International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 8, pp 182-192, August-2024

https://doi.org/10.47001/IRJIET/2024.808020

Enhancing AES Security through Advanced S-Box Design: Strategies and Solutions

[1]Sura Nabil Hamed Alsweedy, [2]Sufyan Salim Aldabbagh

[1,2]Computer Science Department, College of Computer Science and Mathematics, University of Mosul, Mosul-Iraq
Authors E-mail: [1]sura.21csp86@student.uomosul.edu.iq, [2]drsufyan.salim@uomosul.edu.iq

Abstract - **AES is yet one of the prominent cryptographic algorithms of the 21st century with the reputation of excellent performance and reliability. AES revolves around the S-Box, a nonlinear substitutive table that is essential to attain the level of cryptosecurity. In this paper, there are several methods discussed as how to increase AES S-Box functionality with regard to new cryptographic threats and their effectiveness. We detail the consequences of raising S-Box length and size, dynamical S-Box producing, higher nonlinearity, and efficient methods of S-Box calculation by means of hardware and software. Furthermore, it explains the issues with these improvements and how they impact security measures & relevant computations. Here it is possible to state that applying all these modern approaches, it is possible to strengthen the AES S-Box essentially and ensure compliance with present day demands to cryptographic solutions.**

Keywords: Advanced Encryption Standard (AES), S-Box Design, Cryptographic Security, Dynamic S-Box Generation, Nonlinearity, Chaotic Maps, Symmetric Encryption, Cryptanalysis.

I. INTRODUCTION

Penetration to the fact that transmission of information over public networks is a major worry in modern society is worthy to be protected. Thus, for providing the solution with strong end-to-end protection, cryptographic encryption must be used. Of these techniques, the AES stands out as the most preferable cryptographic primitive in the current dispensation [1]. The initial suggestion of AES was known as Rijndael algorithm, and based on the public surveys conducted by the NIST it was chosen in October 2000 as a standard for private key cryptography. Not long after that it was approved by The Federal Information Processing Standards (FIPS) that has cemented its usage in secure communication [2].

AES has been subjected to different cryptanalysis attacks where the main purpose is to determine the flaws of the algorithm used by this protocol. Most specifically, linear and differential attacks applied to AES have target the S-Box, which is essential to create confusion in the process of encryption. The S-Box is another nonlinear block that translates the input data into the output data that are entirely different from the input data and this increases the complexity of the encryption result [3].

These attacks of the S-Box have instigated several other propounded solutions that are meant to enhance the security of AES. Since linear and differential attacks depend on the knowledge of S-Box, different approaches have been offered to eliminate these vices. One of the notable categories of works is known as the introduction of chaotic maps, which utilize chaos theory for strengthening the security to a new level [4]. The other common approach is to use the collection of S-Boxes known as key-dependent S-Boxes whereby these S-Boxes vary with the encryption key and this tends to complicate the tactic for the attackers [5].

Dynamic key-dependent S-Box solutions have become a characteristic advancement to the moves in cryptographic security. These solutions alter the S-Box dependent on the encryption key and therefore provides a different S-Box in every session making it much more immune to attacks that would presume a normal S-Box. Despite the evidence which proved that typically key-dependent S-Box methods execute slower as compared to key-independent methods, it is realized that it provides more security to the execution process by making the process more complex and more difficult to predict [5].

S-Box is an essential component of each block cipher and, of course, AES as well. Its main purpose is to provide non-linearity into the encryption process and it is critical for excluding both, linear and differential cryptanalysis. The vulnerability of the S-Box is the key factor which defines the resistance against the attacks and at the same time the maximization of the non-linear component and minimum differential propagation probability defines the strength of the round function [6]. However, it should be noted that AES S-Box has been earlier generated by using a fixed irreducible polynomial and an affine constant; nevertheless, this is quite flexible and it is possible to create other S-Boxes depending on the current threats [7].

II. ENCRYPTION

A mathematical field in computer science called encryption plays a crucial role in the conversion of multimedia data, guaranteeing safe transport and storage. Using a secret key, raw photos, movies, or audio is transformed into an incomprehensible format in this procedure. Three categories exist for encryption techniques, as shown in Figure (1) [8].

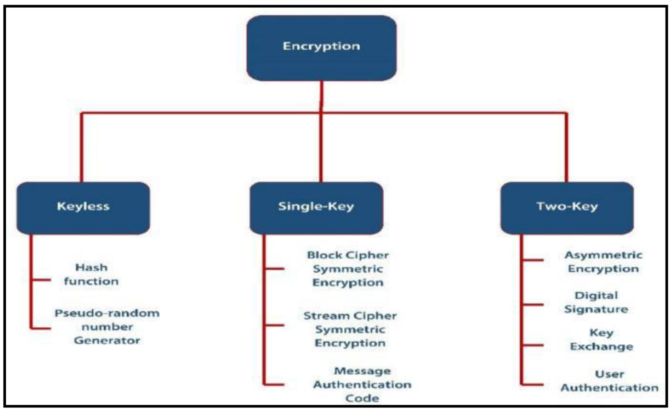
Encryption
Keyless
Single-Key
Two-Key
Hash function
Pseudo-random number Generator
Block Cipher Symmetric Encryption
Stream Cipher Symmetric Encryption
Message Authentication Code
Asymmetric Encryption
Digital Signature
Key Exchange
User Authentication

Figure 1: Encryption Method Types [8]

A. Keyless Methods

The encryption transformation function functions without keys in keyless approaches. The hash function and pseudorandom number generator are two examples. Text with variable length is transformed into a fixed-length hash code by the hash function, and a deterministic but ostensibly random sequence of bits is generated by the pseudorandom number generator [9].

B. Single-Key Methods (Symmetric Encryption)

Single-key techniques are symmetric encryption techniques that use a single key for both encryption and decryption. This key can be shared by several parties to secure communication traffic, or it can be known only by one person to protect stored data. Block cyphers, which operate on data in blocks, and stream cyphers, which operate on data as a sequence of bits, are the two types of symmetric algorithms [10].

C. Two-Key Methods (Asymmetric Encryption)

Asymmetric encryption, or two-key approaches, use two keys: a private key that is known to just one person and a public key that is known to many. Key exchange, user authentication, and digital signatures are just a few of the uses for asymmetric encryption. To guarantee data integrity and origin in digital signatures, an asymmetric method computes a value corresponding to the data. A symmetric key is safely distributed to several parties through key exchange, and the authenticity of a user gaining access to a service or application is confirmed by user authentication [8].

Traditional algorithms like AES and DES are insufficient for high-quality encryption due to the particular characteristics of multimedia data, such as strong correlations and redundancies, especially in real-time applications. As a result, several encryption techniques have been put out to protect multimedia information. These include DNA-based, elliptic curve-based, chaotic-based, and metaheuristic-based techniques [8].

D. Chaotic-Based Encryption

Because of their erratic behaviors, chaotic maps are especially useful for security applications. It is difficult for attackers to predict or replicate the encryption key because of their sensitivity to initial conditions. With benefits including ergodicity, sensitivity to beginning conditions, and the requirement for a large key space, chaotic-based encryption techniques improve security in applications involving the encryption of images, videos, and audio[11]. The unique properties of chaotic maps position them as a potent tool in diverse security implementations.

Recently, Distributed Network Architecture (DNA) technology has attracted a lot of interest from a variety of industries, especially information science and medicine. DNA has demonstrated the ability to store and transform data into a variety of genetic codes due to its innate genetic code. With the use of this technology, biological research simulation settings have been created. Most significantly, DNA has been used in encryption to store and secure multimedia data [12]. DNA has many benefits when it comes to encryption, some of which are its low power consumption, parallel processing capabilities, and ability to store massive datasets. These qualities make DNA a potentially useful technology for encryption applications[13].

Metaheuristic approaches have gained popularity, especially when high optimization levels are sought. Their increasing utilization in encryption reflects their effectiveness in tackling complex problems [14]. Metaheuristic approaches are used in encryption for two main purposes: first, to generate numerous encrypted data and choose the best optimized one; second, to optimize the starting parameters of chaotic maps to produce effective encryption keys. Based on metaheuristic techniques, researchers have developed a variety of encryption techniques while taking into account various factors and viewpoints.

Cellular automata, renowned for their complexity, have found wide application in cryptography, particularly in generating pseudorandom numbers. Their specific rules for

ISSN (online): 2581-3048
Volume 8, Issue 8, pp 182-192, August-2024
https://doi.org/10.47001/IRJIET/2024.808020

generating random sequences contribute to their robustness and efficiency. The parallelism property and straightforward hardware structure of cellular automata make them especially suitable for image encryption, rendering them a significant option for encryption methods [15].

III. ADVANCED ENCRYPTION STANDARD (AES)

The major disadvantage of DES and 3DES is that in software implementation, they are rather slow. This is the case of the original DES, or Data Encryption Standard, that was conceived for mid-1970s hardware and does not efficiently translate into software code. 3DES, thus making the number of rounds three times that of DES, is even slower by this virtue. Another drawback of both DES and 3DES is that both have a block size of 64 bits; here, a larger block size is more desirable, although it slightly decreases efficiency. Because of these constraints, 3DES cannot be used in the long run. Therefore, in parallel to 3DES, in 1997 NIST invited proposals for a new encryption standard, the Advanced Encryption Standard (AES), which is expected not only to be at least as secure as 3DES but also much more efficient. NIST's requirements laid down for AES as, the cipher had to be a block based and symmetric one with a block size of 128 bits and key size of 128, 192 and 256 bits. To start with, fifteen algorithms were approved for assessing at the first level. The first cut down the list to ten algorithms and this was then further reduced to five algorithms in the second rung. In November 2001 once the evaluation is done NIST released the final standard and they opted for Rijndael as the AES algorithm. Rijndael algorithm was designed by two Belgium's scientists namely Dr. Joan Daemen and Dr. Vincent Rijmen[16].

A. The AES Cipher

The Rijndael proposal for AES stipulated for a cipher for which the block as well as the key size could be fixed at 128, 192 or 256 bits. Nevertheless, three mentioned key length options are used within the AES specification while the block length is fixed at 128 bits. Some of the parameters of AES change according to the key length. Taking the 128-bit as the most probable one regarding the current usage of key length, the plaintext block size is set equal to 128, and encryption may proceed through 10 rounds. An extension of this is that, when the key length is 192-bits, the number of rounds becomes 12, and for 256-bits key length, it rises to 14. But, the size of the round key remains the same and that is 128 bits in any case of the two key lengths. The size of the expanded key is dissimilar; where the key size is the 128 bits, the expanded key is 176 bytes; for 192 bits, the expanded key is 208 bytes; and for 256 bits, then the expanded key is 240 bytes[17].

Rijndael, the algorithm selected for AES, was designed with several key characteristics in mind: including resistances against all the known attacks, high velocity and minification besides the capability of addressing all platforms, and simplification of the design[18].

The BASE operation AES uses in the encryption and decryption methods is for block of 128 bid. These blocks are first put into a new State array Before each of the six steps, changes are made to the State. To the end of the final stage, the State array is shifted to an output matrix tentatively called State out matrix[19].

Also, the disposition of the bytes that is within the matrix is done in a column major technique. For instance, the 1st four bits which comprise the first nucleolus of the 128-bit plain text input are shown at the top row of the input matrix, the next four bits compose the second row, and so on. Similarly, the first four bytes of the expanded key create an MSB word that occupies the position in the first column of the KSA matrix[20].

In contrast to the Feistel structure where a half of the data block alters a second half and after that the sands are interchanged, in the AES structure the whole data block is processed in parallel in each round through substitution and permutation. This is also true for another AES finalist, Rijndael, which also does not employ Feistel structure.

The input key is now held in forty-four 32-bit words, designated w[i]. And for every round four different words, 128 bits in total, are used as the round key as shown in the Figure (1).

- AES employs four stages in its rounds: a typical life cycle of a product includes one permutation stage and three substitution stages.
- Substitute Bytes: Also uses an S-box for the byte-by-byte substitution of the block.
- ShiftRows: A basic task of permutation, in which one of the objects is shifted by one position up when combined with another object in the same group.
- MixColumns: An arithmetic step in which the information elements are shifted between the GF(28) representation of the first and second octets, or similarly for the second and third.
- AddRoundKey: A logical exclusory operation involving the current block and a part of the expanded key to offer a new pattern.

The structure of the advanced enrolment system is as follows: Similar to the encryption process, the decryption also begins with an AddRoundKey process, followed by the same sequence of the nine rounds which include all four processes

ISSN (online): 2581-3048
Volume 8, Issue 8, pp 182-192, August-2024
https://doi.org/10.47001/IRJIET/2024.808020

each, and the final one which includes only three processes. The components of the full encryption round are demonstrated in Figure (2).

In AES, the key is incorporated during the AddRoundKey process and that is why both AES-encryption and AES-de-encryption start and end with this process, any other process on these points cannot provide security because it can be reversed without the key. Even though the analysed cipher stage, AddRoundKey is similar to Vernam cipher and is not secure on its own, the preceding stages, Substitute Bytes, ShiftRows, and MixColumns provide confusion, diffusion, and nonlinearity, collectively, in association with the key-dependent AddRoundKey stage provide a further boost to security. Each stage is reversible: The operations of Byte substitution, shift rows, and Mix columns have their operational counterparts during decryption and; the Add round key stage is operationally decrypted by XORing it with the same round key. The decryption algorithm uses the key in the exactly the opposite order to that used in the encryption process and it is not a simple inversion owing to the design of AES encryption. Such a requirement of the process is confirmed by the fact that the processes of encryption and decryption are quite similar. Also, the last level of the encryption and decryption component contains three steps to ensure reversibility [21].

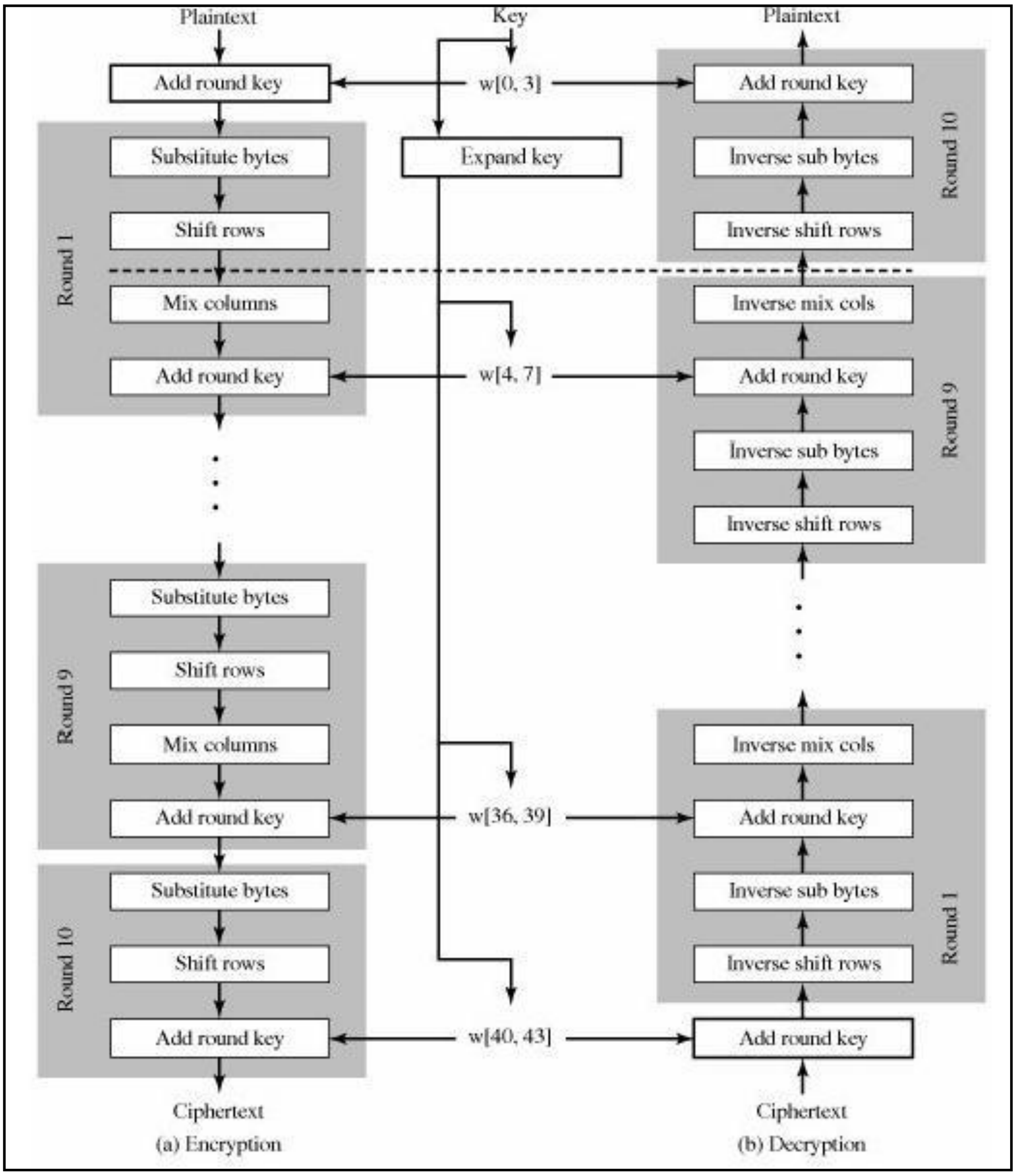

Figure 2: AES Encryption Round

B. Substitute Bytes Transformation

Substitute Bytes Transformation (SBT) is a method of cryptology where each byte within the block of message is substituted by another byte according to some particular rule; usually it follows the algorithm, which is called the substitution box (S-box). This method is vital in many encryption techniques including the AES in creation of confusion and diffusion. Cryptographic hash functions such as MD5 and SHA-1 also use SBT to provide large change s in the output for slight change s in the input. Besides encryption, SBT is employed to scramble the data that should not be seen by unauthorized persons, for example, code or data stored in a database [22].

The key subcomponent of SBT is the S-box, which is a type of lookup table; SBT is the transformation generally defined to be non-linear, so reversing the operation without information on the specific S-box involved is not easily done. In decryption an inverse S-box restores substituted bytes to their normal state. In AES there is known the SBT process as the Sub-Bytes step, the aim of which is the substitution of each byte of AES state matrix by the corresponding byte of the S-box, receiving a new state matrix for further stages of encryption [23].

In SBT, non-linearity and complexity is entered which is mandatory to protect data from linear as well differential cryptanalysis. Proper deployment of SBT means that the flow of encryptions and decryption is fast, which is essential for real-time data. The technique can be used widely ranging from Cryptographic protocols and systems not only in the case of encryption but also in the cases like digital signatures, authentication etc. Non-linearity in S-boxes is transferred in such a way that a small change in its input results in a large change in its output referred to as the avalanche effect besides the capacity to stand cryptanalytic attacks. The knowledge of SBT, as well as its use, is mandatory to create effective cryptographic algorithms and fulfill the requirements of information protection in the communications and storage systems [24].

IV. ENHANCE AES BASED ON S-BOX

Improvement of AES based on the S-box implies a set of approaches to improve the security of the algorithm and its efficiency. It is realized that applying a larger S-Box can increase the level of complication but which in turns may raise the ability to withstand attacks. The Dynamic S-box generation, in which S-box is changed according to the key increases the security level when compared to static S-box but requires more computation[25]. To increase the resistance to linear and differential attacks, one has to increase nonlinearity in the S-box, but it should be noted that it is a complex task in terms of design and verifiability. Improvement of S-box calculation by using hardware support, or good algorithms increases efficiency but can come up with increased costs and dependence on hardware resources. Thus, by including the advanced cryptographic techniques, the S-box can be made to be more secure, but this comes with the disadvantage of more complexity. Employment of updates keeps protection dynamic but can introduce operational problems if not properly carried out on regular basis for S-box[26]. Table (1) Describe the Enhancing methods for AES Based on S-Box.

Table 1: Describe the Enhancing methods for AES Based on S-Box[27][28]

Strategy	Description	Benefits	Challenges
Larger S-box	Use a larger permutation of values than 256 bytes.	Increased security; enhanced resistance to certain attacks.	Compatibility issues; increased complexity.
Dynamic S-box Generation	Generate the S-box dynamically based on encryption parameters.	Enhanced security; adaptable encryption scheme.	Performance overhead; complex implementation.
Improve Nonlinearity	Design the S-box with higher nonlinearity.	Stronger resistance to cryptanalysis; better confusion.	Complex design; verification of cryptographic strength.
Enhanced Resistance to Differential Attacks	Modify the S-box to reduce differential characteristics.	Improved resistance to differential attacks.	Design complexity; balancing with performance.
Optimized S-box Computation	Implement hardware- or software-optimized S-box computations.	Increased efficiency; reduced latency.	Hardware dependencies; higher costs.
Advanced Cryptographic Techniques	Use advanced methods like modular arithmetic in different fields.	Stronger cryptographic guarantees; innovative designs.	Complexity; potential compatibility issues.
Periodic S-box Updates	Regularly update or refresh the S-box based on conditions.	Dynamic protection; enhanced security against evolving threats.	Implementation complexity; operational overhead.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 8, pp 182-192, August-2024

https://doi.org/10.47001/IRJIET/2024.808020

Therefore, by applying of these techniques the S-box in AES can be improved from the viewpoint of the security and performance, and it will correspond to modern requirements for cryptographic tools. For instance, using an S-box of greater size ensures a more comprehensive permutation of the values, which makes it possible to raise the vulnerability of the specific attacks. This expands the permutation and limits the number of opportunities for the attacker to predict an entity's permutation based on the previous values[29].

In Dynamic S-box generation, The S-box is further extended from the static to the dynamic one by varying the S-box with each encryption key or IV. Such an approach guarantees that for each encryption process an S-box is used and this makes it difficult for the attackers to decode the encryption since there is no pattern which is repeated. This scheme inevitably adds computational cost since the creation of the S-box during encryption is an extra process[27].

Introducing changes which increase nonlinearity of the S-box helps to improve its immunity to linear and differential cryptanalysis attacks. The rationale for an S-box to possess a high amount of nonlinearity is to increase the complexity of the link between input and output so that further analysis and attempts at deducing or reverse-engineering the substitution are much more difficult. This enhances the nonlinearity of the AES algorithm generally raising the bar on security of the algorithm[30].

Generally, any enhancement in computation of S-box considerably enhances the parameters of encryption and decryption. While hardware designs can take advantage of the combining of specific circuits for the S-box lookups, better algorithms for optimized software computations can be supported on general processors. While these optimizations may help to decrease latency and increase the throughput, sometimes these input/output operations come at a cost, in terms of extra expense and possibly requiring specialized equipment or software[31].

Cryptography provides a method of constructing an S-box that is more secure, for example, through use of modular arithmetic in different fields or through the use of more complicated transformations. These techniques offer higher levels of cryptographic protection and are already enhanced by new ideas that are less susceptible to the emerging threats. However, incorporating such techniques increases the cipher's structure and could pose some challenges on compatibility with other systems[32].

Applying updates to the S-box by time intervals or changes of the key material provides protection against forthcoming threats. In this way, by periodically replacing it, the S-box helps to maintain the cipher's immunity to the kinds of attacks aimed at fixed cryptographic elements. The approach needs to be done in a way that patches do not first create vulnerabilities and that the operating overhead cost is reasonable[33].

In general, these improvements to the S-box in AES can offer better protection up to present and future cryptographic threats and broaden the cipher's adaptability. When each of the above strategies is implemented, it enhances flexible security and performance concerning encryption algorithms.

V. RELATED WORKS

Table (2) provides descriptive details on different studies carried out so far to improve the encryption of the data with special emphasis to S-Box design and AES, ECC and other cryptographic algorithms. There are several studies done in the literature toward the generation of dynamic S-Boxes, round based permutations and novel 3D concepts to enhance the security as well as variability of S-Boxes but these come at the cost of additional computational complexity. The integration of AES and ECC has also been proposed to increase the security of cloud storage whereas the studies which include ECC and RSA have demonstrated that ECC is more efficient for the large data size. Other research has emphasized on speeding up the AES encryption process and improvisation of diffusion processes with algorithms like DES and IDEA. All these works together emphasize the continuous work on the search for the optimal ratio of security, speed, and feasibility in modern cryptographic systems.

Table 2: Related Works Analysis

Researchers	Problem	Method	Results	Strength Points	Limitations
[34]	Initial S-Box construction with a new irreducible polynomial	S-Box values selected by adding a shifting value obtained from the cipher key	Improved S-Box variability	Increased security due to dynamic S-Box	Complexity in S-Box generation process
[35]	Standard S-Box rotation	Rotation for each round, with rotation value calculated by Xoring all values of a round key	Enhanced diffusion in encryption	Simple and effective method for increasing diffusion	Additional computational overhead
[36]	Dynamic S-Box	Initial S-Box similar to	Higher security	Robust dynamic S-	Increased

	generation	original, columns and rows inverted, cipher key generates dynamic S-Box	through dynamic S-Box	Box creation, increased resistance to attacks	complexity, potential performance impact
[37]	Dynamic S-Box based on affine transformation	Variation of affine transformation constant, using the most non-linear letter of the key	Improved non-linearity of the S-Box	Enhanced non-linearity, increased complexity in S-Box creation	Dependent on key characteristics, may not always provide uniform results
[38]	Round-based S-Box permutation	Permutation of S-Box elements using a swap function and a new dynamically generated S-Box	Increased variability and security of S-Box	Dynamic and round-based variation adds an extra layer of security	High computational cost due to frequent S-Box permutations
[39]	New S-Box construction with permutation indexes	Calculates permutation indexes between S-Box elements and constructs a new S-Box (Sboxm) using the encryption key	New S-Box called Sboxm with enhanced security	Methodical and systematic approach to new S-Box creation	Requires precise calculation of permutation indexes, may introduce complexity
[40]	3D concept in encryption key and S-Box	Dynamic S-Box through a rotation obtained with a random value used in the key rotation	Innovative 3D dynamic S-Box with enhanced security	Novel 3D approach provides a unique method of dynamic S-Box generation	Potentially high computational complexity, may require significant processing power
[41]	Enhancing system security in cloud storage	AES and ECC with Shamir secret sharing	Improved system security	Increased security through combined AES and ECC	High computational cost and time expense
[42]	Secure cloud services	Use of AES, DES, and Blowfish	Enhanced data storage efficiency and integrity	Efficient and secure data storage, managed and expedited data access	Computational complexity due to multiple algorithms
[43]	Comparing ECC and RSA for data larger than 264 bits	Study comparing ECC and RSA	ECC outperforms RSA for data > 256 bits	Higher security for smaller data volumes, reduced storage requirements	Specific to data sizes exceeding 256 bits, not universally applicable
[44]	Hybrid approaches for RSS and ECC	Compression, elliptic curve authorities, digest and sign	Basis for development of hybrid algorithms	Supremacy of RSS and ECC analysis	Potentially complex implementation
[45]	Secure and private data security in cloud computing	Hybrid cryptosystem-based solution using AES and ECC	Efficient, powerful, and safe encryption	Combined symmetric and divergent encryption enhances security	Complexity in simultaneously implementing AES and ECC
[46]	Efficient and secure encryption approach for distributed storage	Investigation of common cryptography methods (AES, ECC, RSA)	Identification of secure encryption methods	Systematic comparison provides insights into effective encryption	Long encryption and decryption times
[47]	Strengthening the diffusion process in encryption techniques	Use of DES, IDEA, and PRESENT algorithms relying on S-boxes	Enhanced diffusion process	Stronger encryption due to improved diffusion	Dependency on S-boxes for strength
[48]	AES as a key encryption technique for media applications	Use of AES for encryption of photos and movies	Widespread use of AES in media encryption	Proven effectiveness in various applications	Latency caused by S-box replacement step

[49]	Producing dynamic S-boxes for a stronger AES system	Method for creating dynamic S-boxes with higher nonlinearity	Suitable for encryption applications due to speed and strength	High nonlinearity increases system strength	Dependency on biometric technique for encryption and decryption
[6]	Investigating the nonlinearity effect of the S-box in AES encryption	Study of S-box nonlinearity effect	Conclusion that strong nonlinearity prevents hacking	Important consideration of avalanche effect for S-box strength evaluation	Focus on nonlinearity may overlook other critical factors
[50]	Evaluating the strength of S-boxes based on the avalanche effect	Study of the avalanche effect on S-box strength	Reinforcement of S-box strength evaluation	Emphasis on the criticality of avalanche effect	May not account for all variables affecting S-box strength
[51]	Addressing latency in the AES algorithm	Examination of the S-box replacement step	Identification of S-box as the cause of most latency	Crucial step in AES, providing non-linearity	Latency remains a challenge despite its necessity
[52]	Improving cloud computing information security with two-level cryptographic method	Utilization of symmetric and asymmetric encryption (AES and ECC)	Enhanced information security and client confidence	Prevention of actual data access, privacy, integrity of information	Complexity in implementing two-level cryptographic methods, potential impact on processing speed

VI. CONCLUSIONS

The optimization of the S-Box that is in the AES is one of a kind recognized significant step towards addressing new threats and increasing efficiency. Some of them include; increased size of the S-Boxes, generation of S-Boxes on the fly, nonlinearity each with the benefit of improving the level of protection against cryptanalytic attacks. We have more permutations with larger S-Boxes as well as genetic S-Box generation that is dependent on the encryption key to make the attacks even harder. Higher nonlinearity enhances the immunity of the S-Box against linear and differential cryptosystems, but the enhancement of computation techniques reduces computational complexity. Even with considerations such as those mentioned above, the optimizations discussed here are necessary for preserving the AES's security against contemporary threats. Further research should be done to incorporate these enhancements, at the same time considering computational cost of introduced algorithms and compatibility of the system that is to be designed.

REFERENCES

[1] ETSI, *Quantum Safe Cryptography and Security: An introduction*, no. 8. 2015.

[2] J. Daemen and V. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*. 2002. doi: 10.1007/978-3-662-04722-4.

[3] A.Altigani, S. Hasan, B. Barry, S. Naserelden, M. A. Elsadig, and H. T. Elshoush, "A Polymorphic Advanced Encryption Standard - A Novel Approach," *IEEE Access*, vol. 9, pp. 20191–20207, 2021, doi: 10.1109/ACCESS.2021.3051556.

[4] F. V. Wenceslao, "Enhancing the Performance of the Advanced Encryption Standard(AES)Algorithm Using Multiple Substitution Boxes," *Int. J. Commun. Networks Inf. Secur.*, vol. 10, no. 3, pp. 496–501, 2018, doi: 10.17762/ijcnis.v10i3.3589.

[5] H. S. Zied, A. G. A. Ibrahim, and A. I. Salem, "S-Box Modification for the Block Cipher Algorithms," *Prz. Elektrotechniczny*, vol. 99, no. 4, pp. 278–281, 2023, doi: 10.15199/48.2023.04.48.

[6] K. Mohamed, M. N. Mohammed Pauzi, F. H. Hj Mohd Ali, S. Ariffin, and N. H. Nik Zulkipli, "Study of S-box properties in block cipher," *I4CT 2014 - 1st Int. Conf. Comput. Commun. Control Technol. Proc.*, no. June, pp. 362–366, 2014, doi: 10.1109/I4CT.2014.6914206.

[7] A.Alamsyah, B. Prasetiyo, and Y. Muhammad, "S-box Construction on AES Algorithm using Affine Matrix Modification to Improve Image Encryption Security," *Sci. J. Informatics*, vol. 10, no. 2, pp. 69–82, 2023, doi: 10.15294/sji.v10i2.42305.

[8] K. M. Hosny, M. A. Zaki, N. A. Lashin, M. M. Fouda, and H. M. Hamza, "Multimedia Security Using Encryption: A Survey," *IEEE Access*, vol. 11, no. June, pp. 63027–63056, 2023, doi: 10.1109/ACCESS.2023.3287858.

[9] S. T. Wu and J. R. Chang, "Secure One-Way Hash Function Using Cellular Automata for IoT," *Sustain.*, vol. 15, no. 4, 2023, doi: 10.3390/su15043552.

[10] T. Bin Azad, "Chapter 4 - Understanding XenApp Security," T. B. B. T.-S. C. P. S. in the E. Azad, Ed., Burlington: Syngress, 2008, pp. 259–316. doi:

https://doi.org/10.1016/B978-1-59749-281-2.00004-4.

[11] K. Suresh Babu, R. K B, K. Kiran, T. H. Devi, V. K R, and L. Patnaik, *Authentication of secret information in image Steganography*. 2008. doi: 10.1109/TENCON.2008.4766581.

[12] Q. Zhang, L. Liu, and X. Wei, "Improved algorithm for image encryption based on DNA encoding and multi-chaotic maps," *AEU - Int. J. Electron. Commun.*, vol. 68, pp. 186–192, Mar. 2014, doi: 10.1016/j.aeue.2013.08.007.

[13] X. Wang and S.-X. Gu, "Novel image encryption algorithm based on cycle shift and chaotic system," *Opt. Lasers Eng.*, vol. 68, May 2015, doi: 10.1016/j.optlaseng.2014.12.025.

[14] M. Kaur, S. Singh, and M. Kaur, "Computational Image Encryption Techniques: A Comprehensive Review," *Math. Probl. Eng.*, vol. 2021, Jul. 2021, doi: 10.1155/2021/5012496.

[15] B. Mondal, S. Singh, and P. Kumar, "A secure image encryption scheme based on cellular automata and chaotic skew tent map," *J. Inf. Secur. Appl.*, vol. 45, pp. 117–130, 2019, doi: 10.1016/j.jisa.2019.01.010.

[16] N. Aleisa, "A comparison of the 3DES and AES encryption standards," *Int. J. Secur. its Appl.*, vol. 9, no. 7, pp. 241–246, 2015, doi: 10.14257/ijsia.2015.9.7.21.

[17] J. Nechvatal, E. Barker, L. Bassham, M. Dworkin, and E. Roback, "Report on the Development of the Advanced Encryption Standard (AES)," vol. 106, no. 3, 2001.

[18] L. Rouquette, D. Gérault, M. Minier, and C. Solnon, "And Rijndael?: Automatic Related-Key Differential Analysis of Rijndael," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 13503 LNCS, pp. 150–175, 2022, doi: 10.1007/978-3-031-17433-9_7.

[19] A.Nadjia and A. Mohamed, "AES IP for hybrid cryptosystem RSA-AES," *12th Int. Multi-Conference Syst. Signals Devices, SSD 2015*, no. March, 2015, doi: 10.1109/SSD.2015.7348109.

[20] K. L. Narayanan, P. Kannan, and S. E. Rajavel, "A Comparative Study and Performance Evaluation of Cryptographic Algorithms : AES and Blowfish," *Int. J. Adv. Res. Trends Eng. Technol.*, vol. 1, no. 3, pp. 81–86, 2014.

[21] F. S. Hossain, M. L. Ali, and M. A. Al Abedin Syed, "A very low power and high throughput AES processor," *14th Int. Conf. Comput. Inf. Technol. ICCIT 2011*, no. December, pp. 339–343, 2011, doi: 10.1109/ICCITechn.2011.6164810.

[22] Abdullah-All-Tanvir, I. Ali Khandokar, A. K. M. Muzahidul Islam, S. Islam, and S. Shatabda, "A gradient boosting classifier for purchase intention prediction of online shoppers," *Heliyon*, vol. 9, no. 4, p. e15163, 2023, doi: 10.1016/j.heliyon.2023.e15163.

[23] G. N. Selimis, A. P. Kakarountas, A. P. Fournaris, A. Milidonis, and O. Koufopavlou, "A Low Power Design for Sbox Cryptographic Primitive of Advanced Encryption Standard for Mobile End-Users," *J. Low Power Electron.*, vol. 3, no. 3, pp. 327–336, 2007, doi: 10.1166/jolpe.2007.139.

[24] A.AlRababah, "Digital Image Encryption Implementations Based on AES Algorithm," *VAWKUM Trans. Comput. Sci.*, vol. 13, no. 1, p. 1, 2017, doi: 10.21015/vtcs.v13i1.453.

[25] M. E. Hameed, M. M. Ibrahim, and N. A. Manap, "Review on improvement of advanced encryption standard (AES) algorithm based on time execution, differential cryptanalysis and level of security," *J. Telecommun. Electron. Comput. Eng.*, vol. 10, no. 1, pp. 139–145, 2018.

[26] M. Veshala and T. Srinivasulu, *Improving the Energy Efficiency of the WSNs by Optimal Relay Node Selection using Modified Gravitational Search Approach*, no. Iccmc. 2021. doi: 10.1109/ICCMC51019.2021.9418240.

[27] A.Singh, P. Agarwal, and M. Chand, "Analysis of Development of Dynamic S-Box Generation," *Comput. Sci. Inf. Technol.*, vol. 5, no. 5, pp. 154–163, 2017, doi: 10.13189/csit.2017.050502.

[28] A.Jana, A. K. Kundu, and G. Paul, *More Vulnerabilities of Linear Structure Sbox-Based Ciphers Reveal Their Inability to Protect DFA.*

[29] J. Cui, L. Huang, H. Zhong, C. Chang, and W. Yang, "An improved AES S-box and its performance analysis," *Int. J. Innov. Comput. Inf. Control*, vol. 7, no. 5 A, pp. 2291–2302, 2011.

[30] W. Zhang and E. Pasalic, "Highly nonlinear balanced S-boxes with good differential properties," *IEEE Trans. Inf. Theory*, vol. 60, no. 12, pp. 7970–7979, 2014, doi: 10.1109/TIT.2014.2360880.

[31] M. Gupta and A. Sinha, "Enhanced-AES encryption mechanism with S-box splitting for wireless sensor networks," *Int. J. Inf. Technol.*, vol. 13, Feb. 2021, doi: 10.1007/s41870-021-00626-w.

[32] O. Sahoo, D. Kole, and H. Rahaman, *An Optimized S-Box for Advanced Encryption Standard (AES) Design*. 2012. doi: 10.1109/ICACC.2012.35.

[33] S. Sudhakar, A. Akashwar, M. Ajay Someshwar, T. Dhaneshguru, and M. Prem Kumar, "Improving Security Using Modified S-Box for AES Cryptographic Primitives," *Open Access by IOS Press Distrib. under terms Creat. Commons Attrib.*, pp. 830–835, 2021, doi: 10.3233/apc210288.

[34] F. Mohammad, A. E. Rohiem, and A. Elbayoumy, "A Novel S-box of AES Algorithm Using Variable Mapping Technique," *Int. Conf. Aerosp. Sci. Aviat. Technol.*, vol. 13, no. AEROSPACE SCIENCES, pp. 1–10, 2009, doi: 10.21608/asat.2009.23494.

[35] J. Juremi, R. Mahmod, S. Sulaiman, and J. Ramli, "Enhancing Advanced Encryption Standard S-Box Generation Based on Round Key," *Int. J. Cyber-Security Digit. Forensics*, vol. 1, no. 3, pp. 183–188, 2012.

[36] R. Hosseinkhani, "Using Cipher Key to Generate Dynamic S-Box in AES Cipher System," *Int. J. Comput. Sci. Secur. (IJCSS), Vol. Issue 2012*, no. 6, pp. 19–28, 2012.

[37] N. Tsedura and C. Chibaya, *Effects of Runtime Generated S-Boxes to the DES Model*. 2020. doi: 10.1109/IMITEC50163.2020.9334146.

[38] A.Alabaichi and A. Salih, *Enhance security of advance encryption standard algorithm based on key-dependent S-box*. 2015. doi: 10.1109/ICDIPC.2015.7323004.

[39] K. Kazlauskas, G. Vaicekauskas, and R. Smaliukas, "An Algorithm for Key-Dependent S-Box Generation in Block Cipher System," *Inform.*, vol. 26, no. 1, pp. 51–65, 2015, doi: 10.15388/Informatica.2015.38.

[40] Z. Rahaman, A. Diana, M. Akter, and A. Newaz, "A Novel Structure of Advance Encryption Standard with 3-Dimensional Dynamic S-box and Key Generation Matrix," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 2, 2017, doi: 10.14569/ijacsa.2017.080241.

[41] D. Shukla, V. Trivedi, and (Dr.) Munesh Trivedi, "Encryption algorithm in cloud computing," *Mater. Today Proc.*, vol. 37, Aug. 2020, doi: 10.1016/j.matpr.2020.07.452.

[42] H. S. Yahia *et al.*, "Comprehensive Survey for Cloud Computing Based Nature-Inspired Algorithms Optimization Scheduling," *Asian J. Res. Comput. Sci.*, no. May, pp. 1–16, 2021, doi: 10.9734/ajrcos/2021/v8i230195.

[43] I.A. Khan and R. Qazi, "Data Security in Cloud Computing Using Elliptic Curve Cryptography," *Online)*, vol. 1, no. 1, pp. 2664–9519, 2019.

[44] M. Manna and M. Ali Mohammed A, "Data Encryption Scheme for Large Data Scale in Cloud Computing," *J. Telecommun. Electron. Comput. Eng.*, vol. 9, no. 2–12, pp. 1–5, 2017.

[45] Y. Li, K. Gai, L. Qiu, M. Qiu, and H. Zhao, "Intelligent cryptography approach for secure distributed big data storage in cloud computing," *Inf. Sci. (Ny).*, vol. 387, pp. 103–115, 2017, doi: https://doi.org/10.1016/j.ins.2016.09.005.

[46] Z. R. Saeed, Z. Ayop, N. Azma, and B. M. Rizuan, "Improved Cloud Storage Security of Using Three Layers Cryptography Algorithms," *Int. J. Comput. Sci. Inf. Secur.*, vol. 16, no. 10, pp. 34–39, 2018.

[47] K. Jithendra and T. K. Shahana, *A New Efficient Sbox for Strengthening PRESENT Like Block Ciphers Against Linear Cryptanalysis*. 2019. doi: 10.1109/ICICICT46008.2019.8993397.

[48] D. Alsaffar *et al.*, "Image Encryption Based on AES and RSA Algorithms," *2020 3rd Int. Conf. Comput. Appl. \& Inf. Secur.*, pp. 1–5, 2020.

[49] A.Y. Al-Dweik, I. Hussain, M. Saleh, and M. T. Mustafa, "A novel method to generate key-dependent s-boxes with identical algebraic properties," *J. Inf. Secur. Appl.*, vol. 64, pp. 1–20, 2022, doi: 10.1016/j.jisa.2021.103065.

[50] K. Mohamed, M. N. M. Pauzi, F. H. H. M. Ali, and S. Ariffin, "Analyse On Avalanche Effect In Cryptography Algorithm," *Proc. Int. Conf. Sustain. Pract. Dev. Urban. (IConsPADU 2021), 16 Novemb. 2021, Univ. Selangor (UNISEL), Malaysia*, vol. 3, no. November, pp. 610–618, 2022, doi: 10.15405/epms.2022.10.57.

[51] D. Kodzo, M. Hodowu, D. R. Korda, and E. Danso Ansong, "An Enhancement of Data Security in Cloud Computing with an Implementation of a Two-Level Cryptographic Technique, using AES and ECC Algorithm," *Int. J. Eng. Res. Technol.*, vol. 9, no. March 2021, pp. 2278–0181, 2020.

[52] X. Chen, J. Li, J. Ma, Q. Tang, and W. Lou, "New algorithms for secure outsourcing of modular exponentiations," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 9, pp. 2386–2396, 2014, doi: 10.1109/TPDS.2013.180.

Citation of this Article:

Sura Nabil Hamed Alsweedy, & Sufyan Salim Aldabbagh. (2024). Enhancing AES Security through Advanced S-Box Design: Strategies and Solutions. *International Research Journal of Innovations in Engineering and Technology - IRJIET*, 8(8), 182-192. Article DOI https://doi.org/10.47001/IRJIET/2024.808020

****