzing this binary composition, it determines the optimal shifting strategy, allowing for bit shifts of up to three positions. This adaptability in the shifting process not only enhances the overall flexibility of key management but also contributes to a more dynamic and secure encryption mechanism. By tailoring the shifts according to the specific distribution patterns observed, the system can effectively improve the unpredictability and complexity of the generated round keys, ultimately strengthening the encryption process.

2.4.1. The proposed mixing shifted key operations

The proposed method presents a mixing operation for the shifted keys, enhancing the complexity and security of the generated round keys. This mixing process ensures a high degree of confusion among the round keys, preventing any sub-key from revealing information about other sub-keys or the secret keys themselves. By systematically mixing the shifted keys in the specific order of left0, right0, left1, and right, the approach effectively obfuscates the relationships between the keys. This strategic mixing not only fortifies the encryption process but also significantly increases resistance against potential cryptanalysis, making it more challenging for attackers to deduce any underlying patterns or correlations within the key structure.

2.4.1. Mathematical model evaluation of the original and proposed circular shifting operation

The traditional DES key shifting algorithm is relatively straightforward and can be susceptible to attacks due to its predictable nature, involving only two one-directional shifts. This predictability allows attackers a 50% chance of correctly guessing the shifted keys. In contrast, the proposed model significantly enhances security by introducing a more complex shifting mechanism. By incorporating variability in the direction and magnitude of shifts, allowing for shifts to occur either left or right, and by two or three bits. This complexity lowers the attacker's chances of accurately guessing the shifted keys to approximately 16.66%. As a result, the proposed model offers a more robust defense against potential cryptanalysis, making it considerably harder for attackers to exploit any predictable patterns in the key generation process.

3. RESULTS AND DISCUSSION

To implement these KSA approaches, we used the Python language for experimental purposes. Input to the algorithm is a block of 64-bit plaintext (data) and a 64-bit key, and a 64-bit ciphertext as output, and the proposed KSA is evaluated from two perspectives in terms of round keys and the ciphertext. Our proposed algorithms are implemented and compared with the original DES algorithms based on the following evaluation metrics: avalanche effect, the effect of confusion and diffusion, hamming distance, randomness, and independence among sub-keys.

3.1. Round key evaluation

A secret key is a key used to create the round keys for subsequent rounds since DES is a round block encryption algorithm [19]. The selection of round keys in block cipher algorithms varies based on the design and specifications of the algorithm. Each algorithm includes its key scheduling process to generate round keys that support its encryption process. This assures algorithm-specific safety and operational efficiency [20]. To assess the round keys in this work, we utilized the following statistical tests: include frequency test, the bit difference between round keys, and the hamming-weight test.

3.2. Frequency test

The frequency test depends on the proportion of 1s and 0s [21]. The percentage of zeros and ones should be near 50% for random data. This test is used to check whether the partial keys generated by a KSA contain equal numbers of 1s and 0s. Therefore, this test is performed to prove the randomness of round keys. The evaluation can be expressed using (1) to (3):

$$p - value = erfc\left(\frac{Sobs}{\sqrt{2}}\right) \quad (1)$$

$$Sobs = \frac{|Sn|}{\sqrt{n}} \quad (2)$$

$$Sn = X1 + X2 + X3 \ldots\ldots + Xn \quad (3)$$

Here, n represents the length of the bit string. Sn is the absolute value of the sequence being observed, and $Sobs$ is the absolute value divided by the square root of the length of the string. Moreover, $erfc(.)$ is a complementary error function. According to the decision rule, the sequence is considered if p-value≥0.01, then the sequence is concluded as being random with a confidence of 99%; otherwise, it is non-random [22]. In our evaluation, we used as an example the key of "1572863AD4A34900" and the plain of "8B9575D6B51A48D2" for evaluating the frequency of round keys.

As shown in the frequency test in Table 1, the frequency test result of the original KSA is 0.42 to 0.89, but in the proposed KSA is 0.89 to 1, so the proposed approach has a better frequency test than the original. Therefore, the proposed KSA approach generates more random round keys than the original KSA.

Figure 3 shows the frequency test in terms of p-value observation. As shown in the graph, the original KSA covers from 0.0 to around 0.92, whereas the enhanced KSA registers from 0.8 up to 1.0, showing that the p-value registers better in our enhanced KSA.

Table 1. Frequency test of round keys, internet of p-value observation

Round keys	Enhanced DES	Original DES
Round one	1	0.79
Round two	0.89	0.69
Round three	0.89	0.69
Round four	0.89	0.69
Round five	1	0.89
Round six	1	0.56
Round seven	1	0.69
Round eight	0.89	0.89
Round nine	0.89	0.42
Round ten	0.89	0.79
Round eleven	1	0.79
Round twelve	0.79	0.89
Round thirty	0.89	0.69

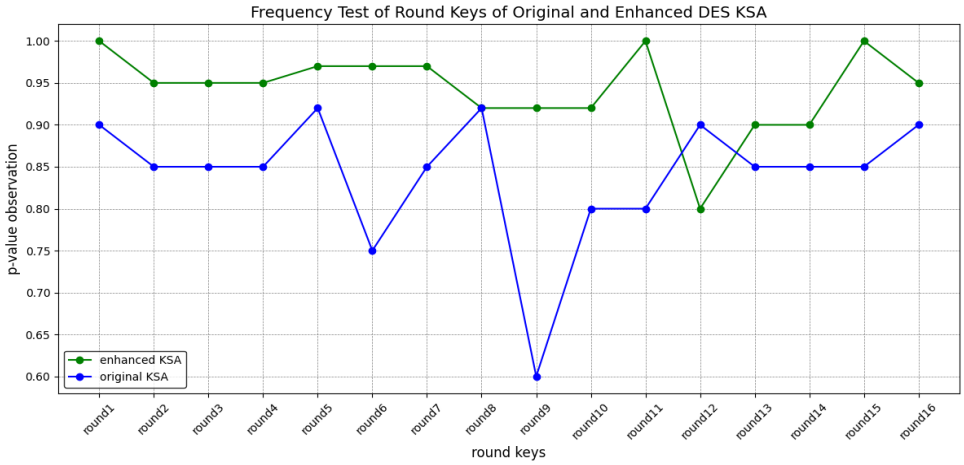


Figure 3. Frequency test line graph in terms of p-value observation

3.3. Bit differences between round keys test

This test aims to observe the complex relationship between round keys, and this complexity leads to the evaluation of round key confusion. Round keys are XORed directly with round data so random round keys should be XORed with round data to make them less predictable and more random, so the generated round key should be random to generate random cipher text. Tables 2 and 3 show the percentage of bit difference between two consecutive round keys generated with the original KSA and the improved KSA, respectively.

Table 2. Bit difference between round keys for enhanced DES KSA

XOR between round keys	No. of bit diff	% of bit diff
K1⊕k2	27	56.25
K2⊕k3	28	58.33
K3⊕k4	28	58.33
K4⊕k5	27	56.25
K5⊕k6	28	58.33
K6⊕k7	28	58.33
K7⊕k8	25	53.19
K8⊕k9	28	58.33
K9⊕k10	28	58.33
Key10⊕K11	27	56.25
K11⊕K12	28	58.33
K12⊕K13	27	56.25
K13⊕K14	26	54.16
K14⊕K15	29	60.41
K15⊕K16	27	56.25

Table 3. A bit difference between round keys for the original DES KSA

XOR between round keys	No. of bit diff	% of bit diff
K1⊕k2	31	64.58
K2⊕k3	24	50
K3⊕k4	24	50
K4⊕k5	26	54.16
K5⊕k6	25	52.08
K6⊕k7	23	47.91
K7⊕k8	24	50
K8⊕k9	27	56.25
K9⊕k10	24	50
K10⊕K11	26	54.16
K11⊕K12	25	52.08
K12⊕K13	24	50
K13⊕K14	25	52.08
K14⊕K15	22	45.83
K15⊕K16	31	64.58

The test evaluates the bit difference for a random secret key and a random plaintext as input for both the original DES and the improved DES. The result shows that the original DES KSA produces 45% to 64.58% of the bit difference between key rounds. Ideally, the bit transition between round keys should be at least 50%. The improved DES KSA generates 53.19% to 60.41% of the bit difference between round keys, so our proposed algorithm achieves greater confusion than the original DES algorithm. A bit difference between generated round keys is depicted visually in Figure 4. So, our enhanced one fluctuated slightly smaller than the original one. This result indicates our algorithm registers better confusion than the original.

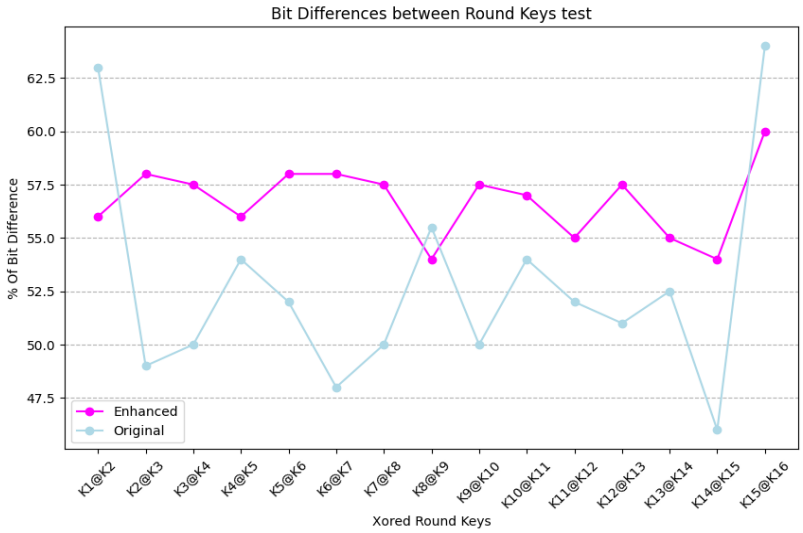


Figure 4. A bit differences between the generated round keys line graph

3.4. Hamming-weight test

The hamming weight test ensures that a perfect balance of zeros and ones between round keys is achieved to maximize randomness in the encoded text [23], [24]. Consequently, an ideally balanced binary string of n bits should have a hamming weight of n/2. To calculate the hamming weight, use (4):

$$\text{Hamming Weight\%} = \frac{\text{Number of non-zero bits}}{\text{Total bits}} * 100 \quad (4)$$

Round keys with low hamming weight can serve as tools for differential crypto attacks. The original DES KSA generates round keys with low hamming weight compared to the improved KSA with random keys and random plaintext.

The original DES KSA generates round keys with hamming weights between 18 and 23. The improved algorithm produces round keys with hamming weights between 22 and 25, with a perfect value of

24 in each round. The perfect value for the hamming weight of each secret key, together with its round keys, should be 384 bits. As shown in Table 4, the average hamming weight value for the original DES KSA is 343, and for the improved DES KSA is 373. The percentage hamming weight value of the original KSA is 342/384×100=89.32%, while the improved KSA value of 373/384×100=97.13%. These results show that the improved KSA achieves a significantly better hamming weight.

The hamming weight of round keys using a special secret key is depicted in Figure 5. As shown in the figure, the original involves 18 to 23, whereas the improved one covers 18 to 25. As shown in the graph, both original and improved degrade towards 18 in round 9. However, the hamming weight of the enhanced one increases drastically from 22 to 25 in round 16. This shows that the enhanced is much better than the original hamming weight.

Table 4. Hamming-weight test of original DES and enhanced DES KSA

Round	Original hamming weight	Enhanced hamming weight
1	22	24
2	21	25
3	21	25
4	21	23
5	23	24
6	20	24
7	21	24
8	23	23
9	18	23
10	22	25
11	22	24
12	23	22
13	21	25
14	22	25
15	22	24
16	21	25
Average	343	373

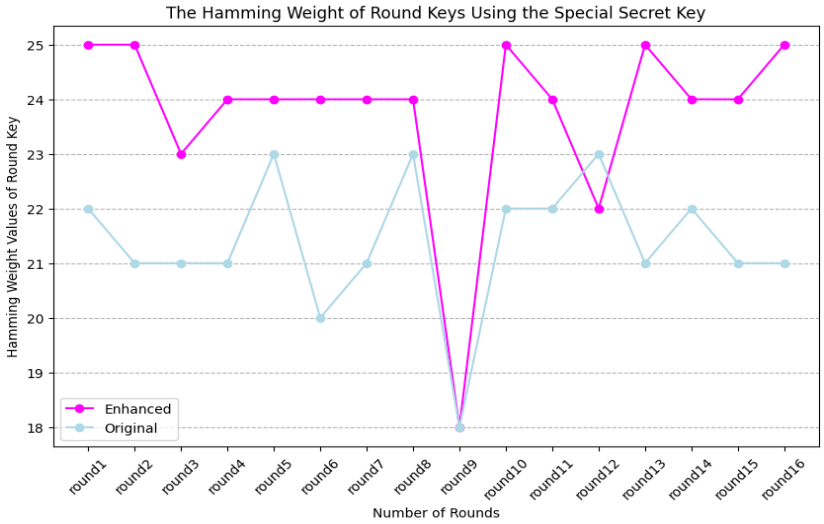


Figure 5. Hamming weight test line graph

3.5. Cipher text evaluation

Cipher text evaluation is essential for analyzing the strength and effectiveness of the enhanced KSA. By assessing the properties of the generated ciphertext, the flexibility of the encryption algorithm against attacks can be identified. For evaluating the impact of the improved KSA on encryption security, the avalanche effect test is an important metric. This test measures the degree of modification in the ciphertext when a small change, such as flipping a single bit, is made to the plaintext or key. A strong KSA approves that even minor modifications outputs in substantial differences in the ciphertext, which implies that there is strong diffusion and high security. Moreover, extra tests such as frequency analysis and randomness evaluation can foil the assessment, giving a comprehensive understanding of the algorithm's performance.

3.5.1. Avalanche effect results

The avalanche effect measures the effect of a change in the ciphertext by changing one bit in the associated plaintext or key [25]. It is one of the most selective methods to measure the algorithm security of cryptography. This test measures the nonlinear properties of the proposed algorithm; the value for a good avalanche effect should be 50% of the bits to ensure that every bit of the ciphertext is affected by the plaintext or the key bit. Avalanche effect test, subtest, number of secret keys as input, number of plaintexts, and out are depicted in Table 5. A high degree of diffusion and confusion, i.e., a high avalanche result, is desired, and the formula for the avalanche effect is as (5):

$$\text{Avalanche effect\%} = \frac{\text{Number of flipped bits in ciphertext}}{\text{Number of total bits in ciphertext}} * 100 \quad (5)$$

Table 1. Avalanche effect test, on plaintexts and output

Test name	Sub test	No. Input plain text	No. input key	Output bits
Avalanche effect confusion	Key	1 Random plaintext	56 random key	3584
Avalanche effect diffusion	Plain text	64 Random plain text	1 random key	4096

The output can be compared by grouping the absolute error (AbE) into four class ranges: AbE between 30% and 40%, AbE between 40% and 50%, AbE between 50% and 60%, and AbE greater than 60%. This grouping helps for a more detailed analysis of the distribution and severity of errors. By classifying the results into these ranges, trends and patterns can be determined, highlighting areas of significant deviation and giving a clearer picture of model performance. This structured approach helps in analyzing overall accuracy and determining potential areas for improvement. The avalanche effect diffusion of the original and enhanced for DES KSA is shown in Figure 6. As shown in the graph, the avalanche effect is lower than the original DES KSA, showing that the enhanced KSA registers a better avalanche effect result than the original DES KSA.

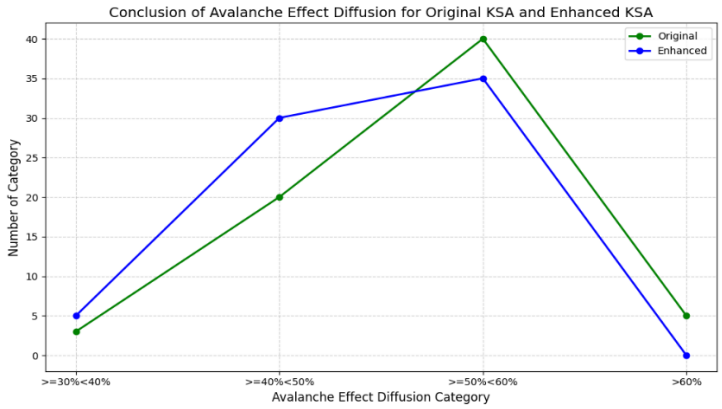


Figure 6. Line graph of avalanche effect diffusion of the original and enhanced DES KSA

As indicated in Table 6, the avalanche effect diffusion of the enhanced KSA decreased from 5% to 2% in the 30% to 40% range, and from 29% to 18% in the 40% to 50% range. Conversely, in the 50% to 60% range, it increased from 30% to 39%, while the category above 60% rose from 0% to 5%. These results demonstrate that the improved DES KSA exhibits superior avalanche effect diffusion compared to the original KSA. Similarly, Table 7 shows that the avalanche effect confusion of the enhanced KSA decreased from 2% to 1% in the 30% to 40% range, and from 22% to 16% in the 40% to 50% range. In contrast, the 50% to 60% range saw a slight increase from 30% to 31%, and the category above 60% increased from 2% to 8%. This indicates that the enhanced DES KSA also provides better avalanche effect confusion than its predecessor. The proposed KSA was assessed based on the generated round key and ciphertext, focusing on the avalanche effect in terms of diffusion and confusion. The evaluation of the round key involved frequency tests, hamming weight tests, and bit difference analyses between generated round keys. The results suggest that our proposed method offers greater security strength than the existing KSA. Although the proposed KSA incorporates additional operations to enhance the security of the DES algorithm, it requires more memory and processing time compared to the current KSA. Researchers can further strengthen security by utilizing

more complex functions for round key generation and by increasing both the number of rounds and key sizes. Here, our work limitation is taking up memory and processing time.

Table 6. Conclusion of avalanche effect diffusion for original KSA and enhanced KSA

Plain text	>=30%<40%	>=40%<50%	>=50%<60%	>60	Category
Random	5	29	30	0	Original KSA
	2	18	39	5	Enhanced KSA

Table 7. Conclusion of the avalanche effect confusion for the original KSA and the enhanced KSA

Key	>=30%<40%	>=40%<50%	>=50%<60%	>60	Category
Random	2	22	30	2	Original KSA
	1	16	31	8	Enhanced KSA

4. DISCUSSION

The analysis of the avalanche effect, diffusion, and confusion between the original and enhanced KSA for the DES reveals significant improvements in the proposed method. As illustrated in Figure 6, the enhanced KSA demonstrates a lower avalanche effect compared to the original DES KSA, indicating that it produces better results in terms of diffusion. Specifically, the results from Table 6 show a notable decrease in avalanche effect diffusion in the lower percentage ranges (from 5% to 2% in the 30% to 40% range, and from 29% to 18% in the 40% to 50% range), while showing an increase in higher ranges (from 30% to 39% in the 50% to 60% range). This suggests that the enhanced KSA is more effective at distributing changes across the ciphertext, thereby improving security.

Moreover, the frequency test results presented in Table 1 highlight a significant improvement in randomness with the proposed KSA. The frequency test result for the original KSA ranges from 0.42 to 0.89, whereas the proposed KSA achieves a range of 0.89 to 1. This indicates that our proposed approach generates a more random round key than the original KSA. Figure 3 further illustrates this point, showing that while the original KSA covers from 0.0 to around 0.92, the enhanced KSA registers from 0.8 up to 1.0, reflecting better p-value observations in our enhanced KSA. In terms of the bit difference between generated round keys, the results indicate that the original DES KSA produces between 45% to 64.58% of bit difference between key rounds, while the improved DES KSA generates between 53.19% to 60.41%. Ideally, the bit transition between round keys should be at least 50%, thus demonstrating that our proposed algorithm achieves greater confusion than the original DES algorithm. Figure 4 visually depicts these differences, indicating that while our enhanced method fluctuates slightly smaller than the original, it nonetheless registers better confusion.

In comparison to earlier research, our proposed KSA incorporates additional operations to strengthen the security of the DES algorithm. Consequently, our contribution lies in enhancing the KSA for DES, resulting in performance that surpasses current state-of-the-art methods due to improved security, as demonstrated through various metrics discussed above.

5. CONCLUSION

This study demonstrates that the enhanced KSA for the DES significantly strengthens security compared to the original KSA. Through a comprehensive evaluation of key generation and ciphertext characteristics, we have shown that our proposed KSA enhances the avalanche effect diffusion, resulting in a more effective distribution of changes across the ciphertext. Additionally, our findings indicate improved randomness in round key generation, as evidenced by favorable frequency test results. The enhanced KSA consistently achieves higher hamming weights and greater bit differences between generated round keys, which contributes to increased confusion and resistance to cryptanalysis. These advancements address critical vulnerabilities in the original DES KSA, providing a more robust framework for secure data transmission, an essential requirement in an era marked by escalating data breaches and cyber threats. The implications of these findings extend beyond theoretical contributions; they hold practical significance for various applications, including finance, healthcare, and secure communications, where data integrity and confidentiality are paramount. By improving diffusion and confusion mechanisms, our enhanced KSA promotes stronger encryption practices that can be adopted in real-world scenarios. Looking ahead, future research could explore the integration of more complex mathematical functions into the KSA to further enhance key generation security. Additionally, increasing the number of rounds and key sizes could provide added protection against emerging threats. Overall, our proposed KSA represents a crucial advancement in encryption technology, paving the way for developing more secure algorithms capable of withstanding future challenges in data security.

ACKNOWLEDGMENTS

We would like to extend our gratitude to the reviewers for their time and insightful feedback provided during the review process of this paper.

FUNDING INFORMATION

No funds or financial support for this work.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Mareye Zeleke Mekonen	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	
Komal Kumar Napa	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓		
Amogne Andulalem Ayalew			✓	✓	✓	✓	✓			✓	✓		✓	
Bommy Manivannan	✓	✓	✓		✓		✓	✓	✓				✓	✓
Tamilarasi Suresh	✓				✓	✓	✓	✓	✓	✓		✓		✓
Janakiraman Senthil Murugan	✓		✓		✓		✓		✓					✓
Tsehay Admassu Assegie			✓			✓	✓		✓	✓	✓			✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY

There is no data which used for this study.

REFERENCES

[1] A. Jan, S. A. Parah, M. Hussan, and B. A. Malik, "Double layer security using crypto-stego techniques: a comprehensive review," *Health and Technology*, vol. 12, no. 1, pp. 9–31, 2022, doi: 10.1007/s12553-021-00602-1.
[2] A. Vuppala, R. S. Roshan, S. Nawaz, and J. V. R. Ravindra, "An Efficient Optimization and Secured Triple Data Encryption Standard Using Enhanced Key Scheduling Algorithm," *Procedia Computer Science*, vol. 171, no. 2019, pp. 1054–1063, 2020, doi: 10.1016/j.procs.2020.04.113.
[3] C. Mu, "Application of optimizing advanced encryption standard encryption algorithm in secure communication of vehicle controller area network bus," *Frontiers in Mechanical Engineering*, vol. 10, 2024, doi: 10.3389/fmech.2024.1407665.
[4] A. Almalawi, S. Hassan, A. Fahad, and A. I. Khan, "A Hybrid Cryptographic Mechanism for Secure Data Transmission in Edge AI Networks," *International Journal of Computational Intelligence Systems*, vol. 17, no. 1, 2024, doi: 10.1007/s44196-024-00417-8.
[5] B. E. H. H. Hamouda, "Comparative Study of Different Cryptographic Algorithms," *Journal of Information Security*, vol. 11, no. 03, pp. 138–148, 2020, doi: 10.4236/jis.2020.113009.
[6] M. M. Hoobi, "Improved Structure of Data Encryption Standard Algorithm," *Journal of Southwest Jiaotong University*, vol. 55, no. 5, 2020, doi: 10.35741/issn.0258-2724.55.5.12.
[7] C. Riman and P. E. Abi-Char, "Comparative Analysis of Block Cipher-Based Encryption Algorithms: A Survey," *Computer Fraud*, vol. 3, no. 1, pp. 1–7, 2015, doi: 10.12691/iscf-3-1-1.
[8] K. Dworak and U. Boryczka, "Breaking Data Encryption Standard with a Reduced Number of Rounds Using Metaheuristics Differential Cryptanalysis," *Entropy*, vol. 23, no. 12, pp. 1-21, 2021, doi: 10.3390/e23121697.
[9] Y. Ding, "Application and Performance Evaluation of DES Data Encryption Algorithm in Computer Information Security Technology," *Journal of Artificial Intelligence Practice*, vol. 7, no. 3, 2024, doi: 10.23977/jaip.2024.070301.
[10] P. Wilson, *Design Optimization Example*, Design Recipes for FPGAs, Elsevier, 2016, pp. 265–281, doi: 10.1016/B978-0-08-

097129-2.00019-2.
[11] L. M. Shamala, G. Zayaraz, K. Vivekanandan, and V. Vijayalakshmi, "Lightweight cryptography algorithms for internet of things enabled networks: An overview," *Journal of Physics: Conference Series*, vol. 1717, no. 1, 2021, doi: 10.1088/1742-6596/1717/1/012072.
[12] T. A. Assegie and P. S. Nair, "A review on software defined network security risks and challenges," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 17, no. 6, pp. 3168-3174, Dec. 2019, doi: 10.12928/TELKOMNIKA.v17i6.13119.
[13] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, "A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish," *Procedia Computer Science*, vol. 78, pp. 617–624, 2016, doi: 10.1016/j.procs.2016.02.108.
[14] I. A. Awan, M. Shiraz, M. U. Hashmi, Q. Shaheen, R. Akhtar, and A. Ditta, "Secure Framework Enhancing AES Algorithm in Cloud Computing," *Security and Communication Networks*, vol. 2020, 2020, doi: 10.1155/2020/8863345.
[15] W. Y. Zibideh and M. M. Matalgah, "Modified data encryption standard encryption algorithm with improved error performance and enhanced security in wireless fading channels," *Security and Communication Networks*, vol. 8, no. 4, pp. 565–573, 2015, doi: 10.1002/sec.1003.
[16] A. J. Hintaw, S. Manickam, S. Karuppayah, M. A. Aladaileh, M. F. Aboalmaaly, and S. U. A. Laghari, "A Robust Security Scheme Based on Enhanced Symmetric Algorithm for MQTT in the Internet of Things," *IEEE Access*, vol. 11, pp. 43019–43040, 2023, doi: 10.1109/ACCESS.2023.3267718.
[17] S. Afzal, M. Yousaf, H. Afzal, N. Alharbe, and M. R. Mufti, "Cryptographic Strength Evaluation of Key Schedule Algorithms," *Security and Communication Networks*, vol. 2020, pp. 1–9, 2020, doi: 10.1155/2020/3189601.
[18] K. Rakhimberdiev, A. Bozorov, and M. Berdimurodov, "Round Key Generation Algorithm Used in Symmetric Block Encryption Algorithms to Ensure the Security of Economic Systems," *ACM International Conference Proceeding Series*, pp. 548–554, 2023, doi: 10.1145/3644713.3644794.
[19] C. Bhaya, A. K. Pal, and S. H. Islam, "A novel image encryption and decryption scheme by using DNA computing," *Advances in Computers*, vol. 129, pp. 129–172, 2023, doi: 10.1016/bs.adcom.2022.08.010.
[20] Z. Mengdi, Z. Xiaojuan, Z. Yayun, and M. Siwei, "Overview of Randomness Test on Cryptographic Algorithms," *Journal of Physics: Conference Series*, vol. 1861, no. 1, pp. 1-8, Mar. 2021, doi: 10.1088/1742-6596/1861/1/012009.
[21] L. Zhao, Y. Chi, Z. Xu, and Z. Yue, "Block Cipher Identification Scheme Based on Hamming Weight Distribution," *IEEE Access*, vol. 11, pp. 21364–21373, 2023, doi: 10.1109/ACCESS.2023.3249753.
[22] D. Ray, Y. Sao, S. Biswas, and S. S. Ali, "On Securing Cryptographic ICs against Scan-based Attacks: A Hamming Weight Distribution Perspective," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 19, no. 2, pp. 1–20, 2023, doi: 10.1145/3577215.
[23] J. Kaur and K. R. R. Kumar, "Analysis of Avalanche effect in Cryptographic Algorithms," in *2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, Noida, India, 2022, pp. 1-4, doi: 10.1109/ICRITO56286.2022.9965127.
[24] A. K. Mandal and A. Tiwari, "Comparative study of Avalanche effect in DES using binary codes," in *2012 National Conference on Computing and Communication Systems, NCCCS 2012*, 2012, vol. 46, pp. 247–250, doi: 10.1109/NCCCS.2012.6413007.
[25] K. Mohamed, M. N. M. Pauzi, F. H. H. M. Ali, and S. Ariffin, "Analyse On Avalanche Effect In Cryptography Algorithm," in *Proceedings of the International Conference on Sustainable Practices, Development and Urbanisation (IConsPADU 2021), 16 November 2021, Universiti Selangor (UNISEL), Malaysia*, 2022, pp. 610–618, doi: 10.15405/epms.2022.10.57.

BIOGRAPHIES OF AUTHORS

Mareye Zeleke Mekonen graduated with a bachelor's degree in 2015 from Hawassa University with a major in Information Technology in Ethiopia and an advanced standing master's degree in Information Technology in 2019 from Jimma University in Ethiopia. He is currently employed with the College of Engineering and Technology as a lecturer in the Department of Information Technology. His areas of interest in study are computer vision, deep learning, NLP, artificial intelligence, machine learning, bioinformatics, and IR. He is the author of two papers and one book. He can be contacted at email: mareye132@gmail.com.

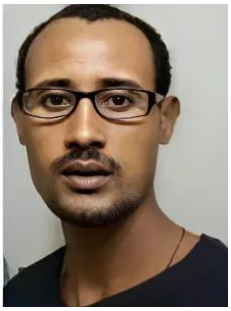
Komal Kumar Napa is currently working as an assistant professor (SG) in the Department of Artificial Intelligence and Data Science at Saveetha Engineering College, Chennai, India. His research interests include machine learning, data mining, and cloud computing. He can be contacted at email: komalkumarnapa@gmail.com.

Amogne Andulalem Ayalew holds a Master of Science degree in Information Technology from Bahir Dar University, Ethiopia (2021), and a Bachelor of Science degree in Information Technology from Debre Markos University, Ethiopia (2017). Currently he is a Ph.D student in Institute of High Energy Physics, Chinese Academy of Sciences. His research interests include natural language processing, image processing, machine learning, data mining, network security, cloud computing, and artificial intelligence, . He can be contacted at amogneandualem@gmail.com.

Bommy Manivannan is currently working as an assistant professor in the Department of Computer Science and Engineering at Madanapalle Institute of Technology & Science, Madanapalle, Andhra Pradesh, India. She can be contacted at email: bommym@mits.ac.in.

Tamilarasi Suresh is heading the Department of Information Technology at St. Peter's Institute of Higher Education and Research. Her broad areas of research interest are Machine learning, software engineering, artificial intelligence, and cloud computing. Her journey in SPIHER started 2 years ago, and it's been a wonderful environment to nurture her academic capabilities. She can be contacted at email: tamilarasisu@gmail.com.

Janakiraman Senthil Murugan is currently working as an associate professor in the Department of Computer Science and Engineering at Vel Tech High Tech Dr. Rangarajan Dr. Sakunthala Engineering College, Chennai, India. His research interests include machine learning and image processing. He can be contacted at email: jsenthilmuruganmtech@gmail.com.



Tsehay Admassu Assegie holds a Master of Science degree in Computer Science from Andhra University, India, 2016. He received his B.Sc. in Computer Science from Dilla University, Ethiopia, in 2013. His research interests include machine learning and biomedical image processing. He has published many articles in reputed international journals and at international conferences. He can be contacted at email: tsehayadmassu2006@gmail.com.